Anaphors, PPs and disambiguation process for conceptual analysis

Saliha Azzam
CRIL INGENIERIE - CAMS Universite dc Pans-Sorbonnc
174, rue de la Republiquc 92817
Putcaux France
email azzam@cnl-ingfr

Abstract

During the conceptual analysis, the coexistence of several kinds of ambiguities tends to complicate the task of every kind of disambiguation module We tackle this problem here for two types of ambiguities anaphors and the PP attachment We show what kind of problems every conceptual analyser has to face We present, then, our solution to resolve these problems We study the efficiency of the proposed solution and its adequacy regarding dependencies between both ambiguities

1 Introduction

The problems of anaphors and PPs are largely tackled in the NLP domain and still under study For PPs see eg [Frazier and Fodor 1979, Hobbs and Bear, 1990. Wilks, 1985, Shubert, 1986], and for anaphora see among others [Carter, 1987, Sidner, 1983] However, one less addressed issue is that of managing the treatment of both ambiguities in the same conceptual analysis We tackle this issue here by proposing an algorithm to co-ordinate the treatment of these two ambiguities We have already given preliminary ideas about this problem in [Azzam, 1995J presenting an algorithm for this co-ordination We present here more details about the proposed algorithm, moreover we study its efficiency considering the frequency of each kind of ambiguity and their apparition order in the sentence We study also the adequacy of the algorithm considering how deeply these two phenomena depend on each other

The section 2 expands the motivations to tackle the co ordination of anaphors and PP attachment treatment The section 3 gives a brief description of the conceptual analyser CLAM (Cobalt Linguistic Analyser Module) We have implemented CLAM in the context of the COBALT project (LRE 61-011), [Azzam, 1994] The examples are extracted from its corpus in the financial domain (Reuters news) The section 4 presents the proposed algorithm The section 5 evaluates the efficiency of the proposed algorithm and the section 6 its adequacy considering the dependencies existing between both ambiguities

2 What's the problem?

Let's take an example given by Wilks whose study about PP attachment rules is one of the closest to ours In its using essentially semantic information [Wilks 1985] First, he proposes first, for the PP attachment, a rule called the "First Trail Attachment rule", before presenting a more elaborated rule , one of the main reasons he gives to explain the failure of this rule the problem of pronouns *"This rule would of course have to be modified for many special factors, eg pronouns"* . for example, in *"She wanted the dress on the shelf"* this rule normally attaches "dress" to "shelf, but it fails for *"She wanted it on the shelf*[1] Our claim is that the correct application of PP attachment rules must not depend on anaphors occurring before in the text, I e, anaphors should not constitute an obstacle to such rules

More generally, if a sentence contains several kinds of ambiguities and if each kind of ambiguity is treated locally by its own module (e g anaphora module, PP attachment module), then the problem of the interaction between several linguistic phenomena should be managed first

For what concerns anaphors and PPs, how do anaphora resolution and PP attachment procedure interact together? Having in mind that such procedures operate on Conceptual Structures (CSs), we list three main problems that lead to a "stuck situation" There are

0 A PP that should be attached to an anaphor antecedent can not be attached to the CSs until the anaphor has been resolved, i e until *its* antecedent is represented in the CSs

11) Conversely, an anaphor whose antecedent *is* a part of an ambiguous PP can not be resolved until the PP is attached to CSs, 1 e an anaphor can not refer to an entity that is not in the CSs

in) If a PP contains anaphors, the attachment procedure can not perform the attachment, because attachment rules need information about the semantic content of the PP "object" lessened with unresolved anaphors

3 The conceptual analyser

Unlike the way of solving ambiguities (anaphora or PP) that can differ from one system to another, the method we propose for managing interactions between the two disambiguation modules does not depend on our conceptual analyser strategy The problem arises in every conceptual analysis and have the same consequences on the CSs (desenbed in *i, u* and HI)

Our conceptual analyser strategy is described briefly to show how the invoked problem arises dunng the conceptual analysis The strategy of the semantic module consists in a *continuous step by-step "translation"* of the original natural language sentences to CSs represented in NKRL (Narrative Knowledge Representation Language) [Zarn, 1994]

This "translauon" *refers constantly* to the results of the syntactic analysis (syntactic tree) It is a progressive substituuon of the NL terms located in the syntactic tree with *concepts* and *templates* of the conceptual representation language "Tnggennng rules' are a sort of production rules, that are evoked by words of the sentence and allow the activation of NKRL templates The antecedent part of a tnggennng rule is a "syntacuco-semantic filter" expressed under the form of a tree-like structure, to be unified with the syntactic tree If the unification succeeds, the consequent parts are used 10 generate well-formed templates The values caught in the syntactic tree by the filter vanables will fill the *roles* of CSs In case of PP attachment ambiguities, the values associated to "object" PPs are not caught because of their "uncertain' position in the syntactic trees This causes empty or *unfilled* roles in CSs If the caught values are anaphors, they are considered as *unbound* vanables in the CSs and result in unfilled roles in the *CSs*

3 1 PP attachment module

The strategy adopted to treat PP ambiguities in CLAM is charactensed by the following items

It operates on a single incomplete syntactic tree instead of several trees each representing a possible attachment "Incomplete" means that attachments are not performed The syntactic representation Is a kind of "forest" containing the main syntactic tree and the unattached sub-trees, each representing an ambiguous PP

- Each preposition has its own set of *attachment rules* that express syntactic, semantic and pragmatic restnctions on both the PP "object" and CSs roles in order *to* attach the PP to *CSs* For example, one attachment rule associated to the preposition "from" is (stated in a simplified procedural form) *"if the object* of"from" *PP is a location, then fill the empty role location in the CSs"*

- The *disambiguation* procedure fills the *empty roles* in the CSs using the attachment rules and it uses set of heuristic rules to reduce the number of possible readings

3 2 Anaphora module

The anaphora module is characterized by the following items (see also [Carter, 1987]

- It uses a set of resolution rules based on the focusing approach, see [Sidner, 1983]
- Each land of anaphor has Its own set of resolution rules These rules are applied to the conceptual representation Their outputs are candidate antecedents
- The antecedents are validated using a filter based on syntactic restrictions,, e g , the c-command rule [Reinhart, 1983] and semanuc restrictions associated with CSs

4 Interaction between ambiguities

The procedure we propose is based on successive calls to the anaphora module and the PP attachment module (see figure 1) Of course the modules work exclusively on the semantic representation (CSs) of the sentences The output of each call is a set of CSs representing the intermediate results exchanged between each call and on which both modules

operate in turn. Each module tnes to fill the empty roles due to anaphors or unattached PPs

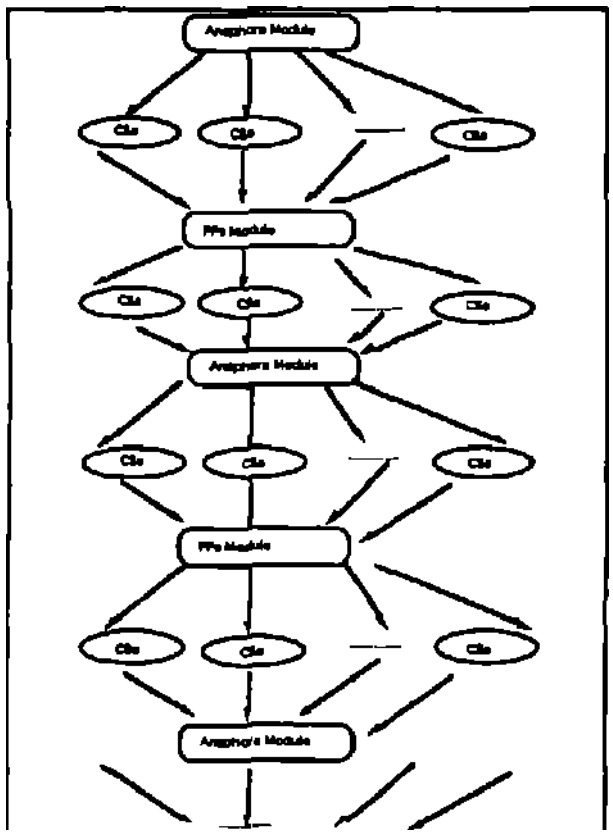

Figure 1

These are the successive steps of the algorithm

1) If the sentence contains anaphora to be resolved, the anaphora module is applied

2) If there are unattached prepositions, the PP attachment procedure is applied

3) If some anaphora are left unresolved, the anaphora module is applied again for the remaining anaphora

4) If there are still unattached PPs, the attachment procedure is applied again only on these prepositions

5) Repeat (3) and (4), until all PPs and Anaphora are treated

As shown in figure 1, each module may have several sets of CSs as an output They represent several disjunctive readings and are due to ambiguities that can not be ruled out by the anaphora resolunon or attachment modules For example, if the attachment rule (see, *infra.* Section 3) associated to the preposition "from" applies on several CSs (which means that there are several "empty location roles" that can be filled by the object of "from"), the vanous possibilities will represent disjunctive readings

The successive calls that are necessary to process all the unattached PPs and anaphors of a given sentence are referred to as a *cycle* The *number* of calls in the cycle is the cycle *length*

The algonthm of interaction for the resolution of ambiguities is based on two main principles

Pnnciple 1

The algorithm is applied on the text sentence by sentence, so that the ambiguioes of the previous sentences have already been considered (they may be resolved or not)

Principle 2

The procedure skips the resolution of a given anaphor when this anaphor is preceded by an unattached preposition This is because the unattached preposition may result in an *empty role* as a parameter in the anaphor resolution rules, which might be a problem The anaphor resolution is postponed until a further phase of anaphora resolution The same applies for PPs that are preceded by unresolved anaphora This explains why each module is called several times

The algonthm faces the three problematic situations presented in Section 2 *(i, u,* and *ill)* It performs as follows

a) When an anaphor occurs before a given preposition in the sentence, its resolution does not depend on where the preposition is to be attached (except for cataphors that are quite rare) Moreover, the attachment of the PP may depend on the resolution of the anaphor In this case, the anaphora module can be applied before the attachment procedure The example 1 (second sentence) below shows how the resolution of the pronouns "it" must be performed first, the "with" PP being attached later

1 General Partners said it sold 1.930,500 shares of GenCorpon Apnl 10 at 118 25 dlrs a share in an open market transaction on the New York Stock. Exchange *Jl said the sale leaves it with 108 GenCorp common shares*

b) When the anaphor occurs after one or several unattached prepositions, it can be an Intra-sentenual anaphor (i e referring to an entity in the same sentence), then its resolution may depend on one of the previous prepositional phrases, as it is

shown in example 2 (second sentence), where "Its" cannot be resolved until the PP "by Matsushita" is attached (so that "Matsushita" fills a role in the CSs) In this case, the resolution of the anaphor is postponed till the next call to the anaphora module according to principle 2 stated above

2 Matsushita Electric Industrial Co said it signed an agreement with the city of Peking to establish a joint venture to produce color television picture tubes *The venture, which would be the Japanese electronics company s first in China, would be equally owned by Matsushita and Us. Chinese partners and would require an Investment of about $100 million*

c) When the anaphor is included in a PP (part cular case of b), the situation is different again PP attachment rules need semantic information about the "object" (head) of the PP (see example 3) , when it is a pronoun, no semantic informanon is available, so that the attachment rules can not be applied The pronouns have to be resolved first, so as to determine what semantic class they refer to , the PP attachment procedure can then be applied When a sequence contains more than two PPs containing anaphors as objects, the length of a cycle is more than 4 Concerning example 3 (second sentence) a cycle of length 4 is required, as follows

3 Miura Kogyo Co , Yamamoto Kozai Co and Ondo Kosakusho Co, all of Hiroshima, Japan, each will own 25% o£ the joint venture *The remaining 25% will be held kx Mazda, throuek its Mazda North America Inc subsidiary*

- The pronoun "its" can not be resolved by the anaphora resolution module because it is preceded by the unattached "by" PP. the resolution of "its" is skipped

- The PP attachment procedure is then called to determine the attachment of "by", but not that of "through" since the object of the "through" PP comprises the pronoun "its" (case c)
- The anaphora module is called again to resolve the anaphoric pronoun "its", which is now possible since all the previous PPs ("by") have been attached and no other anaphors occur before
- Finally, the PP attachment procedure has to be called again for "through"

Let's notice that postponing the resoludon for "its" is justified In this example because the antecedent of this pronoun is the object of "by" PP that has not yet been attached to the CSs This Is not the case for the pronoun "its" *(to its partner)* in example 4, since its resolution does not depend on the attachment of the "in" PP However all the sentences must be treated by the same algorithm, i e in the same way (see also section 6 about equity)

4 *Taiwan s President Enterprises group has agreed to sell its 50 percent stake in Premierfoods Corpjfl Us partner ,* a company official said m an interview

5 Algorithm efficiency

Why is the anaphora module applied before the attachment procedure, and not in the reverse order "> Let's call the algorithm with the reversed order RA (Reversed Algorithm), and the proposed one NA (Normal Algorithm) As a matter of fact, the main reason for our choice between RA and NA can be expressed in terms of efficiency Efficiency is measured in thus case with the length of the cycle This means that the most efficient algorithm is that which processes most of the sentences with the shortest cycle length Our choice is then based on statistical data performed on COBALT corpuses In fact, a higher number of news can be processed with a snorter cycle using NA rather than RA

To illustrate this efficiency, let's first consider an example for which NA is more advantageous than RA example 5, which comprises a sequence of more than two PPs containing anaphors *Anaph PP(Prep Anaph) PP (Prep Anaph)* The processing of this sentence requires a cycle of 4 calls, while it requires 5 calls with the RA Let's consider this sentence in detail with NA

5 General Partners, a Texas Partnership that recently ended us bid to take over GenCorp IC, told the Securities and Exchange Commission, *H* sold neail) of lis 8 5 percent remaining stake m. llLfi company

1) Anaphora that, its, it, its can be resolved (because no unattached PPs occur before)

2) PP the " o f PP can be attached because its object is resolved (its) , in can not be attached because its object is anaphoric ("the company" is a definite noun phrase that refers to "GenCorp")

3) Anaphor "the company' can be resolved because the of PP before has been attached, in the previous call

4) PP the "in" PP can be attached since its object is now known

	NA	RA
Ex 1	2	3
Ex 2	3	4
Ex 3	4	2
Ex 4	4	5
Ex 5	4	5

Figure 2

If we consider the same sentence with RA, 1 e , starting with a call to the PP module, vt will be processed in a cycle of length 5 In fact, the RA involves an additional call to the PP. the first call that is useless since neither the of PP nor the in PP can be processed (their objects are both anaphors) The same remark applies for example 4 for the two in PPs Figure 2 summarises each cycle length for the five examples presented in this section.

To conclude on efficiency, it seems that NA architecture is more suitable for sentences where anaphors precede PPs at the beginning of the sentence (like in examples 1, 4, 5), whereas RA is more adapted for sentences where PPs precede anaphors at the beginning of the sentence (examples 2 and 3) For the financial news we are dealing with In COBALT, in a corpus of 110 news, there are 68 news having anaphors as the first ambiguity to process while the 42 others have PP as the first ambiguity

An alternative solution would be to use a procedure that starts either with PP attachment (RA) or anaphora resolution (NA), depending on the first ambiguity of the sentence to be processed

6 How much do PP and anaphora processing depend on each other [9]

One crucial issue has to be discussed in this paper, namely how much does PP attachment depend on anaphora resolution[7] Conversely, how much does anaphora resolution depend on PP attachment? We remind first the meaning of "dependency"

a)"One anaphor depends on a PP" if the anaphor antecedent is contained in this PP , in this case the anaphor can not be "interpreted" if the PP is not attached to the CSs (see, example 3 for the pronoun "its")

b) "One PP depends on an anaphor" if either the PP attachment position is this anaphor (see example 1 where with is attached to it) , in this case the PP can not be attached if the anaphor is not resolved (i e , its antecedent is not present in the CSs), or the PP "object" is an anaphor , the semantic class of the PP "object" is unknown if the anaphor is not resolved (see examples 3 and 5)

We demonstrated how our algorithm manages these various situations Now, what is the adequacy of this algorithm considering the kind of dependencies existing between these two phenomena? Is it fair to process both phenomena equally, while one of them does not always depend on the other[17] This may appear as a drawback For instance, in example 4, the algorithm attaches the preposition in ("in an interview") only after the two pronouns "its" have been treated, even if the attachment of in does not depend on them However, we consider that postponing PP attachment or anaphora resolution, is not really a deficiency Even if each module is called several times, there is no redundancy in the processing the algonthm should be considered as the splitting of both anaphora resolution and PP attachment procedures into several phases and not as the repetition of each procedure Each anaphor or PP is processed only once, i e if it has already been resolved it will not be processed again

We measured in a certain way, the dependency between the two phenomena Our feeling was that *"PP attachment can give better results without the anaphors being resolved than anaphora resolution without PPs being attached"* We studied three sets of corpuses. corresponding to financial sub-fields, respectively to the fields of "Stake Modification", "Joint Venture" and both fields We calculated (see figure 3) the number of PPs, of anaphors, of anaphors that depend on PPs (according to the dependency "a"above) and of PPs that depend on anaphors (the dependency "b" above) The ratio of PPs that depend on anaphors is very low comparing to that of anaphors mat depend on PPs One of the main reasons is that PPs are largely more frequent in corpuses than anaphors are

Number of news	Anaph	PPs	Anaphors depend on PPs	PPs depend on Anaphors
1st corpus 20 news	66	236	6/66	7/236
2nd corpus 30 news	87	444	16/87	6/444
3rd corpus 60 news	117	460	6/117	15/460

Figure 3

7 Conclusion

Instead of slowing down each other, the anaphors and PPs treatment could be managed efficiently to produce a complete conceptual analysis The solution we propose is to exploit at each step all the results that can be provided by each component, taking into account the general dependencies between the two phenomena This is realised by considering only the results given by the two modules and by avoiding to introduce particular rules for particular cases inside each module The algorithm we proposed in this paper is independent of the approaches we used in both anaphora and attachment modules It concerns rather the way of managing the interaction between the two modules

Future work addresses rather the problems within each disambiguation module We are presently studying how to take into account another important problem that weakens our results that is ' conjunct identification , see e g , [Agarwal and Boggess, 1992] It concerns the identification of the appropriate conjuncts of the co-ordinate conjunctions in a sentence In example 2, the and in *'would be equally owned by Matsushita and its Chinese partners* points to this problem and suggests to take into account other properties between anaphors and PPs as the "parallelism" induced by the form of PPs and anaphors

References

[AgarwaJ and *Bogges,* 1992] Agarwal, R and Boggess L , A Simple but Useful Approach to conjunct identification In *Proceedings of the 30rd Annual meeting of ACL,* pages 15-21, 1992

[Azzam, 1994] Azzam, S CLAM COBALT conceptual analyser Technical Report Dehv6 2 Pans CRIL INGENIERIE 1994

[Azzam, 1995] Azzam, S How to co-ordinate anaphora resolution and PP attachment in a conceptual analyser[1]* In *Proceedings of the 7th European Chapter of ACL* pages 284-285, 1995

[Carter, 1987] Carter, D *Interpreting Anaphors in natural language Texts* Chichester Ellis Horwood 1987

[Frazier and Fodor, 1979] Frazier, L and Fodor, J The sausage machjne A New Two Stage Parsing Model, *Cognition,* 6, 1979

[Hobbs and Bear 1990] Hobbs, J R and Bear J Two Principles of Parse Reference In *Proceeaings of the 13th International Conference on Computational Linguistics* - COLING/90, 3, Karigren, H , ed Helsinki Umveisity Press, 1990

[Reinhart, 1983] Remhart T *Anaphora and Semantic Interpretation* London Croom Helm, 1983

[Schubert 1984] Schubert, L On Parsing Preferences In *Proceedings of the 10th International Conference on Computational Linguistics - COLING 84* Stanford. California, 1984

[Sidner 1983] Sidner, CL Focusing in the *Comprehension of Definite Anaphora Computational Models of Discourse,* Brady, M, and Berwick, R C eds Cambridge (MA) MIT Press, 1983

[Wilks *et at* 1985] Wilks, Y Huang, X , and Fass, D , Syntax, Preference and Right Attachment, In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence,* pages 779-784 1985

[Zam, 1994] Zam GP A Glimpse of NKRL, the 'Narrative Knowledge Representation Language' *In Working Notes of the AAA1 Fall Symposium on Knowledge Representation for NLP in Implemented Systems* MenJo Park (CA) AAAI 1994