Factors in anaphora resolution: they are not the only things that matter. A case study based on two different approaches

Ruslan Mitkov
School of Languages and European Studies
University of Wolverhampton
Stafford Street
Wolverhampton WV1 1SB
United Kingdom
R.Mitkov@wlv.ac.uk

ABSTRACT

The paper discusses the significance of factors in anaphora resolution and on the basis of a comparative study argues that what matters is not only a good set of reliable factors but also the strategy for their application. The objective of the study was to find out how well the *same* set of factors worked within two *different* computational strategies. To this end, we tuned two anaphora resolution approaches to use the same core set of factors. The first approach uses constraints to discount implausible candidates and then consults preferences to rank order the most likely candidate. The second employs only preferences and does not discard any candidate but assumes initially that the candidate examined is the antecedent; on the basis of uncertainty reasoning formula this hypothesis is either rejected or accepted.

The last section of the paper addresses some related unresolved issues which need further research.

1. Approaches and factors in anaphora resolution

Approaches to anaphora resolution usually rely on a set of "anaphora resolution factors". Factors used frequently in the resolution process include gender and number agreement, c-command constraints, semantic consistency, syntactic parallelism, semantic parallelism, salience, proximity etc. These factors can be "eliminating" i.e. discounting certain noun phrases from the set of possible candidates (such as gender and number constraints, c-command constraints, semantic consistency) or "preferential", giving more preference to certain candidates and less to others (such as parallelism, salience). Computational linguistics literature uses diverse terminology for these - for example E. Rich and S. LuperFoy ([Rich & LuperFoy 88]) refer to the "eliminating" factors as "constraints", and to the preferential ones as "proposers", whereas Carbonell and Brown ([Carbonell & Brown 88] use the terms "constraints" and "preferences". Other authors argue that all factors should be regarded as preferential, giving higher preference to more restrictive factors and lower - to less "absolute" ones, calling them simply "factors" ([Preuß et al. 94]), "attributes" ([Pérez 94]) or "symptoms" ([Mitkov 95]).

The impact of different factors and/or their coordination have already been described in the literature (e.g. [Carter 90], [Dagan et al. 91]). In his work David Carter argues that a flexible control structure based on numerical scores assigned to preferences allows greater co-operation between factors as opposed to a more limited depth-first architecture. His discussion is grounded in comparisons between two different implemented systems - SPAR ([Carter 87]) and the SRI Core Language Engine ([Alshawi 90]). I. Dagan, J. Justeson, Sh. Lappin, H. Leass and A. Ribak ([Dagan et al. 91] attempt to determine the relative importance of distinct informational factors by comparing a syntactically-based salience algorithm for pronominal anaphora resolution (RAP) ([Lappin & Leass 94]) with a procedure for reevaluating the decisions of the algorithm on the basis of statistically modelled lexical semantic/pragmatic preferences ([Dagan 92]). Their results suggest that syntactically measured salience preferences are dominant in anaphora resolution.

While a number of approaches use a similar set of factors, the "computational strategies" for the application of these factors may differ (by "computational strategy" we mean here the way antecedents are computed, tracked down, i.e. the algorithm, formula for assigning antecedents and not computational issues related to programming languages, complexity etc.). Some approaches incorporate a traditional model which discounts unlikely candidates until a minimal set of plausible candidates is obtained (then make use of center or focus, for instance), whereas others compute the most likely candidate on the basis of statistical or AI techniques/models. This observation led us to term the approaches to anaphora resolution "traditional knowledge-based" and "alternative"

([Mitkov 96]). In the experiment[1] described below, we have kept the set of factors constant and sought to explore which of two approaches, different in terms of "computational strategy" ([Mitkov 94a], [Mitkov 95]) was the more successful.

In the first of the two approaches, constraints rule out impossible candidates and those left are further evaluated according to various preferences and heuristics but above all the "opinion" of the discourse module which strongly suggests the center of the previous clause as the most likely antecedent. The second approach regards all candidates as equal to start with and seeks to collect evidence about how plausible each candidate is on the basis of the presence/absence of certain symptoms (influence/non-influence of certain factors). All factors (symptoms) are unconditional preferences (i.e. there are no "absolute", "ruling out" factors) and are assigned numerical values. Candidates are proposed or rejected as antecedents by an uncertainty reasoning hypothesis verification formula. From the results obtained, we shall see that some of our conclusions coincide with Carter's. Further, we shall see that to achieve improved performance, a compromise, two-engine approach incorporating both strategies is an even better option.

The results of this study have an implication for building a practical anaphora resolution system: what matters is not only the careful selection of factors, but also the choice of approach (e.g. traditional or statistic, AI etc.) or combination of approaches.

2. Comparing two different approaches using the same factors

Before discussing the results of our comparative study, we shall briefly outline the approaches which served as a basis for the experiment.

2.1 The integrated anaphora resolution approach ([Mitkov 94a])

The Integrated Approach (IA) relies on both constraints and preferences, with constraints discounting implausible candidates, and preferences working towards the selection of the most likely antecedent. The IA is built on modules consisting of different types of rule-based knowledge - syntactic, semantic, domain, discourse and heuristic ([Mitkov 94a]).

The syntactic module, for example, knows that the anaphor and antecedent must agree in number, gender and person. It checks whether the c-command constraints hold and establishes disjoint reference. In cases of syntactic parallelism, it prefers the noun phrase with the same syntactic role as the anaphor as the most probable antecedent. It knows when cataphora is possible and can indicate syntactically topicalised noun phrases, which are more likely to be antecedents than non-topicalised ones.

The semantic module checks for semantic consistency between the anaphor and the possible antecedent. It filters out semantically incompatible candidates following verb semantics or animacy of the candidate. In cases of semantic parallelism, it prefers the noun phrase which has the same semantic role as the anaphor as the most likely antecedent.

The syntactic and semantic modules are enhanced by a discourse module which plays a very important role because it keeps a track of the centers of each discourse segment (it is the center which is, in most cases, the most probable candidate for an antecedent). Based on empirical studies from the sublanguage of computer science, we have developed a statistical approach to determine the probability of a noun (verb) phrase to be the center of a sentence. Unlike other approaches known to us, our method is able to propose the center with a high probability in every discourse sentence, including the first. The approach uses an inference engine based on Bayes' formula which draws an inference in the light of some new piece of evidence. This formula calculates the new probability, given the old probability plus some new piece of evidence ([Mitkov 94b]).

The domain knowledge module is a small knowledge base of the concepts of the domain considered, while the heuristic knowledge module is a set of useful rules (e.g. the antecedent is likely to be located in the current sentence or in the previous one) which can forestall certain impractical search procedures.

The referential expression filter plays an important role in filtering out expressions where 'it' is not anaphoric (e.g. "it is important", "it is necessary", "it should be pointed out" etc.).

The IA operates as follows. Syntax and semantic constraints (agreement, configurational, semantic consistency) reduce the set of all candidates to the set of possible ones. If the latter consists of more than one noun phrase, then preferences are activated. Highest preference (score) is given to noun phrases which are the center of the previous clause, but syntactic parallelism, semantic parallelism and referential distance also contribute (though less significantly) to the overall score.

2.2 The uncertainty reasoning approach ([Mitkov 95]).

The Uncertainty Reasoning Approach (URA) uses AI uncertainty reasoning techniques. Uncertainty reasoning was selected as an alternative because:

[1]The idea for this study was suggested by Allan Ramsey

- In Natural Language Understanding, the program is likely to estimate the antecedent of an anaphor on the basis of *incomplete information*: even if information about constraints and preferences is available, one can to assume that a Natural Language Understanding program is not able to understand the input completely ;
- The necessary initial constraint and preference scores are determined by humans; therefore the scores are originally subjective and should be regarded as *uncertain facts*.

The uncertainty reasoning approach makes use of "standard" anaphor resolution "symptoms" such as agreement, c-command constraints, parallelism, topicalisation, verb-case roles, but also of further symptoms based on empirical evidence, such as subject preference, domain concept preference, object preference, section head preference, reiteration preference, definiteness preference, main clause preference etc. Note that this strategy does not regard factors as absolute constraints; all symptoms are in practice preferences with numerical values assigned.

More specifically, the presence/absence of a certain symptom corresponds to an appropriate score - certainty factor (CF) which is attached to it. For instance, the presence of a certain symptom s assigns $CF_{s_{pr}}$ ($0<CF_{s_{pr}}<1$), whereas the absence corresponds to $CF_{s_{ab}}$ ($-1<CF_{s_{ab}}\leq 0$). For easier reference and brevity, we associate with the symptom s only the certainty factor CF_S which we regard as a two-value function ($CF_S \in \{CF_{s_{pr}}, CF_{s_{ab}}\}$).

The antecedent searching procedure employs an uncertainty reasoning strategy: the search for an antecedent is regarded as an affirmation (or rejection) of the hypothesis that a certain noun phrase is the correct antecedent. The certainty factor (CF) serves as a quantitative approximation of the hypothesis. The presence/absence of each symptom s causes recalculation (increase or decrease) of the global hypothesis certainty factor CF_{hyp} until: $CF_{hyp} > CF_{threshold}$ for affirmation or $CF_{hyp} < CF_{min}$ for rejection of the hypothesis. Hypothesis verification operates from right to left: first the closest to the anaphor noun phrase is tried. If this noun phrase does not survive the hypothesis of being the antecedent, the next rightmost is tried and so on.

We use a hypothesis verification formula for recalculation of the hypothesis on the basis of presence (in our case also of absence) of certain symptoms. Our formula is a modified version of van Melle's formula in ([Buchanan & Shortliffe 84]).

$$CF_{hyp}(s, CF_{old}) =$$
$$CF_s + CF_{old} - CF_s * CF_{old} \Leftrightarrow CF_s > 0,\ CF_{old} > 0 \text{ or}$$
$$(CF_s + CF_{old})/[1-\min(|CF_s|, |CF_{old}|)] \Leftrightarrow CF_s > 0,$$
$$CF_{old} < 0 \text{ or } CF_s > 0,\ CF_{old} < 0 \text{ or}$$
$$-CF_{hyp}(s, CF_{old}) \Leftrightarrow CF_s < 0,\ CF_{old} < 0$$

where $CF_{hyp}(s, CF_{old})$ is the hypothesis certainty factor, contributed by the presence/absence of symptom s and the current (old) hypothesis certainty factor CF_{old}. As an illustration, suppose a certain NP has reached a CF=0.5 after testing the presence of some symptoms (e.g. syntactic agreement) and that the symptom s with CF=0.45 holds. Then $CF_{hyp}(s, CF_{old}) = 0.5+0.45-0.5*0.45=0.725$

2.3. The same set of factors but different computational strategies

The objective of the study was to compare the IA and the URA with both using the same repertoire of factors to see what was the impact of the different computational strategies.

2.3.1 *Factors used*

We used the same set of factors in both approaches - the factors selected were deemed to be a "core set" from the point of view of both approaches. The factors used in our experiment were:

- Gender agreement
- Number agreement

  Anaphors and their antecedents must agree in number and gender.

- Syntactic parallelism
  Preference is given to antecedents with the same syntactic function as the anaphor.

The programmer$_i$ combined successfully Prolog$_j$ with C, but he$_i$ had combined it$_j$ with Pascal last time.
The programmer$_i$ combined successfully Prolog with C$_j$, but he$_i$ had combined Pascal with it$_j$ last time.

- Topicalisation
  Topicalised structures are given preferential treatment as possible antecedents.

It was Ruslan$_i$ who convinced me to go to Madrid. Why did he$_i$ do it?

- Semantic consistency
  If satisfied by the anaphor, semantic consistency constraints must be satisfied also by its antecedent.

Vincent removed the diskette from the computer$_i$ and then disconnected it$_i$.
Vincent removed the diskette$_i$ from the computer and then copied it$_i$.

- Semantic parallelism
  Those antecedents are favoured which have the same semantic role as the anaphor.

Vincent gave the diskette to Sody$_i$. Kim also gave him$_i$ a letter.
Vincent$_i$ gave the diskette to Sody. He$_i$ also gave Kim a letter.

- Subjects
  From the list of potential candidates the subject of the previous sentence (clause) is the preferred antecedent; the second preferred antecedent is the direct object.

- Domain concepts
  The NP representing a domain concept is preferred to NPs which are not domain concepts.

The last two preferences can be illustrated by the example:

When the Prolog system$_i$ finds a solution to a query, it$_i$ will print the values given to variables used in the query.

- Object preference indicated by verbs
  If the verb is a member of the Verb_set = {discuss, present, illustrate, summarise, examine, describe, define, show, check, develop, review, report, outline, consider, investigate, explore, assess, analyse, synthesise, study, survey, deal, cover}, then consider the object as the preferred antecedent.

- Object preference indicated by nouns
  If the subject is "chapter", "section", "table", "document", "paper" etc. or a personal pronoun "I", "we", "you", then consider the object as the preferred antecedent.

This table shows a minimal configuration$_i$; it$_i$ does not leave much room for additional applications or other software for which you may require additional swap space.

- Repetition
  Repeated NPs are considered to be the preferred candidate for antecedent.

- Heading
  If an NP occurs in the head of the section, part of which is the current sentence, then consider it as the candidate likeliest to be the antecedent.

System programs

System programs$_i$ such as the supervisor and the language translator should not have to be translated every time they$_i$ are used, otherwise this would result in a serious increase in the time spent in processing a user's program. System programs$_i$ are usually written in the assembly version of the machine language and are translated once into the machine code itself. From then on they$_i$ can be loaded into memory in machine code without the need for any intermediate translation phase.

- Distance
  Candidates from the previous clause or sentence are preferred.

The objective of our study was to use the *same* set of factors. As listed above, number and gender agreement, as well as semantic consistency, were used as constraints by the IA whereas the remaining were used as preferences; the URA used all factors as preferences. Note also that the factors "subject", "repetition", "head", "verb", "object" and "distance" were used as "anaphora resolution symptoms" (preferences) in the URA, whereas they played the role of center tracking preferences in the IA. In both approaches these factors were duly "consulted" and taken into consideration .

2.3.2 *Evaluation*

The evaluation was conducted on the basis of a manually annotated test corpus from the sublanguage of Computer Science. We selected 133 paragraphs containing the anaphor "it" (altogether 512 occurrences of "it") and tested both approaches tuned to activate only the core set of factors described.

Our preliminary results showed a success rate of 83% for the IR as opposed to 82% for the URA with $CF_{threshold}$ 0.7. Out of the 17% uncorrectly solved anaphors by the IR, 5% were solved correctly by the URA. Out of the 18% uncorrectly solved anaphors by the URA, 4% were solved correctly by the IR. With a higher threshold of 0.8, however, the URA went down to a level of accuracy of 71%. The lower success rates (as compared to [Mitkov 95]) are due to the fact that both approaches are restricted to the "core set of factors" and thus cannot draw on others which they would normally have at their disposal (e.g. c-command constraints were not included in the experimental core set). In particular, when the number of symptoms is reduced, the URA cannot benefit from all its sources of evidence and thus high thresholds cannot realistically be reached.

2.3.3 *Discussion of the results*

In terms of performance it looks like the IA has a slight edge over the URA. However, such a suggestion may be misleading because it turned out that the URA was in general "safer". Our study prompts the following conclusions.

(i) In most cases both approaches were correct

This applies to the majority of cases. One example is:

> Installing the battery in your Portable StyleWriter
>
> In most cases a Print dialogue box appears, with options for printing your document. The dialogue box_i may not look exactly like the window shown here, but it_i will have the options shown in this one.

The IR concludes that "dialogue box" is the antecedent mainly on the basis of assigning "dialogue box" as center of the preceding clause. It is evident that syntactic or semantic constraints cannot be very helpful here. The URA reaches confidence factor 0.9227 which is sufficient for accepting the hypothesis.

(ii) When information is insufficient, the URA is less "decisive"

As an illustration, consider the test text:

> Why C++ is better than C?
>
> Because $C++_i$ is based on C, it_i retains much of that language, including a rich operator set, nearly orthogonal design, terseness and extendibility.

The URA reaches confidence factor 0.893367. It cannot arrive at a confidence factor above 0.9 because the number of indicative symptoms is insufficient. In this case, the URA works towards the hypothesis on the basis of the following symptoms only: number, gender, semantic consistency, syntactic parallelism, subjecthood.

Our evaluation also showed that

(iii) The IA is more decisive but could be "iffy"
(iv) When information is ample, the URA is more "confident"
(v) The URA is better in cases of gender and number discrepancy between anaphor and antecedent

Because the IA followed the traditional rule that anaphor and antecedent must agree in gender and number, its initial version did not capture a number of exceptions (e.g. in the case of collective nouns) where the anaphor may be plural and the antecedent singular or vice versa.

> Computer memory, also known as primary storage, is closely associated with the central processing $unit_i$ but not actually part of it_i. Memory holds the $data_k$ after it_k is input to the system but before it_k is processed.

One way of coping with such "irregularities" is to draw up a comprehensive list of all such discrepancies. However, it would be more natural to use a preferences-only approach which assigns preferences and in which ruling out on the basis of non-agreement can be overturned by the joint influence of other preferences.

(vi) The IA is better in cases where "it" occurs frequently and refers to different antecedents

> The Central Processing Unit and Memory: Data Manipulation
>
> Computer memory, also known as primary storage, is closely associated with the central processing $unit_i$ but not actually part of it_i. $Memory_k$ holds the $data_j$ after it_j is input to the system but before it_j is processed. It_k also holds the data after it_j has been processed but before it_j has been released to the output device.

(vii) Both approaches lose on performance because of the lack of corpus-based collocation information

Neither approach relies on collocation patterns which is seen as a disadvantage in cases where syntactical and semantic constraints/preferences are not able to discriminate between more than one candidate.

These results inspired us to venture towards a two-engine strategy which would combine the benefits of the "speed" of the IR and the "safety" of the URA.

2.4 The two-engine strategy

Two engines are better than one: a combined strategy which incorporates the advantages of each of these approaches, generates more power and confidence in the search for the antecedent.

The two-engine strategy evaluates each candidate for anaphor from the point of view of both the IA and the URA. If opinions coincide, the evaluating process is stopped earlier than would be the case if only one engine were acting. This also makes the searching process shorter: our tests show that the integrated approach engine needs about 90% of the search it would make when operating on its own; similarly, the uncertainty reasoning engine does only 67% of the search it would do when operating as a separate system. In

addition, the results from using both approaches are more accurate (see the figure below).

This combined strategy enables the system to consider all the symptoms in a consistent way; it does not regard any symptom as absolute or unconditional. This "attitude" is particularly appropriate for symptoms like 'gender' or 'number' (which could be regarded as absolute in some languages but 'conditional' in other)[2].

Additional reasons for selecting a two-engine approach are the following:

- two independent judgements, if confirmed, lend more credibility to the selected antecedent
- using two approaches means complementarity: e.g. the "conditionality" of gender is better captured by uncertainty reasoning; in addition, in sentences with more than one pronoun, center tracking alone (and therefore the integrated approach) is not very helpful for determining the corresponding antecedents
- though the URA may be considered more stable in such situations, it is comparatively slow: if intermediate results obtained by both engines are reported to be close, it could adopt a lower hypothesis threshold (thus speeding up the decision process)

We have implemented the two-engine model as a program and the following table shows its success rate. Five documents served as inputs, each text taken from a computer science book. The documents ranged from 3000 to 5000 words and were estimated to contain a comparatively high number of pronouns (it was not always easy to find texts abundant in pronominal anaphors). These documents were different from the corpus initially used for the development of various 'resolution rules' and were hand-annotated (syntactic and semantic roles). Other versions of these documents, which contained anaphoric references marked by a human expert, were used as an evaluation corpus.

We tested on these inputs (i) the integrated approach, (ii) the uncertainty reasoning approach and (iii) the two-engine approach. Note that the two-engine version did not work on a "core set" of factors only, but benefited from the full range of "constraints" and "preferences" used by the IA and the complete list of "symptoms" utilised by the URA. The results show an improvement when the IA and the URA were combined into a two-engine strategy:

	Integrated approach	Uncertainty reasoning	Two-engine strategy
Document 1	89,1	87,3	91,7
Document 2	90,6	91,6	93,1
Document 3	91,7	90,4	93,8
Document 4	85,9	83,3	88,4
Document 5	88,6	89,2	93,7

3. Factors in anaphora resolution: further issues that need attention

In this section we address four questions that remain unresolved or debatable: (i) how dependent are factors? (ii) are preferences better than constraints? (iii) do factors hold good for all genres? and (iv) which is the best order to apply the factors?

3.1 Dependence and mutual dependence of factors

While it is vital to arrive at a comprehensive list of contributory factors (or a core set of maximally contributory factors), it is worthwhile to consider not only the impact of each factor on the resolution process but also the impact of factors on other factors. A key issue which needs further attention is the "(mutual) dependence" of factors.

In order to clarify the notion of (mutual) dependence, it would be helpful to view the "factors" as "symptoms", "indicators" i.e. as "present" or "absent" in a certain discourse situation. For instance, if "gender agreement" holds between a candidate for an anaphor and the anaphor itself, we say that the symptom or indicator "gender agreement" is present. Similarly, if the candidate is in a subject position, we say that the symptom "subjecthood" is present.

We define dependence/mutual dependence of factors in the following way. Given the factors x and y, we say that factor y is dependent on factor x to the extent that the presence of x implies y. Two factors will be termed mutually dependent if each depends on the other.

The phenomenon of (mutual) dependence has not yet been fully investigated, but we feel that it can play an important role in the process of anaphora resolution, especially in algorithms based on the ranking of preferences. Information on the degree of dependence would be especially welcome in a comprehensive probabilistic model and will undoubtedly lead to more precise results.

Our preliminary (and insufficient) observations suggest that there are more preferences which are dependent, than there are constraints. The preferences "object preference indicated by verbs" and "object preference indicated by nouns" (see sec-

[2] Often in English singular pronouns (e.g. some singular pronouns denoting a collective notion) may be referred to by plural pronoun and vice-versa; In German, there is no absolute gender agreement: "Mädchen" (girl) is neuter, but one can refer to "Mädchen" by a female pronoun (sie).

tion 2.3.1) are a good example of mutual dependence. Indeed, I had difficulties finding a discourse situation in which those two factors did not occur together. In a simple scoring formula it might be wiser to take only one of them into account; in a more sophisticated probabilistic model what we need is sufficient empirical evidence on the degree of this dependence in order to incorporate it in the model. In addition, the preference "lexical reiteration" is dependent (though to a lower degree) on the preference "section heading" (this dependence does not seem to hold in the reverse direction, so these two factors are not mutually dependent). Finally, it seems that "syntactic parallelism" and "semantic parallelism" are not completely independent.

As far as constraints are concerned, those that we looked at (gender and number agreement, c-command constraints, semantic consistency), do not appear to be dependent at least for English.

We have attempted to correct the mutual dependence between "object preference indicated by verbs" and "object preference indicated by nouns" by giving the latter symptom a lower numerical value. However, more exact data on the degree of dependence are needed and have to be captured in an appropriate probabilistic model. An investigation into the (mutual) dependence of factors on the basis of large annotated corpora is one of our priority research objectives.

A simple, safe alternative would be to use a core set restricted to "independent factors" but this would mean a compromise on performance since the benefit from some additional (though not independent) factors would be lost.

3.2 What is better: preferences or constraints?

This is a another question which does not have an unambiguous answer. Preferences may be safer in that they, as opposed to constraints, may not rule out a situation not modelled by the resolution engine. On the other hand, as shown in our experiment, constraints could make the procedure faster and more accurate.

3.3. Do the factors hold good for all genres?

Perhaps we can speak of less "general" factors and more "general" factors. The factors "object preference indicated by verbs", "object preference indicated by nouns" and "section heading" appear to be more "genre specific". Their role, however, should not be underestimated - we have found them very useful in the textbook genre which spans way beyond the sublanguage of Computer Science. In our experiments, we gave the factor "object preference indicated verbs" highly preferential treatment. As an illustration, the RAP algorithm has been reported ([Dagan et al 91] as having difficulty in identifying the antecedent of "it" in the sentence

> The utility (CDVU) shows you a LIST4250, LIST38PP, or LIST3820 $file_i$ on your terminal for a format similar to that in which it_i will be printed.

where it pointed out "utility" as the most salient candidate for the anaphor "it". Both IA and URA, however, would correctly identify "file" as the antecedent because the "object preference indicated by verbs" (and "object preference indicated by nouns") factors would regard "file" as highly salient and would considerably raise its score.

3.4 Order of constraints and priority of preferences

Does order of constraints matter? Since "absolute" constraints have to be met, not complying with any of them means discounting candidates. Therefore, in our opinion, the order in which the constraints are applied does not matter.

In a system which incorporates both constraints and preferences, it would be natural to start with constraints and then to switch to preferences. We fear, however, that unless we have a comprehensive list of exceptions, simply discounting candidates on the basis of gender and number agreement in English could be risky (we are referring to the number of cases where collective nouns may be referred to by plural pronouns[3] and cases where plural nouns may be referred to by a singular pronoun[4]). Therefore, unless we have such a comprehensive list, our personal inclination would be to rely on a preferences-based architecture.

As far as preferences are concerned, it would be natural to start with the more "contributory" factors in terms of numerical value. In our experiments so far we have tried both descending (starting first the higher value factors) and ascending orders of application. We did not find any essential difference in the final result. However, the searching process in the second option was, as expected, longer.

4. Conclusion

The paper demonstrates, on the basis of a comparative study, that an anaphora resolution system needs not only a good set of contributory factors but also a clear strategy for their application. The results of the study have implications for the development of anaphora resolution systems, sug-

[3]For instance, nouns such as "government", "parliament", "police" "team" etc. are usually referred to by "they

[4]See section 2.3.3, the examples which follow conclusions (v) and (vi)

gesting careful selection of both factors and computational strategies, or combination of them.

Acknowledgements

Many thanks to Allan Ramsey for suggesting the idea of the comparative study. I am also indebted to Chris Paice and to the 3 referees for their useful comments.

References

[Alshawi 90] H. Alshawi - *Resolving quasi logical forms.* Computational Linguistics, 16:3, 1990

[Buchanan & Shortliffe 84] B. Buchanan, Ed. Shortliffe - *Rule-based expert systems.* Addison-Wesley, 1984

[Carbonell & Brown 88] J. Carbonell, R. Brown - *Anaphora resolution: a multi-strategy approach.* Proceedings of the 12. International Conference on Computational Linguistics COLING'88, Budapest, August, 1988

[Carter 87] D. Carter - *Interpreting anaphora in natural language texts.* Chichester: Ellis Horwood, 1987

[Carter 90] David M. Carter - *Control issues in anaphor resolution.* Journal of Semantics, 7, 1990

[Dagan 92] I. Dagan - *Multilingual statistical approaches for natural language disambiguation* (in Hebrew). PhD dissertation. Technion-Israel Institute of Technology, Haifa

[Dagan et al. 91] Ido Dagan, John Justeson, Shalom Lappin, Hergert Leass and Amnon Ribak - *Syntax and lexical statistics in anaphora resolution.* Applied Artificial Intelligence, 9, 1995

[Lappin & Leass 94] Sh. Lappin, H. Leass - *An algorithm for pronominal anaphora resolution. Computational Linguistics*, 20(4), 1994

[Mitkov 94a] Mitkov R. - *An integrated model for anaphora resolution.* Proceedings of the 15th International Conference on Computational Linguistics COLING'94, Kyoto, Japan, 5-9 August 1994

[Mitkov 94b] Mitkov R. - *A new approach for tracking center.* In Proceedings of the International Conference "New Methods in Language Processing", UMIST, Manchester, UK, 13-16 September 1994

[Mitkov 95] R. Mitkov - *An uncertainy reasoning approach to anaphora resolution.* Proceedings of the Natural Language Pacific Rim Symposium, 4-7 December 1995, Seoul, Korea

[Mitkov 96] Mitkov R. - *Anaphor resolution in Natural Language Processing and Machine Translation.* Proceedings of the International Colloquium on Discourse Anaphora and Anaphora Resolution. Lancaster, 17-19 July 1996 (keynote speech)

[Rico Pérez 94] C. Rico Pérez - *Resolución de la anáfora discursiva mediante una estrategia de inspiración vectoral.* Proceedings of the SEPLN'94 conference, Cordoba 20-22 July 1994

[Preuß 94 et al] Preuß S., Schmitz B., Hauenschild C., Umbach C. - *Anaphora Resolution in Machine Translation.* In W. Ramm (ed): Studies in Machine Translation and Natural Language Processing, Volume 6 "Text and content in Machine Translation: Aspects of discourse representation and discourse processing", Office for Official Publications of the European Community, Luxembourg, 1994

[Rich & LuperFoy 88] E. Rich, S. LuperFoy - *An architecture for anaphora resolution.* Proceedings of the Second Conference on Applied Natural Language Processing, Austin, Texas, 9-12 February 1988