An Algorithm for Pronominal Anaphora Resolution

Shalom Lappin*
SOAS, University of London

Herbert J. Leass†
Sietec Systemtechnik

This paper presents an algorithm for identifying the noun phrase antecedents of third person pronouns and lexical anaphors (reflexives and reciprocals). The algorithm applies to the syntactic representations generated by McCord's Slot Grammar parser and relies on salience measures derived from syntactic structure and a simple dynamic model of attentional state. Like the parser, the algorithm is implemented in Prolog. The authors have tested it extensively on computer manual texts and conducted a blind test on manual text containing 360 pronoun occurrences. The algorithm successfully identifies the antecedent of the pronoun for 86% of these pronoun occurrences. The relative contributions of the algorithm's components to its overall success rate in this blind test are examined. Experiments were conducted with an enhancement of the algorithm that contributes statistically modelled information concerning semantic and real-world relations to the algorithm's decision procedure. Interestingly, this enhancement only marginally improves the algorithm's performance (by 2%). The algorithm is compared with other approaches to anaphora resolution that have been proposed in the literature. In particular, the search procedure of Hobbs' algorithm was implemented in the Slot Grammar framework and applied to the sentences in the blind test set. The authors' algorithm achieves a higher rate of success (4%) than Hobbs' algorithm. The relation of the algorithm to the centering approach is discussed, as well as to models of anaphora resolution that invoke a variety of informational factors in ranking antecedent candidates.

1. Introduction

We present an algorithm for identifying both intrasentential and intersentential antecedents of pronouns in text. We refer to this algorithm as RAP (Resolution of Anaphora Procedure). RAP applies to the syntactic structures of McCord's (1990, 1993, in press) Slot Grammar parser, and like the parser, it is implemented in Prolog. It relies on measures of salience derived from syntactic structure and a simple dynamic model of attentional state to select the antecedent noun phrase (NP) of a pronoun from a list of candidates. It does not employ semantic conditions (beyond those implicit in grammatical number and gender agreement) or real-world knowledge in evaluating candidate antecedents; nor does it model intentional or global discourse structure (as in Grosz and Sidner 1986).

* School of Oriental and African Studies, University of London, London WC1H OXG, UK. E-mail: slappin@clus1.ulcc.ac.uk
Most of the first author's work on this paper was done while he was a Research Staff Member in the Computer Science Department of the IBM T.J. Watson Research Center.
† Sietec Systemtechnik (Siemens AG), D-13623 Berlin, Germany. E-mail: leass@sietec.de
The second author's work on this paper was done while he was a visiting scientist at the IBM Germany Scientific Center.

In Section 2 we present RAP and discuss its main properties. We provide examples of its output for different sorts of cases in Section 3. Most of these examples are taken from the computer manual texts on which we trained the algorithm. We give the results of a blind test in Section 4, as well as an analysis of the relative contributions of the algorithm's components to the overall success rate. In Section 5 we discuss a procedure developed by Dagan (1992) for using statistically measured lexical preference patterns to reevaluate RAP's salience rankings of antecedent candidates. We present the results of a comparative blind test of RAP and this procedure. Finally, in Section 6 we compare RAP to several other approaches to anaphora resolution that have been proposed in the computational literature.

2. The Anaphora Resolution Algorithm

RAP contains the following main components.

- An intrasentential syntactic filter for ruling out anaphoric dependence of a pronoun on an NP on syntactic grounds (This filter is presented in Lappin and McCord 1990a.)
- 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 lexical anaphor (reciprocal or reflexive pronoun) within the same sentence (This algorithm is presented in Lappin and McCord 1990b.)
- A procedure for assigning values to several salience parameters (grammatical role, parallelism of grammatical roles, frequency of mention, proximity, and sentence recency) for an NP. (Earlier versions of these procedures are presented in Leass and Schwall 1991.) This procedure employs a grammatical role hierarchy according to which the evaluation rules assign higher salience weights to (i) subject over non-subject NPs, (ii) direct objects over other complements, (iii) arguments of a verb over adjuncts and objects of prepositional phrase (PP) adjuncts of the verb, and (iv) head nouns over complements of head nouns.[1]
- A procedure for identifying anaphorically linked NPs as an equivalence class for which a global salience value is computed as the sum of the salience values of its elements.
- A decision procedure for selecting the preferred element of a list of antecedent candidates for a pronoun.

1 This hierarchy is more or less identical to the NP accessibility hierarchy proposed by Keenan and Comrie (1977). Johnson (1977) uses a similar grammatical role hierarchy to specify a set of constraints on syntactic relations, including reflexive binding. Lappin (1985) employs it as a salience hierarchy to state a non-coreference constraint for pronouns. Guenthner and Lehmann (1983) use a similar salience ranking of grammatical roles to formulate rules of anaphora resolution. Centering approaches to anaphora resolution use similar hierarchies as well (Brennan, Friedman, and Pollard 1987; Walker, Iida, and Cote 1990).

2.1 Some Preliminary Details

RAP has been implemented for both ESG and GSG (English and German Slot Grammars); we will limit ourselves here to a discussion of the English version. The differences between the two versions are at present minimal, primarily owing to the fact that we have devoted most of our attention to analysis of English. As with Slot Grammar systems in general (McCord 1989b, 1993, in press), an architecture was adopted that "factors out" language-specific elements of the algorithm.

We have integrated RAP into McCord's (1989a, 1989b) Logic-Based Machine Translation System (LMT). (We are grateful to Michael McCord and Ullrike Schwall for their help in implementing this integration.) When the algorithm identifies the antecedent of a pronoun in the source language, the agreement features of the head of the NP corresponding to the antecedent in the target language are used to generate the pronoun in the target language. Thus, for example, neuter third person pronouns in English are mapped into pronouns with the correct gender feature in German, in which inanimate nouns are marked for gender.

RAP operates primarily on a clausal representation of the Slot Grammar analysis of the current sentence in a text (McCord et al. 1992). The clausal representation consists of a set of Prolog unit clauses that provide information on the head–argument and head–adjunct relations of the phrase structure that the Slot Grammar assigns to a sentence (phrase). Clausal representations of the previous four sentences in the text are retained in the Prolog workspace. The discourse representation used by our algorithm consists of these clausal representations, together with additional unit clauses declaring discourse referents evoked by NPs in the text and specifying anaphoric links among discourse referents.[2] All information pertaining to a discourse referent or its evoking NP is accessed via an identifier (ID), a Prolog term containing two integers. The first integer identifies the sentence in which the evoking NP occurs, with the sentences in a text being numbered consecutively. The second integer indicates the position of the NP's head word in the sentence.

2.1.1 The Syntactic Filter on Pronoun–NP Coreference. The filter consists of six conditions for NP–pronoun non-coreference within a sentence. To state these conditions, we use the following terminology. The *agreement features* of an NP are its number, person, and gender features. We will say that a phrase P is in the *argument domain* of a phrase N iff P and N are both arguments of the same head. We will say that 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. P is in the *NP domain* of N iff N is the determiner of a noun Q and (i) P is an argument of Q, or (ii) P is the object of a preposition PREP and PREP is an adjunct of Q. A phrase P is *contained in* a phrase Q iff (i) P is either an argument or an adjunct of Q, i.e., P is *immediately contained in* Q, or (ii) P is immediately contained in some phrase R, and R is contained in Q.

A pronoun P is non-coreferential with a (non-reflexive or non-reciprocal) noun phrase N if any of the following conditions hold:

1. P and N have incompatible agreement features.
2. P is in the argument domain of N.
3. P is in the adjunct domain of N.

2 The number of sentences whose syntactic representations are retained is a parametrically specified value of the algorithm. Our decision to set this value at four is motivated by our experience with the technical texts we have been working with.

4. P is an argument of a head H, N is not a pronoun, and N is contained in H.
5. P is in the NP domain of N.
6. P is a determiner of a noun Q, and N is contained in Q.

Examples of coindexings that would be rejected by these conditions are given in Figure 1.

Condition 1:
The woman$_i$ said that he$_i$ is funny.

Condition 2:
She$_i$ likes her$_i$.
John$_i$ seems to want to see him$_i$.

Condition 3:
She$_i$ sat near her$_i$.

Condition 4:
He$_i$ believes that the man$_i$ is amusing.
This is the man$_i$ he$_i$ said John$_i$ wrote about.

Condition 5:
John$_i$'s portrait of him$_i$ is interesting.

Condition 6:
His$_i$ portrait of John$_i$ is interesting.
His$_i$ description of the portrait by John$_i$ is interesting.

Figure 1
Conditions on NP–pronoun non-coreference (examples).

2.1.2 Test for Pleonastic Pronouns. The tests are partly syntactic and partly lexical. A class of modal adjectives is specified. It includes the following items (and their corresponding morphological negations, as well as comparative and superlative forms).

necessary	possible	certain	likely	important
good	useful	advisable	convenient	sufficient
economical	easy	desirable	difficult	legal

A class of cognitive verbs with the following elements is also specified.

recommend think believe know anticipate assume expect

It appearing in the constructions of Figure 2 is considered pleonastic (Cogv-ed = passive participle of cognitive verb); syntactic variants of these constructions (*It is not/may be* Modaladj..., *Wouldn't it be* Modaladj..., etc.) are recognized as well.

To our knowledge, no other computational treatment of pronominal anaphora resolution has addressed the problem of pleonastic pronouns. It could be argued that recognizing pleonastic uses of pronouns is a task for levels of syntactic/semantic analysis that precede anaphora resolution. With the help of semantic classes defined in the lexicon, it should be possible to include exhaustive tests for these constructions in

It is **Modaladj** that **S**
It is **Modaladj** (for **NP**) to **VP**
It is **Cogv-ed** that **S**
It seems/appears/means/follows (that) **S**
NP makes/finds it **Modaladj** (for **NP**) to **VP**
It is time to **VP**
It is thanks to **NP** that **S**

Figure 2
Pleonastic uses of *it*.

analysis grammars.[3]

2.1.3 The Anaphor Binding Algorithm. The notion *higher argument slot* used in the following formulation of the binding algorithm is defined by the following hierarchy of argument slots:

subj > agent > obj > (iobj|pobj)

Here subj is the surface subject slot, agent is the deep subject slot of a verb heading a passive VP, obj is the direct object slot, iobj is the indirect object slot, and pobj is the object of a PP complement of a verb, as in *put NP on NP*. We assume the definitions of argument domain, adjunct domain, and NP domain given above.

A noun phrase N is a possible antecedent binder for a lexical anaphor (i.e., reciprocal or reflexive pronoun) A iff N and A do not have incompatible agreement features, and one of the following five conditions holds.

1. A is in the argument domain of N, and N fills a higher argument slot than A.
2. A is in the adjunct domain of N.
3. A is in the NP domain of N.
4. N is an argument of a verb V, there is an NP Q in the argument domain or the adjunct domain of N such that Q has no noun determiner, and (i) A is an argument of Q, or (ii) A is an argument of a preposition PREP and PREP is an adjunct of Q.
5. A is a determiner of a noun Q, and (i) Q is in the argument domain of N and N fills a higher argument slot than Q, or (ii) Q is in the adjunct domain of N.

Examples of bindings licensed by these conditions are given in Figure 3.

2.1.4 Salience Weighting. Salience weighting is accomplished using *salience factors*. A given salience factor is associated with one or more discourse referents. These discourse referents are said to be in the factor's *scope*. A weight is associated with each

3 ESG does, in fact, recognize some pleonastic uses of *it*, viz. in constructions involving extraposed sentential subjects, as in *It surprised me that he was there*. A special slot, subj(it), is used. We expect that enhancements to ESG and to the Slot Grammar English lexicon will ultimately render our tests for pleonastic pronouns redundant.

Condition 1:
They$_i$ wanted to see themselves$_i$.
Mary knows the people$_i$ who John introduced to each other$_i$.

Condition 2:
He$_i$ worked by himself$_i$.
Which friends$_i$ plan to travel with each other$_i$?

Condition 3:
John likes Bill$_i$'s portrait of himself$_i$.

Condition 4:
They$_i$ told stories about themselves$_i$.

Condition 5:
[John and Mary]$_i$ like each other$_i$'s portraits.

Figure 3
Conditions for antecedent NP–lexical anaphor binding.

factor, reflecting its relative contribution to the total salience of individual discourse referents. Initial weights are degraded in the course of processing.

The use of salience factors in our algorithm is based on Alshawi's (1987) context mechanism. Other than sentence recency, the factors used in RAP differ from Alshawi's and are more specific to the task of pronominal anaphora resolution. Alshawi's framework is designed to deal with a broad class of language interpretation problems, including reference resolution, word sense disambiguation, and the interpretation of implicit relations. While Alshawi does propose emphasis factors for memory entities that are "referents for noun phrases playing syntactic roles regarded as foregrounding the referent" (Alshawi 1987, p. 17), only topics of sentences in the passive voice and the agents of certain *be* clauses receive such emphasis in his system. Our emphasis salience factors realize a much more detailed measure of structural salience.

Degradation of salience factors occurs as the first step in processing a new sentence in the text. All salience factors that have been assigned prior to the appearance of this sentence have their weights degraded by a factor of two. When the weight of a given salience factor reaches zero, the factor is removed.

A *sentence recency* salience factor is created for the current sentence. Its scope is all discourse referents introduced by the current sentence.

The discourse referents evoked by the current sentence are tested to see whether other salience factors should apply. If at least one discourse referent[4] satisfies the conditions for a given factor type, a new salience factor of that type is created, with the appropriate discourse referents in its scope.

In addition to sentence recency, the algorithm employs the following salience factors:

Subject emphasis
Existential emphasis: predicate nominal in an existential construction, as in

`There are only` *`a few restrictions`* `on LQL query construction for WordSmith.`

4 In this paper we do not distinguish between properties of a discourse referent and properties of the NP that evokes it.

Table 1
Salience factor types with initial weights

Factor type	Initial weight
Sentence recency	100
Subject emphasis	80
Existential emphasis	70
Accusative emphasis	50
Indirect object and oblique complement emphasis	40
Head noun emphasis	80
Non-adverbial emphasis	50

Accusative emphasis: direct object (i.e., verbal complement in accusative case)
Indirect object and oblique complement emphasis
Head noun emphasis: any NP not contained in another NP, using the Slot Grammar notion of "containment within a phrase" (see Section 2.1.1). This factor increases the salience value of an NP that is not embedded within another NP (as its complement or adjunct). Examples of NPs *not* receiving head noun emphasis are

> the configuration information copied by *Backup configuration*
>
> the assembly in *bay C*
>
> the connector labeled *P3* on *the flat cable*

Non-adverbial emphasis: any NP not contained in an adverbial PP demarcated by a separator. Like head noun emphasis, this factor penalizes NPs in certain embedded constructions. Examples of NPs *not* receiving non-adverbial emphasis are

> Throughout *the first section* of *this guide*, these symbols are also used ...
>
> In *the Panel definition panel*, select the ''Specify'' option from the action bar.

The initial weights for each of the above factor types are given in Table 1. Note that the relative weighting of some of these factors realizes a hierarchy of grammatical roles.[5]

2.1.5 Equivalence Classes. We treat the antecedent–anaphor relation in much the same way as the "equality" condition of Discourse Representation Theory (DRT) (Kamp 1981), as in

$$u = y.$$

This indicates that the discourse referent u, evoked by an anaphoric NP, is anaphorically linked to a previously introduced discourse referent y. To avoid confusion with

5 The specific values of the weights are arbitrary. The significance of the weighting procedure is in the comparative relations among the factors as defined by the weights. We have determined the efficacy of this relational structure of salience factors (and refined it) experimentally (see Section 4.2).

mathematical equality (which, unlike the relation discussed here, is symmetric), we represent the relation between an anaphor u and its antecedent y by

$$\text{y antecedes u.}$$

Two discourse referents u and y are said to be *co-referential*,[6] written as

$$\text{coref}(u, y)$$

if any of the following holds:

- y antecedes u
- u antecedes y
- z antecedes u for some discourse referent z and coref(z,y)
- z antecedes y for some z and coref(z,u)

Also, coref(u,u) is true for any discourse referent u. The coref relation defines equivalence classes of discourse referents, with all discourse referents in an "anaphoric chain" forming one class:

$$\text{equiv}(u) = \{y \mid \text{coref}(u, y)\}$$

Each equivalence class of discourse referents (some of which consist of only one member) has a *salience weight* associated with it. This weight is the sum of the current weight of all salience factors in whose scope at least one member of the equivalence class lies.

Equivalence classes, along with the sentence recency factor and the salience degradation mechanism, constitute a dynamic system for computing the relative attentional prominence of denotational NPs in text.

2.2 The Resolution Procedure

RAP's procedure for identifying antecedents of pronouns is as follows.

1. Create a list of IDs for all NPs in the current sentence and classify them as to their type (definite NP, pleonastic pronoun, other pronoun, indefinite NP).
2. Examine all NPs occurring in the current sentence.
   a. Distinguish among NPs that evoke new discourse referents, those that evoke discourse referents which are presumably coreferential with already listed discourse referents, and NPs that are used non-referentially.
   b. Apply salience factors to the discourse referents evoked in the previous step as appropriate.
   c. Apply the syntactic filter and reflexive binding algorithm (first phase).

6 We have not attempted to distinguish among various types of anaphoric relations between discourse referents. Our use of "coreference" is in the spirit of Sidner's (1981) "co-specification" and Webber's (1988) "reference$_m$."

(i) If the current sentence contains any personal or possessive pronouns, a list of pairs of IDs from the current sentence is generated. This list contains the pronoun–NP pairs in the sentence for which coreference can be ruled out on syntactic grounds (using the conditions stated above).

(ii) If the current sentence contains any lexical anaphors (i.e., reciprocal or reflexive pronouns), a list of ID pairs is generated. Each lexical anaphor is paired with all of its possible antecedent binders.

d. If any non-pleonastic pronouns are present in the current sentence, attempt to identify their antecedents. Resolution is attempted in the order of pronoun occurrence in the sentence.

In the case of *lexical anaphors* (reflexive or reciprocal pronouns), the possible antecedent binders were identified by the anaphor binding algorithm. If more than one candidate was found, the one with the highest salience weight was chosen (see second example of Section 3.1).

In the case of *third person pronouns*, resolution proceeds as follows:

1. A list of possible antecedent candidates is created. It contains the most recent discourse referent of each equivalence class. The salience weight of each candidate is calculated and included in the list. The salience weight of a candidate can be modified in several ways:

   a. If a candidate follows the pronoun, its salience weight is reduced substantially (i.e., cataphora is strongly penalized).
   b. If a candidate fills the same slot as the pronoun, its weight is increased slightly (i.e., parallelism of grammatical roles is rewarded).

   It is important to note that, unlike the salience factors described in Section 2.1.4, these modifications of the salience weights of candidates are local to the the resolution of a particular pronoun.

2. A salience threshold is applied; only those candidates whose salience weight is above the threshold are considered further.

3. The possible agreement features (number and gender) for the pronoun are determined. The possible *sg* (singular) and *pl* (plural) genders are determined; either of these can be a disjunction or nil. Pronominal forms in many languages are ambiguous as to number and gender; such ambiguities are taken into account by RAP's morphological filter and by the algorithm as a whole. The search splits to consider singular and plural antecedents separately (steps 4–6) to allow a general treatment of number ambiguity (as in the Spanish possessive pronoun *su* or the German pronoun *sie* occurring as an accusative object).

4. The best *sg* candidate (if any) is selected:

   a. If no *sg* genders were determined for the pronoun, proceed to Step 5.
   b. Otherwise, apply the morphological filter.

c. The syntactic filter is applied, using the list of disjoint pronoun-NP pairs generated earlier. The filter excludes any candidate paired in the list with the pronoun being resolved, as well as any candidate that is anaphorically linked to an NP paired with the pronoun.
d. If more than one candidate remains, choose the candidate with the highest salience weight. If several candidates have (exactly) the highest weight, choose the candidate closest to the anaphor. Proximity is measured on the surface string and is not directional.
e. The remaining candidate is considered the best *sg* candidate.

5. The best *pl* candidate (if any) is selected. The procedure parallels that outlined above for the best *sg* candidate:

a. If no *pl* gender is specified for the pronoun, proceed to Step 6.
b. Otherwise, apply the morphological filter.
c. Apply the syntactic filter.
d. If more than one candidate remains, choose the candidate with the highest salience weight; if several candidates have the highest weight, choose the candidate closest to the anaphor.
e. The remaining candidate is considered the best *pl* candidate.

6. Given the best *sg* and *pl* candidates, find the best overall candidate:

a. If a *sg* candidate was found, but no *pl* candidate, or vice versa, choose that candidate as the antecedent.
b. If both a *sg* and a *pl* candidate were found, choose the candidate with the greater salience weight (this will never arise in analysis of English text, as all English pronominal forms are unambiguous as to number).

7. The selected candidate is declared to be the antecedent of the pronoun.

The following properties of RAP are worth noting. First, it applies a powerful syntactic and morphological filter to lists of pronoun–NP pairs to reduce the set of possible NP antecedents for each pronoun. Second, NP salience measures are specified largely in terms of syntactic properties and relations (as well as frequency of occurrence). These include a hierarchy of grammatical roles, level of phrasal embedding, and parallelism of grammatical role. Semantic constraints and real-world knowledge play no role in filtering or salience ranking. Third, proximity of an NP relative to a pronoun is used to select an antecedent in cases in which several candidates have equal salience weighting. Fourth, intrasentential antecedents are preferred to intersentential candidates. This preference is achieved by three mechanisms:

- An additional salience value is assigned to NPs in the current sentence.
- The salience values of antecedent candidates in preceding sentences are progressively degraded relative to the salience values of NPs in the current sentence.
- Proximity is used to resolve ties among antecedent candidates with equal salience values.

The fifth property which we note is that anaphora is strongly preferred to cataphora.

3. Examples of RAP's Output

RAP generates the list of non-coreferential pronoun–NP pairs for the current sentence, the list of pleonastic pronouns, if any, in the current sentence, the list of possible antecedent NP–lexical anaphor pairs, if any, for the current sentence, and the list of pronoun–antecedent NP pairs that it has identified, for which antecedents may appear in preceding sentences in the text. Each NP appearing in any of the first three lists is represented by its lexical head followed by the integer that corresponds to its position in the sequence of tokens in the input string of the current sentence. The NPs in the pairs of the pronoun–antecedent list are represented by their lexical heads followed by their IDs, displayed as a list of two integers.[7]

3.1 Lexical Anaphors

```
After installation of the option, the backup copy of the Reference
Diskette was started for the computer to automatically configure
itself.

Antecedent NP--lexical anaphor pairs.

computer.18 - itself.22

Anaphor--Antecedent links.

itself.(1.22) to computer.(1.18)

John talked to Bill about himself.

Antecedent NP--lexical anaphor pairs.

John.1 - himself.6, Bill.4 - himself.6

Anaphor--Antecedent links.

himself.(1.6) to John.(1.1)
```

In the second example, John.(1.1) was preferred to Bill.(1.4) owing to its higher salience weight.

3.2 Pleonastic and Non-Lexical Anaphoric Pronouns in the Same Sentence

```
Most of the copyright notices are embedded in the EXEC, but this
keyword makes it possible for a user-supplied function to have its
own copyright notice.

Non-coreferential pronoun--NP pairs.

it.16 - most.1, it.16 - notice.5, it.16 - keyword.14,
it.16 - function.23, it.16 - user.20, it.16 - notice.29,
it.16 - copyright.28, its.26 - most.1, its.26 - notice.5,
its.26 - notice.29, its.26 - copyright.28
```

7 Recall that the first integer identifies the sentence in which the NP appears, and the second indicates the position of its head word in the sentence.

```
Pleonastic Pronouns.

it.16

Anaphor--Antecedent links.

its.(1.26) to function.(1.23)
```

function.(1.23) and keyword.(1.14) share the highest salience weight of all candidates that pass the morphological and syntactic filters; they are both subjects and therefore higher in salience than the third candidate, EXEC.(1.10). function.(1.23) is then selected as the antecedent owing to its proximity to the anaphor.

3.3 Multiple Cases of Intrasentential Anaphora

```
Because of this, MicroEMACS cannot process an incoming ESC until it
knows what character follows it.

Non-coreferential pronoun--NP pairs.

it.12 - character.15, it.17 - character.15

Anaphor--Antecedent links.

it.(1.12) to MicroEMACS.(1.4)
it.(1.17) to ESC.(1.10)
```

MicroEMACS.(1.4) is preferred over ESC.(1.10) as an antecedent of it.(1.12)—MicroEMACS.(1.4) receives subject emphasis versus the lower object emphasis of ESC.(1.10). In addition, MicroEMACS.(1.4) is rewarded because it fills the same grammatical role as the anaphor being resolved.

In the case of it.(1.17), the parallelism reward works in favor of ESC.(1.10), causing it to be chosen, despite the general preference for subjects over objects.

3.4 Intersentential and Intrasentential Anaphora in the Same Sentence

```
At this point, emacs is waiting for a command.

It is prepared to see if the variable keys are TRUE, and executes
some lines if they are.

Non-coreferential pronoun--NP pairs.

it.1 - key.9, it.1 - line.16, it.1 - they.18, they.18 - it.1

Anaphor--Antecedent links.

it.(2.1) to emacs.(1.5)
they.(2.18) to key.(2.9)
```

3.5 Displaying Discourse Referents

The discourse referents currently defined can be displayed with their salience weights. The display for the two-sentence text of Section 3.4 is as follows: the members of an equivalence class are displayed on one line. Since salience factors from previous sentences are degraded by a factor of two when each new sentence is processed,

discourse referents from earlier sentences that are not members of anaphoric chains extending into the current sentence rapidly become "uncompetitive."

Salience weight	Discourse referent(s)
465	emacs.(1.5) s(it,1).(2.1)
310	s(key,1).(2.9) s(they,1).(2.18)
280	s(line,1).(2.16)
135	s(command,2).(1.10)
90	s(point,4).(1.3)

3.6 Detailed Displays of Salience Weights

```
You have not waited for the file to close.

You may have asked to print on the virtual printer, but it cannot
print until the output file is closed.

Non-coreferential pronoun--NP pairs:

you.1 - printer.10, you.1 - it.13, you.1 - output.19,
you.1 - file.20, it.13 - you.1, it.13 - output.19,
it.13 - file.20

Salience values:
  printer.(2.10) - 270
  file.(1.7) - 190

Salience factor values:
  printer.(2.10)                file.(1.7)
    sentence_rec - 100          sentence_rec - 50
    non_adverbial_emph - 50     non_adverbial_emph - 25
    pobj_emph - 40              subj_emph - 40
    head_emph - 80              head_emph - 40

Local salience factor values:
  file.(1.7)
    parallel_roles_reward - 35

Anaphor--Antecedent links:

  it.(2.13) to printer.(2.10)
```

This example illustrates the strong preference for intrasentential antecedents. printer.(2.10) is selected, despite the fact that it is much lower on the hierarchy of grammatical roles than the other candidate, file.(1.7), which also benefits from the parallelism reward. Degradation of salience weight for the candidate from the previous sentence is substantial enough to offset these factors.

```
The PARTNUM tag prints a part number on the document.

&name.'s initial setting places it on the back cover.
```

```
Non-coreferential pronoun--NP pairs:

it.6 - setting.4, it.6 - cover.10

Salience values:

number.(1.7) - 175
tag.(1.3) - 155
scsym(name).(2.1) - 150
document.(1.10) - 135
PARTNUM.(1.2) - 75

Salience factor values:

  number.(1.7)                tag.(1.3).
    sentence_rec - 50           sentence_rec - 50
    non_adverbial_emph - 25     non_adverbial_emph - 25
    acc_emph - 25               subj_emph - 40
    head_emph - 40              head_emph - 40

  scsym(name).(2.1)           document.(1.10)
    sentence_rec - 100          sentence_rec - 50
    non_adverbial_emph - 50     non_adverbial_emph - 25
  PARTNUM.(1.2)                 pobj_emph - 20
    sentence_rec - 50           head_emph - 40
    non_adverbial_emph - 25

Local salience factor values:

number.(1,7)
  parallel_roles_reward - 35

Anaphor--Antecedent links:

s(it,1).(2.6) to s(number,1).(1.7)
```

Four candidates receive a similar salience weighting in this example. Two potential intrasentential candidates that would have received a high salience ranking, setting.(2.4) and cover.(2.10), are ruled out by the syntactic filter. The remaining intrasentential candidate, scsym(name).(2.1)[8] ranks relatively low, as it is a possessive determiner—it scores lower than two candidates from the previous sentence. The parallelism reward causes number.(1.7) to be preferred.

4. Testing of RAP on Manual Texts

We tuned RAP on a corpus of five computer manuals containing a total of approximately 82,000 words. From this corpus we extracted sentences with 560 occurrences

8 *&name.* is a document formatting symbol: it is replaced by a predefined character string when the text is formatted. ESG treats such symbols as being unspecified for number and gender; number may be assigned during parsing, owing to agreement constraints.

Table 2
Results of training phase

	Total	Intersentential cases	Intrasentential cases
Number of pronoun occurrences	560	89	471
Number of cases that the algorithm resolves correctly	475 (85%)	72 (81%)	403 (86%)

of third person pronouns (including reflexives and reciprocals) and their antecedents.[9]

In the training phase, we refined our tests for pleonastic pronouns and experimented extensively with salience weighting. Our goal was, of course, to optimize RAP's success rate with the training corpus. We proceeded heuristically, analyzing cases of failure and attempting to eliminate them in as general a manner as possible. The parallelism reward was introduced at this time, as it seemed to make a substantial contribution to the overall success rate. A salience factor that was originally present, viz. *matrix emphasis*, was revised to become the *non-adverbial emphasis* factor. In its original form, this factor contributed to the salience of any NP *not* contained in a subordinate clause or in an adverbial PP demarcated by a separator. This was found to be too general, especially since the relative positions of a given pronoun and its antecedent candidates are not taken into account. The revised factor could be thought of as an adverbial *penalty* factor, since it in effect penalizes NPs occurring in adverbial PPs.[10]

We also experimented with the initial weights for the various factors and with the size of the parallelism reward and cataphora penalty, again attempting to optimize RAP's overall success rate. A value of 35 was chosen for the parallelism reward; this is just large enough to offset the preference for subjects over accusative objects. A much larger value (175) was found to be necessary for the cataphora penalty. The final results that we obtained for the training corpus are given in Table 2.

Interestingly, the syntactic–morphological filter reduces the set of possible antecedents to a single NP, or identifies the pronoun as pleonastic in 163 of the 475 cases (34%) that the algorithm resolves correctly.[11] It significantly restricts the size of the candidate list in most of the other cases, in which the antecedent is selected on the basis of salience ranking and proximity. This indicates the importance of a powerful syntactic–morphological filtering component in an anaphora resolution system.

We then performed a blind test of RAP on a test set of 345 sentences randomly selected from a corpus of 48 computer manuals containing 1.25 million words.[12] The results which we obtained for the test corpus (without any further modifications of RAP) are given in Table 3.[13]

This blind test provides the basis for a comparative evaluation of RAP and Dagan's

9 These sentences and those used in the blind test were edited slightly to overcome parse inaccuracies. Rather than revise the lexicon, we made lexical substitutions to improve parses. In some cases constructions had to be simplified. However, such changes did not alter the syntactic relations among the pronoun and its possible antecedents.

For a discussion of ESG's parsing accuracy, see McCord (1993).

10 See comments at the end of Section 4 about refining RAP's measures of structural salience.

11 Forty-three of the pronoun occurrences in the training corpus (~ 8%) were pleonastic; a random sample of 245 pronoun occurrences extracted from our test corpus included 15 pleonastic pronouns (~ 6%).

12 The test set was filtered in order to satisfy the conditions of our experiments on the role of statistically measured lexical preference in enhancing RAP's performance. See Section 5.1 for a discussion of these

Table 3
Results of blind test

	Total	Intersentential cases	Intrasentential cases
Number of pronoun occurrences	360	70	290
Number of cases that the algorithm resolves correctly	310 (86%)	52 (74%)	258 (89%)

(1992) system, RAPSTAT, which employs both RAP's salience weighting mechanism and statistically measured lexical preferences, as well as for a detailed analysis of the relative contributions of the various elements of RAP's salience weighting mechanism to its overall success rate. We will discuss the blind test in greater detail in the following sections.

4.1 Limitations of the Current Algorithm

Several classes of errors that RAP makes are worthy of discussion. The first occurs with many cases of intersentential anaphora, such as the following:

```
This green indicator is lit when the controller is on.

It shows that the DC power supply voltages are at the correct
levels.
```

Morphological and syntactic filtering exclude all possible intrasentential candidates. Because the level of sentential embedding does not contribute to RAP's salience weighting mechanism, indicator.(1.3) and controller.(1.8) are ranked equally, since both are subjects. RAP then erroneously chooses controller.(1.8) as the antecedent, since it is closer to the pronoun than the other candidate.

The next class of errors involves antecedents that receive a low salience weighting owing to the fact that the evoking NP is embedded in a matrix NP or is in another structurally nonprominent position (such as object of an adverbial PP).

```
The users you enroll may not necessarily be new to the system
and may already have a user profile and a system distribution
directory entry.

&ofc. checks for the existence of these objects and only
creates them as necessary.
```

Despite the general preference for intrasentential candidates, user.(1.2) is selected as the antecedent, since the only factor contributing to the salience weight of object.(2.8) is sentence recency. Selectional restrictions or statistically measured lexical preferences (see Section 5) could clearly help in at least some of these cases.

In another class of cases, RAP fails because semantic/pragmatic information is required to identify the correct antecedent.

conditions.

13 Proper resolution was determined by a consensus of three opinions, including that of the first author.

Table 4
Relative contribution of elements of salience weighting mechanism

	Total correct	Correctly disagrees with RAP	Incorrectly disagrees with RAP
I	310 (86%)		
II	308 (86%)	2	4
III	308 (86%)	2	4
IV	302 (84%)	3	11
V	301 (84%)		9
VI	297 (83%)	12	25
VII	294 (82%)	1	17
VIII	231 (64%)		79
IX	212 (59%)	21	119
X	184 (51%)	17	143

```
Again, the Migration Aid produces an exception report
automatically at the end of every migration run.

As you did with the function, use it to verify that the items
have been restored to your system successfully.
```

function.(2.6) is selected as the antecedent, rather than aid.(1.5).

4.2 The Relative Contributions of the Salience Weighting Mechanisms

Using the test corpus of our blind test, we conducted experiments with modified versions of RAP, in which various elements of the salience weighting mechanism were switched off. We present the results in Table 4 and discuss their significance.

Ten variants are presented in Table 4; they are as follows:

- I "standard" RAP (as used in the blind test)
- II parallelism reward deactivated
- III non-adverbial and head emphasis deactivated
- IV matrix emphasis used instead of non-adverbial emphasis
- V cataphora penalty deactivated
- VI subject, existential, accusative, and indirect object/oblique complement emphasis (i.e., hierarchy of grammatical roles) deactivated
- VII equivalence classes deactivated
- VIII sentence recency and salience degradation deactivated
- IX all "structural" salience weighting deactivated (II + III + V + VI)
- X all salience weighting and degradation deactivated

The single most important element of the salience weighting mechanism is the recency preference (sentence recency factor and salience degradation; see VIII). This is not surprising, given the relative scarcity of intersentential anaphora in our test corpus (less than 20% of the pronoun occurrences had antecedents in the preceding sentence). Deactivating the equivalence class mechanism also led to a significant deterioration in RAP's performance; in this variant (VII), only the salience factors applying to a

particular NP contribute to its salience weight, without any contribution from other anaphorically linked NPs. The performance of the syntactic filter is degraded somewhat in this variant as well, since NPs that are anaphorically linked to an NP fulfilling the criteria for disjoint reference will no longer be rejected as antecedent candidates. The results for VII and VIII indicate that attentional state plays a significant role in pronominal anaphora resolution and that even a simple model of attentional state can be quite effective.

Deactivating the syntax-based elements of the salience weighting mechanism individually led to relatively small deteriorations in the overall success rate (II, III, IV, V, and VI). Eliminating the hierarchy of grammatical roles (VI), for example, led to a deterioration of less than 4%. Despite the comparatively small degradation in performance that resulted from turning off these elements individually, their combined effect is quite significant, as the results of IX show. This suggests that the syntactic salience factors operate in a complex and highly interdependent manner for anaphora resolution.

X relies solely on syntactic/morphological filtering and proximity to choose an antecedent. Note that the sentence pairs of the blind test set were selected so that, for each pronoun occurrence, at least two antecedent candidates remained after syntactic/morphological filtering (see Section 5.1). In the 17 cases in which X correctly disagreed with RAP, the proper antecedent happened to be the most proximate candidate.

We suspect that RAP's overall success rate can be improved (perhaps by 5% or more) by refining its measures of structural salience. Other measures of embeddedness, or perhaps of "distance" between anaphor and candidate measured in terms of clausal and NP boundaries, may be more effective than the current mechanisms for non-adverbial and head emphasis.[14] Empirical studies of patterns of pronominal anaphora in corpora (ideally in accurately and uniformly parsed corpora) could be helpful in defining the most effective measures of structural salience. One might use such studies to obtain statistical data for determining the reliability of each proposed measure as a predictor of the antecedent–anaphor relation and the orthogonality (independence) of all proposed measures.

5. Salience and Statistically Measured Lexical Preference

Dagan (1992) constructs a procedure, which he refers to as **RAPSTAT**, for using statistically measured lexical preference patterns to reevaluate RAP's salience rankings of antecedent candidates. RAPSTAT assigns a statistical score to each element of a candidate list that RAP generates; this score is intended to provide a measure (relative to a corpus) of the preference that lexical semantic/pragmatic factors impose upon the candidate as a possible antecedent for a given pronoun.[15]

14 Such a distance measure is reminiscent of Hobbs' (1978) tree search procedure. See Section 6.1 for a discussion of Hobbs' algorithm and its limitations.

The results for IV confirm our suspicions from the training phase that matrix emphasis (rewarding NPs not contained in a subordinate clause) does not contribute significantly to successful resolution.

15 Assume that P is a non-pleonastic and non-reflexive pronoun in a sentence such that RAP generates the non-empty list L of antecedent candidates for P. Let H be the lexical head (generally a verb or a noun) of which P is an argument or an adjunct in the sentence. RAPSTAT computes a statistical score for each element C_i of L, on the basis of the frequency, in a corpus, with which C_i occurs in the same grammatical relation with H as P occurs with H in the sentence. The statistical score that RAPSTAT assigns to C_i is intended to model the probability of the event where C_i stands in the relevant grammatical relation to H, given the occurrence of C_i (but taken independently of the other elements of L).

RAPSTAT reevaluates RAP's ranking of the elements of the antecedent candidate list L in a way that combines both the statistical scores and the salience values of the candidates. The elements of L appear in descending order of salience value. RAPSTAT processes L as follows. Initially, it considers the first two elements C_1 and C_2 of L. If (i) the difference in salience scores between C_1 and C_2 does not exceed a parametrically specified value (the salience difference threshold) and (ii) the statistical score of C_2 is significantly greater than that of C_1, then RAPSTAT will substitute the former for the latter as the currently preferred candidate. If conditions (i) and (ii) do not hold, RAPSTAT confirms RAP's selection of C_1 as the preferred antecedent. If these conditions do hold, then RAPSTAT selects C_2 as the currently preferred candidate and proceeds to compare it with the next element of L. It repeats this procedure for each successive pair of candidates in L until either (i) or (ii) fails or the list is completed. In either case, the last currently preferred candidate is selected as the antecedent.

An example of a case in which RAPSTAT overules RAP is the following.

```
The Send Message display is shown, allowing you to enter your
message and specify where it will be sent.
```

The two top candidates in the list that RAP generates for it.(1.17) are display.(1.4) with a salience value of 345 and message.(1.13), which has a salience value of 315. In the corpus that we used for testing RAPSTAT, the verb–object pair *send–display* appears only once, whereas *send–message* occurs 289 times. As a result, message receives a considerably higher statistical score than display. The salience difference threshold that we used for the test is 100, and conditions (i) and (ii) hold for these two candidates. The difference between the salience value of message and the third element of the candidate list is greater than 100. Therefore, RAPSTAT correctly selects message as the antecedent of it.

5.1 A Blind Test of RAP and RAPSTAT

Dagan et al. (in press) report a comparative blind test of RAP and RAPSTAT. To construct a database of grammatical relation counts for RAPSTAT, we applied the Slot Grammar parser to a corpus of 1.25 million words of text from 48 computer manuals. We automatically extracted all lexical tuples and recorded their frequencies in the parsed corpus. We then constructed a test set of pronouns by randomly selecting from the corpus sentences containing at least one non-pleonastic third person pronoun occurrence. For each such sentence in the set, we included the sentence that immediately precedes it in the text (when the preceding sentence does not contain a pronoun).[16] We filtered the test set so that for each pronoun occurrence in the set, (i) RAP generates a candidate list with at least two elements, (ii) the actual antecedent NP appears in the candidate list, and (iii) there is a total tuple frequency greater than 1 for the candidate

See Dagan 1992 and Dagan et al. (in press) for a discussion of this lexical statistical approach to ranking antecedent candidates and possible alternatives.

16 In the interests of simplicity and uniformity, we discarded sentence pairs in which the first sentence contains a pronoun. We decided to limit the text preceding the sentence containing the pronoun to one sentence because we found that in the manuals which we used to tune the algorithm, almost all cases of intersentential anaphora involved an antecedent in the immediately preceding sentence. Moreover, the progressive decline in the salience values of antecedent candidates in previous sentences ensures that a candidate appearing in a sentence which is more than one sentence prior to the current one will be selected only if no candidates exist in either the current or the preceding sentence. As such cases are relatively rare in the type of text we studied, we limited our test set to textual units containing the current and the preceding sentence.

list (in most cases, it was considerably larger).[17] The test set contains 345 sentence pairs with a total of 360 pronoun occurrences. The results of the blind test for RAP and RAPSTAT are as follows.[18]

	RAP	RAPSTAT
Total correct:	310 (86%)	319 (89%)
Total decided:	360 (100%)	182 (51%)
Correctly decided:	310 (86%)	144 (79%)
RAPSTAT		
Disagrees with RAP:		41 (22% of cases decided)
Correctly disagrees with RAP:		25 (61%)
Incorrectly disagrees with RAP:		16 (39%)
RAP/RAPSTAT		
Both wrong:		22 (12%)
Either RAP or RAPSTAT is correct:		335 (93%)

When we further analyzed the results of the blind test, we found that RAPSTAT's success depends in large part on its use of salience information. If RAPSTAT's statistically based lexical preference scores are used as the only criterion for selecting an antecedent, the statistical selection procedure disagrees with RAP in 151 out of 338 instances. RAP is correct in 120 (79%) of these cases and the statistical decision in 31 (21%) of the cases. When salience is factored into RAPSTAT's decision procedure, the rate of disagreement between RAP and RAPSTAT declines sharply, and RAPSTAT's performance slightly surpasses that of RAP, yielding the results that we obtained in the blind test.

In general, RAPSTAT is a conservative statistical extension of RAP. It permits statistically measured lexical preference to overturn salience-based decisions only in cases in which the difference between the salience values of two candidates is small and the statistical preference for the less salient candidate is comparatively large.[19] The comparative blind test indicates that incorporating statistical information on lexical preference patterns into a salience-based anaphora resolution procedure can yield a modest improvement in performance relative to a system that relies only on syntactic salience for antecedent selection. Our analysis of these results also shows that statistically measured lexical preference patterns alone provide a far less efficient basis for anaphora resolution than an algorithm based on syntactic and attentional measures of salience.[20]

6. Comparison with Other Approaches to Anaphora Resolution

We will briefly compare our algorithm with several other approaches to anaphora resolution that have been suggested.

17 In previous tests of RAP we found that it generates a candidate list that includes the correct antecedent of the pronoun in approximately 98% of the cases to which it applies.

18 We take RAPSTAT as deciding a case when it considers at least two candidates rather than deferring to RAP after the initial candidate because of a large salience difference between this candidate and the next one in the list. In cases in which RAPSTAT does not make an independent decision, it endorses RAP's selection. RAPSTAT's total success rate includes both sorts of cases.

19 John Justeson did the statistical analysis of the comparative blind test of RAP and RAPSTAT. These results are described in Dagan et al. (in press).

20 Dagan (1992) reaches a similar conclusion on the basis of a much smaller experiment.

6.1 Hobbs' Algorithm

Hobbs' (1978) algorithm relies on a simple tree search procedure formulated in terms of depth of embedding and left–right order. By contrast, RAP uses a multi-dimensional measure of salience that invokes a variety of syntactic properties specified in terms of the head–argument structures of Slot Grammar, as well as a model of attentional state.

Hobbs' tree search procedure selects the first candidate encountered by a left–right depth first search of the tree outside of a minimal path to the pronoun that satisfies certain configurational constraints. The algorithm chooses as the antecedent of a pronoun P the first NP_i in the tree obtained by left-to-right breadth-first traversal of the branches to the left of the path T such that (i) T is the path from the NP dominating P to the first NP or S dominating this NP, (ii) T contains an NP or S node N that contains the NP dominating P, and (iii) N does not contain NP_i. If an antecedent satisfying this condition is not found in the sentence containing P, the algorithm selects the first NP obtained by a left-to-right breadth first search of the surface structures of preceding sentences in the text.

We have implemented a version of Hobbs' algorithm for Slot Grammar. The original formulation of the algorithm encodes syntactic constraints on pronominal anaphora in the definition of the domain to which the search for an antecedent NP applies. In our implementation of the algorithm, we have factored out the search procedure and substituted RAP's syntactic–morphological filter for Hobbs' procedural filter. Let the Mods (modifiers) of a head H be the sisters of H in the Slot Grammar representation of the phrase that H heads. Our specification of Hobbs' algorithm for Slot Grammar is as follows:

1. Find a node N_1 such that (i) N_1 contains the pronoun P; (ii) N_1 is an S or NP; and (iii) it is not the case that there is a node $N_{1'}$ such that N_1 contains $N_{1'}$ and $N_{1'}$ satisfies (i) and (ii).

2. Check the list of Mods of N_1 left to right for NPs that are not elements of the list of pairs <P–NP> identified by the syntactic-morphological filter as noncoreferential and that occur to the left of P.

3. Select the leftmost NP in the filtered list of NP Mods of N_1.

4. If this list is nil, then repeat steps 2 and 3 recursively for each Mod in the list of Mods of N_1, each Mod in this second list of Mods, etc., until an NP antecedent is found.

5. If no NP antecedent is found by applying step 4, then identify a node N_2 that is the first NP/S containing N_1.

6. If N_2 is an NP and is not an element of the list of pairs <P–NP> identified by the filter, propose it as the antecedent.

7. Otherwise, apply steps 2–4 to N_2.

8. If no antecedent NP is found, continue to apply steps 5 and 6 and then steps 2–4 to progressively higher NP/S nodes.

9. If no antecedent NPs are found at the highest S of the sentence, then take N_1 to be the highest S node of the immediately preceding sentence and apply steps 2–4 to N_1.

Table 5
Results of blind test (Hobbs' algorithm)

	Total	Intersentential cases	Intrasentential cases
Number of pronoun occurrences	360	70	290
Number of cases that the algorithm resolves correctly	295 (82%)	61 (87%)	234 (81%)
Number of cases for which HOBBS correctly disagrees with RAP	22	9	13
Number of cases for which HOBBS incorrectly disagrees with RAP	38	4	34

We ran this version of Hobbs' algorithm on the test set that we used for the blind test of RAP and RAPSTAT; the results appear in Table 5.

It is important to note that the test set does not include pleonastic pronouns or lexical anaphors (reflexive or reciprocal pronouns), neither of which are dealt with by Hobbs' algorithm. Moreover, our Slot Grammar implementation of the algorithm gives it the full advantage of RAP's syntactic–morphological filter, which is more powerful than the configurational filter built into the original specification of the algorithm. Therefore, the test results provide a direct comparison of RAP's salience metric and Hobbs' search procedure.

Hobbs' algorithm was more successful than RAP in resolving intersentential anaphora (87% versus 74% correct).[21] Because intersentential anaphora is relatively rare in our corpus of computer manual texts and because RAP's success rate for intrasentential anaphora is higher than Hobbs' (89% versus 81%), RAP's overall success rate on the blind test set is 4% higher than that of our version of Hobbs' algorithm. This indicates that RAP's salience metric provides a more reliable basis for antecedent selection than Hobbs' search procedure for the text domain on which we tested both algorithms.

It is clear from the relatively high rate of agreement between RAP and Hobbs' algorithm on the test set (they agree in 83% of the cases) that there is a significant degree of convergence between salience as measured by RAP and the configurational prominence defined by Hobbs' search procedure. This is to be expected in English, in which grammatical roles are identified by means of phrase order. However, in languages in which grammatical roles are case marked and word order is relatively free, we expect that there will be greater divergence in the predictions of the two algorithms. The salience measures used by RAP have application to a wider class of languages than Hobbs' order-based search procedure. This procedure relies on a correspondence of grammatical roles and linear precedence relations that holds for a comparatively small class of languages.

6.2 Discourse Based Methods

Most of the work in this area seeks to formulate general principles of discourse structure and interpretation and to integrate methods of anaphora resolution into a computational model of discourse interpretation (and sometimes of generation as well). Sidner (1981, 1983), Grosz, Joshi, and Weinstein (1983, 1986), Grosz and Sidner (1986),

21 The difficulty that RAP encounters with such cases was discussed in Section 4.1. We are experimenting with refinements in RAP's scoring mechanism to improve its performance in these and other cases.

Brennan, Friedman, and Pollard (1987), and Webber (1988) present different versions of this approach. Dynamic properties of discourse, especially coherence and focusing, are invoked as the primary basis for identifying antecedence candidates; selecting a candidate as the antecedent of a pronoun in discourse involves additional constraints of a syntactic, semantic, and pragmatic nature.

In developing our algorithm, we have not attempted to consider elements of discourse structure beyond the simple model of attentional state realized by equivalence classes of discourse referents, salience degradation, and the sentence recency salience factor. The results of our experiments with computer manual texts (see Section 4.2) indicate that, at least for certain text domains, relatively simple models of discourse structure can be quite useful in pronominal anaphora resolution. We suspect that many aspects of discourse models discussed in the literature will remain computationally intractable for quite some time, at least for broad-coverage systems.

A more extensive treatment of discourse structure would no doubt improve the performance of a structurally based algorithm such as RAP. At the very least, formatting information concerning paragraph and section boundaries, list elements, etc., should be taken into account. A treatment of definite NP resolution would also presumably lead to more accurate resolution of pronominal anaphora, since it would improve the reliability of the salience weighting mechanism.

However, some current discourse-based approaches to anaphora resolution assign too dominant a role to coherence and focus in antecedent selection. As a result, they establish a strong preference for intersentential over intrasentential anaphora resolution. This is the case with the anaphora resolution algorithm described by Brennan, Friedman, and Pollard (1987). This algorithm is based on the centering approach to modeling attentional structure in discourse (Grosz, Joshi, and Weinstein 1983, 1986).[22] Constraints and rules for centering are applied by the algorithm as part of the selection procedure for identifying the antecedents of pronouns in a discourse. The algorithm strongly prefers intersentential antecedents that preserve the center or maximize continuity in center change, to intrasentential antecedents that cause radical center shifts. This strong preference for intersentential antecedents is inappropriate for at least some text domains—in our corpus of computer manual texts, for example, we estimate that less than 20% of referentially used third person pronouns have intersentential antecedents.[23]

There is a second difficulty with the Brennan et al. centering algorithm. It uses a hierarchy of grammatical roles quite similar to that of RAP, but this role hierarchy does not directly influence antecedent selection. Whereas the hierarchy in RAP contributes to a multi-dimensional measure of the relative salience of all antecedent candidates, in Brennan et al. 1987, it is used only to constrain the choice of the backward-looking center, *Cb*, of an utterance. It does *not* serve as a general preference measure for antecedence. The items in the forward center list, *Cf*, are ranked according to the hierarchy of grammatical roles. For an utterance U_n, $Cb(U_n)$ is required to be the highest ranked element of $Cf(U_{n-1})$ that is realized in U_n. If an element E in the list of possible

22 "A discourse segment consists of a sequence of utterances $U_1, \ldots, U_m$. With each utterance, U_n is associated with a list of *forward-looking centers*, $Cf(U_n)$, consisting of those discourse entities that are *directly realized* or *realized* by linguistic expressions in that utterance. Ranking of an entity on this list corresponds roughly to the likelihood that it will be the primary focus of subsequent discourse; the first entity on this list is the *preferred center*, $Cp(U_n)$. U_n actually centers, or is 'about,' only one entity at a time, the *backward-looking center*, $Cb(U_n)$. The backward center is a confirmation of an entity that has already been introduced into the discourse; more specifically, it must be realized in the immediately preceding utterance, U_{n-1}" (Brennan, Friedman, and Pollard 1987, p. 155).

23 This estimate is based on the small random sample used in our blind test (see Section 5.1).

forward centers, $Cf(U_{n-1})$, is identified as the antecedent of a pronoun in U_n, then E is realized in U_n. The Brennan et al. centering algorithm does *not* require that the highest ranked element of $Cf(U_{n-1})$ actually *be realized* in U_n, but only that $Cb(U_n)$ be the highest ranked element of $Cf(U_{n-1})$ which is, in fact, realized in U_n. Antecedent selection is constrained by rules that sustain cohesion in the relations between the backward centers of successive utterances in a discourse, but it is not determined directly by the role hierarchy used to rank the forward centers of a previous utterance. Therefore, an NP in U_{n-1} that is relatively low in the hierarchy of grammatical roles can serve as an antecedent of a pronoun in U_n, provided that no higher ranked NP in U_{n-1} is taken as the antecedent of some other pronoun or definite NP in U_n.[24] An example will serve to illustrate the problem with this approach.

```
The display shows you the status of all the printers.

It also provides options that control printers.
```

The (ranked) forward center list for the first sentence is as follows:

```
([DISPLAY] [STATUS] [YOU] [PRINTERS]).
```

Applying the filters and ranking mechanism of Brennan, Friedman, and Pollard (1987) yields two possible anchors.[25] Each anchor determines a choice of $Cb(U_n)$ and the antecedent of it. One anchor identifies both with display, whereas the second takes both to be status. The hierarchy of grammatical roles is *not* used to select display over status. Nothing in the algorithm rules out the choice of status as the backward center for the second sentence and as the antecedent of it. If this selection is made, display is not realized in the second sentence, and so $Cb(U_n)$ is status, which is then the highest ranked element of $Cf(U_{n-1})$ that is realized in U_n, as required by constraint 3 of the Brennan et al. centering algorithm.

In general, we agree with Alshawi (1987, p. 62) that an algorithm/model relying on the relative salience of *all* entities evoked by a text, with a mechanism for removing or filtering entities whose salience falls below a threshold, is preferable to models that "make assumptions about a single (if shifting) focus of attention."[26]

6.3 Mixed Models

This approach seeks to combine a variety of syntactic, semantic, and discourse factors into a multi-dimensional metric for ranking antecedent candidates. On this view, the score of a candidate is a composite of several distinct scoring procedures, each of which reflects the prominence of the candidate with respect to a specific type of information or property. The systems described by Asher and Wada (1988), Carbonell and Brown (1988), and Rich and LuperFoy (1988) are examples of this mixed evaluation strategy.

In general, these systems use composite scoring procedures that assign a global rank to an antecedent candidate on the basis of the scores that it receives from several

24 Other factors, such as level of embedding, may also be considered in generating an ordering for the list of forward-looking centers. Walker, Iida, and Cote (1990) discuss ordering conditions appropriate for Japanese.

25 An anchor is an association between a backward-looking center, *Cb*, and a list of forward-looking centers, *Cf*, for an utterance. An anchor establishes a link between a pronoun and its antecedent by associating the reference marker of the antecedent with that of the pronoun in the *Cf* list of the utterance.

26 See Walker (1989) for a comparison of the algorithm of Brennan, Friedman, and Pollard (1987) with that of Hobbs (1978) based on a hand simulation.

evaluation metrics. Each such metric scores the likelihood of the candidate relative to a distinct informational factor. Thus, for example, Rich and LuperFoy (1988) propose a system that computes the global preference value of a candidate from the scores provided by a set of constraint source modules, in which each module invokes different sorts of conditions for ranking the antecedent candidate. The set of modules includes (among others) syntactic and morphological filters for checking agreement and syntactic conditions on disjoint reference, a procedure for applying semantic selection restrictions to a verb and its arguments, a component that uses contextual and real-world knowledge, and modules that represent both the local and global focus of discourse. The global ranking of an antecedent candidate is a function of the scores that it receives from each of the constraint source modules.

Our algorithm also uses a mixed evaluation strategy. We have taken inspiration from the discussions of scoring procedures in the works cited above, but we have avoided constraint sources involving complex inferencing mechanisms and real-world knowledge, typically required for evaluating the semantic/pragmatic suitability of antecedent candidates or for determining details of discourse structure. In general, it seems to us that reliable large scale modelling of real-world and contextual factors is beyond the capabilities of current computational systems. Even constructing a comprehensive, computationally viable system of semantic selection restrictions and an associated type hierarchy for a natural language is an exceedingly difficult problem, which, to our knowledge, has yet to be solved. Moreover, our experiments with statistically based lexical preference information casts doubt on the efficacy of relatively inexpensive (and superficial) methods for capturing semantic and pragmatic factors for purposes of anaphora resolution. Our results suggest that scoring procedures which rely primarily on tractable syntactic and attentional (recency) properties can yield a broad coverage anaphora resolution system that achieves a good level of performance.

7. Conclusion

We have designed and implemented an algorithm for pronominal anaphora resolution that employs measures of discourse salience derived from syntactic structure and a simple dynamic model of attentional state. We have performed a blind test of this algorithm on a substantial set of cases taken from a corpus of computer manual text and found it to provide good coverage for this set. It scored higher than a version of Hobbs' algorithm that we implemented for Slot Grammar.

Results of experiments with the test corpus show that the syntax-based elements of our salience weighting mechanism contribute in a complexly interdependent way to the overall effectiveness of the algorithm. The results also support the view that attentional state plays a significant role in pronominal anaphora resolution and demonstrate that even a simple model of attentional state can be quite effective.

The addition of statistically measured lexical preferences to the range of factors that the algorithm considers only marginally improved its performance on the blind test set. Analysis of the results indicates that lexical preference information can be useful in cases in which the syntactic salience ranking does not provide a clear decision among the top candidates, and there is a strong lexical preference for one of the less salient candidates.

The relatively high success rate of the algorithm suggests the viability of a computational model of anaphora resolution in which the relative salience of an NP in discourse is determined, in large part, by structural factors. In this model, semantic and real-world knowledge conditions apply to the output of an algorithm that resolves pronominal anaphora on the basis of syntactic measures of salience, recency,

and frequency of mention. These conditions are invoked only in cases in which salience does not provide a clear-cut decision and/or there is substantial semantic–pragmatic support for one of the less salient candidates.[27]

Acknowledgments

We would like to thank Martin Chodorow, Ido Dagan, John Justeson, Slava Katz, Michael McCord, Hubert Lehman, Amnon Ribak, Ulrike Schwall, and Marilyn Walker for helpful discussion of many of the ideas and proposals presented here. The blind test and evaluation of RAPSTAT reported here was done jointly with Ido Dagan, John Justeson, and Amnon Ribak. An early version of this paper was presented at the Cognitive Science Colloquium of the University of Pennsylvania, in January 1992, and we are grateful to the participants of the colloquium for their reactions and suggestions. We are also grateful to several anonymous reviewers of *Computational Linguistics* for helpful comments on earlier drafts of the paper.

References

Alshawi, Hiyan (1987). *Memory and Context for Language Interpretation*. Cambridge: Cambridge University Press.

Asher, Nicholas, and Wada, Hajime (1988). "A computational account of syntactic, semantic and discourse principles for anaphora resolution." *Journal of Semantics* 6:309–344.

Bosch, Peter (1988). "Some good reasons for shallow pronoun processing." In *Proceedings, IBM Conference on Natural Language Processing*. New York: Thornwood.

Brennan, Susan; Friedman, Marilyn; and Pollard, Carl (1987). "A centering approach to pronouns." In *Proceedings, 25th Annual Meeting of the Association for Computational Linguistics*, 155–162.

Carbonell, Jaime, and Brown, Ralf (1988). "Anaphora resolution: A multi-strategy approach." In *Proceedings, 12th International Conference on Computational Linguistics*, 96–101.

Dagan, Ido (1992). "Multilingual statistical approaches for natural language disambiguation" (in Hebrew). Doctoral dissertation, Israel Institute of Technology, Haifa, Israel.

Dagan, Ido; Justeson, John; Lappin, Shalom; Leass, Herbert; and Ribak, Amnon (in press). "Syntax and lexical statistics in anaphora resolution." *Applied Artificial Intelligence*.

Grosz, Barbara; Joshi, Aravind; and Weinstein, Scott (1983). "Providing a unified account of definite noun phrases in discourse." In *Proceedings, 21st Annual Meeting of the Association of Computational Linguistics*, 44–50.

Grosz, Barbara; Joshi, Aravind; and Weinstein, Scott (1986, unpublished). "Towards a computational theory of discourse interpretation." Harvard University and University of Pennsylvania.

Grosz, Barbara, and Sidner, Candice (1986). "Attention, intentions, and the structure of discourse." *Computational Linguistics* 12:175–204.

Guenthner, Franz, and Lehmann, Hubert (1983). "Rules for pronominalization." In *Proceedings, First Annual Meeting of the European Chapter of the ACL*, 144–151.

Hobbs, Jerry (1978). "Resolving pronoun references." *Lingua* 44:311–338.

Johnson, David (1977). "On relational constraints on grammars." In *Syntax and Semantics 8*, edited by P. Cole and J. Sadock, 151–178. New York: Academic Press.

Kamp, Hans (1981). "A theory of truth and semantic representation." In *Formal Methods in the Study of Language*, edited by J. Groenendijk, T. Janssen, and M. Stokhof. Amsterdam: Mathematisch Centrum Tracts.

Keenan, Edward, and Comrie, Bernard (1977). "Noun phrase accessibility and universal grammar." *Linguistic Inquiry* 8:62–100.

Lappin, Shalom (1985). "Pronominal binding and coreference." *Theoretical Linguistics* 12:241–263.

Lappin, Shalom, and McCord, Michael (1990a). "A syntactic filter on pronominal anaphora in slot grammar." In *Proceedings, 28th Annual Meeting of the Association for Computational Linguistics*, 135–142.

Lappin, Shalom, and McCord, Michael (1990b). "Anaphora resolution in slot grammar." *Computational Linguistics* 16:197–212.

27 Bosch (1988) suggests a psychological processing model in which hearers rely on first pass syntactically based strategies for initial linking of pronouns to antecedent NPs.

Leass, Herbert, and Schwall, Ulrike (1991). "An anaphora resolution procedure for machine translation." IWBS Report 172, IBM Germany Scientific Center, Heidelberg, Germany.

McCord, Michael (1989a). "Design of **LMT**: A prolog-based machine translation system." *Computational Linguistics* 15:33–52.

McCord, Michael (1989b). "A new version of the machine translation system **LMT**." *Literary and Linguistic Computing* 4:218–229.

McCord, Michael (1990). "Slot grammar: A system for simpler construction of practical natural language grammars." In *Natural Language and Logic: International Scientific Symposium*, edited by R. Studer, 118–145. Lecture Notes in Computer Science, Berlin: Springer Verlag.

McCord, Michael (1993). "Heuristics for broad-coverage natural language parsing." In *Proceedings, ARPA Human Language Technology Workshop*, University of Pennsylvania.

McCord, Michael (in press). "The slot grammar system." In *Unification in Grammar*, edited by Jürgen Wedekin and Christian Rohrer (also IBM Research Report RC 17313). Cambridge, MA: MIT Press.

McCord, Michael; Bernth, Arendse; Lappin, Shalom; and Zadrozny, Wlodek (1992). "Natural language processing within a slot grammar framework." *International Journal on Artificial Intelligence Tools* 1:229–277.

Rich, Elaine, and LuperFoy, Susann (1988). "An architecture for anaphora resolution." In *Proccedings, ACL Conference on Applied Natural Language Processing*, 18–24.

Sidner, Candice (1981). "Focusing for interpretation of pronouns." *American Journal of Computational Linguistics* 7:217–231.

Sidner, Candice (1983). "Focusing in the comprehension of definite anaphora." In *Computational Models of Discourse*, edited by Michael Brady and Robert Berwick, 267–330. Cambridge, MA: MIT Press.

Walker, Marilyn (1989). "Evaluating discourse processing algorithms." In *Proceedings, 27th Annual Meeting of the Association for Computational Linguistics*, 251–261.

Walker, Marilyn; Iida, Masayo; and Cote, Sharon (1990). "Centering in Japanese discourse." In *Proceedings, 13th International Conference on Computational Linguistics*, 1–6.

Webber, Bonnie (1988). "Discourse Deixis: Reference to discourse segments." In *Proceedings, 26th Annual Meeting of the Association for Computational Linguistics*, 113–121.

Williams, Edwin (1984). "Grammatical relations." *Linguistic Inquiry* 15:639–673.