

An Acquisition Model for both Choosing and Resolving Anaphora in Conjoined Mandarin Chinese Sentences

Benjamin L. Chen
Application Software Department
Computer and Communication Research Laboratories
Industrial Technology Research Institute
W300/CCL, #1 R&D Rd. 1,
Hsinchu, 30045, Taiwan, R. O. C.
e-mail: blchen@jaguar.ccl.itri.org.tw

Von-Wun Soo
Department of Computer Science
National Tsing-Hua University
Hsinchu, 30043, Taiwan, R. O. C.
e-mail: soo@cs.nthu.edu.tw

Abstract

Anaphoric reference is an important linguistic phenomenon to understand the discourse structure and content. In Chinese natural language processing, there are both the problems of choosing and resolving anaphora. In Mandarin Chinese, several linguists have attempted to propose criteria to explain the phenomenon of anaphora but with controversial results. On the other hand, search-based computational techniques for resolving anaphora are neither the best way to resolve Chinese anaphora nor to facilitate choosing anaphora. Thus, to facilitate both choosing and resolving anaphora with accuracy and efficiency, we propose a case-based learning model G-UNIMEM to automatically acquire anaphoric regularity from a sample set of training sentences¹, which are annotated with a list of features. The regularity acquired from training was then tested and compared with other approaches in both choosing and resolving anaphora.

Keywords: anaphoric reference, semantic roles(case), natural language acquisition, case-based learning.

1 Introduction

In discourse, there may be anaphora in two consecutive sentences. When anaphora appear in a pair of consecutive sentences, the two consecutive sentences are called conjoined sentences. In real life conversation, we frequently choose and resolve anaphora to understand the utterances. There are primarily three types of anaphora in Mandarin Chinese: zero (ellipsis), pronominal (using pronoun) and nominal anaphora[4]. Let's take the conjoined Chinese sentences in (B) to illustrate the phenomenon. The con-

joined sentence in (C) is the English translation of the Chinese sentences in (B).

(B) Yueh-han sheng-bing. [] i-ching
hui-chia-le.
John got sick [] already
gone home
(C) Because John was sick, he has gone home.

Because the anaphora in (B) is a zero anaphora and there is no zero anaphora in English, the antecedent of zero anaphora in (B) must be resolved first before choosing an appropriate pronominal anaphora in Chinese to English translation. In the translation from (C) to (B), it is not good to directly translate an English pronoun to a Chinese pronoun. A better way is to resolve the anaphora in (C) and then choose an appropriate type of anaphora in Chinese.

In natural language processing, better results seems to be attainable if rich linguistic or domain knowledge is available. However it generally costs much and doesn't seem to be realistic. The same situation applies for resolving and choosing anaphora in Mandarin Chinese. If we only used search-based approaches(those that merely used heuristic and algorithmic methods without much linguistic knowledge), the performance was limited. However, when we intended to adopt linguistic knowledge, we found linguists' theories tended to be controversial and less computable. Thus, it motivated us to pursue an acquisition model that could acquire linguistic regularity from corpora and then used the regularity to resolve and choose anaphora.

2 Review of previous approaches

2.1 Search-based approaches

Both history list [1] and Hobbs's naive syntactic algorithm [7] are search-based approaches for resolving anaphora. However, it's not quite obvious to tell which was better than the other with only few examples. Thus, we collected 120 testing instances to test them. Those instances were selected from linguists'

¹This paper is partially supported by the Minister of Economic Affairs, R.O.C. under the project no. 33H3100 at ITRI and by National Science Council of R.O.C. under the grant no. NSC81-0408-E007-02 at National Tsing Hua University. The authors also want to thank Dr. Martha Pollack for valuable comments during 1991 UCSC linguistic institute.

examples, textbooks, essays and novels. Half of them contained zero anaphora the other pronominal.

The result showed that the correct number was 111(92.5%) with Hobbs's syntactic algorithm and 87(72.5%) with the history list approach if first matched were selected. There was 109(90.8%) correct for history list if the last matched were selected. It seemed that both approaches were applicable to resolve anaphora. However, when there are several NPs with the same semantic features, both approaches may get into troubles. Furthermore, both cannot be used to choose anaphora.

2.2 Linguist's criteria

Among linguists' works [3] [5] [11] [12] [14], Tai's criteria [14] was applicable to both choose and resolve anaphora. Others' suffered from difficulties of extracting features or resolving anaphora. Table 1 shows 4 co-references for Tai's criteria, which are all applicable when co-referred NPs are human. For example, consider the following conjoined sentences:

Tai:[Lao Zhang] dao-le Meiguoyihou. [] jiao-le hen-duo pengyou.

John came U.S.A after [] made many friends

Since John came to the U.S.A., he has made many friends.

The subject in the first sentence is human and co-referred by the subject in the second sentence, so this is a subject-subject co-reference. According to Table 1, zero anaphora is preferred to the pronominal one and nominal anaphora is not permitted in this example. Though Tai didn't propose the criteria for resolving anaphora, it was possible to get these criteria just by transforming the choosing criteria in reverse order.

After Tai's criteria were applied to choose and resolve anaphora on the 120 testing instances, we got the success numbers 86(71.7%) and 85(54.2%) respectively. The results failed to meet our satisfaction.

Through above paragraphs, it appears that search-based methods have their limitations due to lack of enough linguistic knowledge and Tai's criteria seems to be applicable to both choose and resolve anaphora. It might be that Tai's criteria were too general to lead to a high success rate. More reliable method to acquire regularity might be required to promote the success rate. We hypothesized the regularity of anaphora could be accounted by causal relations between the features in the conjoined sentences and the antecedents. In the following section, an acquisition model is introduced.

3 G-UNIMEM: A Case-Based Learning Model

In natural language acquisition problem, the restriction of positive-only examples [2] has prohibited many machine learning models as a feasible natural language model. However, a case-based learning approach such as Lebowitz's UNIMEM [9] [10] seems to

be a candidate due to its capability to form concepts incrementally from a rich input domain. Nevertheless, to apply UNIMEM directly to the acquisition of anaphoric regularity in Mandarin Chinese is still not sufficient. We have therefore modified UNIMEM into G-UNIMEM.

G-UNIMEM, a modified version of UNIMEM, is an incremental learning system that uses GBM(Generalized-based Memory) to generalize concepts from a large set of training instances. The program was implemented in Quintus PROLOG and on SUN workstation.

G-UNIMEM differs from UNIMEM in two respects. Firstly, if a drinker got drunk many times after taking either whiskey and water or brandy and water, he would induce that water made him drunk with UNIMEM. This is intuitively incorrect. Whereas, with G-UNIMEM, he would induce that whiskey and water, brandy and water or water would cause him drunk. In this case, G-UNIMEM retains the possible causal accounts without committing to erroneous conclusion. Secondly, G-UNIMEM can extract explicit causal rules from memory hierarchy.

Similar to UNIMEM, G-UNIMEM organizes input training instances into a memory hierarchy according to the frequencies of features. However, its goal is to explicitly express the generalized causal relationships between two specified types of features: *cause features* and *goal features*. Since there may be inconsistency due to lack of cause features, further refinement is needed to obtain consistent causal relations. Thus, there are four different modules in G-UNIMEM to complete different functions in order to achieve this purpose.

3.1 The classifier

The classifier is the first module that processes all training instances for G-UNIMEM. Its function is close to UNIMEM that organizes a hierarchy structure to incrementally accommodate a training instance and at the same time generalize the features based on similarities among training instances. The forming hierarchy is organized as either a *g-c-hierarchy* or a *c-g-hierarchy* depending on the setup of system, which is defined in Definition 1. In Appendix A we show the basic classifier algorithm.

Definition 1 A *g-c-hierarchy* is the hierarchy that every generalized goal feature resides in a *GEN-NODE* and there is no generalized cause feature that resides between the root node and this *GEN-NODE*. A *c-g-hierarchy* doesn't allow any generalized goal feature to reside in the *GEN-NODE* between the root node and any *GEN-NODE* where generalized cause features reside.

Figure 1 and Figure 2 show the forming *g-c-hierarchy* and *c-g-hierarchy* respectively after 13 annotated training sentences are entered into G-UNIMEM. Generally, *g-c-hierarchy* would be chosen since it retained all possible causal accounts. For example, the drinker with *g-c-hierarchy* would induce that whiskey and water, brandy and water or water would cause him drunk; whereas, he would induce whiskey and

water, brandy and water with c-g-hierarchy. The c-g-hierarchy is more efficient since no rules are needed to be generated. Fig. 3 and Fig. 4 show the updating of a GBM before and after inserting a new training instance.

3.2 The rule generator

Once a hierarchy has been constructed by the classifier, the causal rules can be extracted. The rule generator module serves as the role to extract causal rules from the hierarchy. It generates all causal rules from the hierarchy as the regularity is retrieved for predictions.

In Fig. 6, if a testing instance is given for choosing anaphora with a query feature list [(g,type(*)), (g,ante(theme)), (c,fl(theme)), (c,anaphor(theme)), (c,s2(obj)), (c,p(pv))], the retrieval process is searched with a post-order traverse, namely, in the order sequence of the node number 1, 2, 3, 4, 5 and 6. Since there may be more than one candidate, the system can be setup to select either the *first* or the most *specific* one. If the first one is preferred, type(nil) is yielded as the prediction. If the most specific answer is preferred, all possible rules will be tried and the one with the most number of contingent features matched will be the answer (i.e. type(pronoun)).

The sample rules generated from Fig. 1 are shown in Fig. 5. Before generating rules, the GBM is adjusted so that all children of a GEN-NODE are ordered according to their confidence scores of features. Then all rules are generated in a post-order traversal.

3.3 The rule filter

The rule filter removes those rules that are ill-formed and useless. For example, the causal rule 5 in Fig. 5 has no causes which is not a well-formed rule. It also detects conflicting rules. Conflicting rules are those that have different goal feature descriptions, which are accounted by the same cause. For example, the rule 1 and rule 6 in Fig. 5 are conflicting. These rules will be detected in this module and then to be resolved by the feature selector.

3.4 The feature selector

Any two conflicting rules are resolved by the feature selector through augmenting the two rules with mutual exclusive *contingent* cause features, which are prepared in advance. *Dominant* features were used in initial regularity acquisition stage; whereas *contingent* features were used in feature selection stage. The dominant features such as goal features are assumed to be those that must be present in every anaphoric rule. Contingent features are optional. Fig. 6. shows the GBM with g-c hierarchy after feature selection process.

4 Tests using sentences annotated with mixed features

We trained G-UNIMEM with 30, 60, 90, 120 instances using those features mentioned by Tai, and

used all the 120 instances as testing instances. It showed that the approach using Tai's criteria was not promising. There are two reasons. First, none of the success rates was as high as those using the history list approach or Hobbs's algorithm. Second, many conflicting rules remained conflicting due to either that no further features from feature selection were available or too many specific training leading to too many specific rules. These factors decreased the success rate.

4.1 Selecting mixed features

Since Tai's features were not sufficient, more semantic features were considered. Among several linguists' works, we tentatively selected some computational feasible syntactic and semantic features from different sources [3] [5] [11] [12] [13] [14] [15] as in Table 2. An example with annotated features is shown below. The notation [] represents zero anaphora.

(C)[Lao zheng]i qu-le ji-ge [nuren].{ [] hen hui zuo-cai.
 John married a woman { [] well can cook.
 agent theme agent
 hm sub,bv
 nondefinite

John married a woman, and the woman cooked well.

The training feature list for the sentences (C) is :
 [(g,ante(theme)), (g,type(nil)), (c,fl(agent)),
 (c,fl2(theme)), (c,anaphor(agent)), (c,p(bv)),
 (c,s2(sub)), (c,h(hm)), (c,d(nondefinite))] where the notations g and c represent goal and cause features respectively.

4.2 Testing using mixed features

After semantic features has been determined, we trained G-UNIMEM with 30, 60, 90, 120 instances and used all the 120 instances as testing instances each time. We hypothesized to choose semantic roles (i.e. case) as dominant cause features. The features such as ante(CASE), type(X), anaphor(CASE) and fi (CASE) are dominant features and the number of fi is variant. The hypothesis was motivated by Sidner [13] who used semantic roles to determine focus and resolve definite anaphora. The others such as h(Hm), p(POS), s2(SYN), d(D), con(s) belong to contingent features.

4.3 The experimental results

It is interesting that the success numbers in Table 3 increased with the number of training instances. Finally, our results showed that experiments with c-g-hierarchy had a little high accuracy rates (95.8% for resolving and 90.8% for choosing anaphora with 120 training instances) than those with g-c-hierarchy. Both accuracy rates were higher than those with Tai's criteria [14]. Thus, G-UNIMEM with semantic roles as dominant features promised much higher accuracy rate.

In Appendix B we show some sample rules acquired in Horn-like clauses. After examination, either the *agent* or *theme* of first sentence is most likely to

act as antecedents of anaphora. This phenomenon is in coincidence with the investigation on anaphora by Sidner. That is, the agent often appeared as *actor focus* and theme as *default focus*. This is similar to Tai's criteria but is in more compact interpretation.

5 Discussion

There are two concerns in implementing G-UNIMEM:

(1) *The feature set*: Is the assignment of dominant features and contingent features objective? If there is any contingent feature in the assignment that obviously improves the accuracy rate, it should be assigned as dominant feature. We use statistical methods [8] to analyze if contingent features actually improve accuracy rates. If there is no obvious improvement with contingent features, the division of dominant and contingent features is acceptable.

We made the null hypothesis "G-UNIMEM with c-g-hierarchy doesn't have obvious improvement with contingent features" and the alternative hypothesis "G-UNIMEM with c-g-hierarchy has obvious improvement with contingent features". We then got two test values from test statistics: $t_1 = 0.8472$ and $t_2 < 0$. Both test statistics were less than $t_\alpha = .05$ ($= 1.734$ with d.f. = 18). Thus, the null hypothesis "G-UNIMEM with c-g-hierarchy doesn't have obvious improvement with contingent features" was not rejected, which justified that G-UNIMEM using semantic roles as dominant features was valid.

(2) *The sample size*: Compared with actual linguistic domain, the 120 training and testing instances are small. A large corpus is desirable to test the system's performance. If it becomes available, our results would be more objective and reliable.

6 Conclusion

We have illustrated a way of using machine learning techniques to acquire anaphoric regularity in conjoined Mandarin Chinese sentences. The regularity was used to both choose and resolve anaphora with considerable accuracy. Table 4 shows a comparison between different approaches.

In comparison to other approaches, the proposal of using G-UNIMEM as the acquisition model and using semantic roles as dominant features is practical and serves multiple purposes.

References

[1] James Allen (1987), *Natural Language Understanding*, The Benjamin/Cummings Publishing Co.
 [2] Robert Berwick (1986), Learning From Positive-only Examples: The Subjective Principle and Three Case Studies. In R. Michalski, J. Carbonell, and T. Mitchell (ed.) *Machine Learning: An Artificial Intelligence Approach Vol. II*, Chapter 21, Morgan Kaufmann Publishers, Inc.

[3] Ping Chen (1984), *A Discourse Analysis of Third Person Zero Anaphora in Chinese*, reproduced by Indianan University Linguistics Club, Bloomington, Indiana.
 [4] Ping Chen (1987), *A Discourse Approach to Zero Anaphora in Chinese*, Zhongguo Yuwen, Beijing.
 [5] Chauncey C. Chu (1983), *Definiteness, Presupposition, Topic and Focus in Mandarin Chinese*, Student Book company, Taipei, Taiwan.
 [6] John H. Gennari, Pat Langley and Doug Fisher (1989), *Models of Incremental Concept Formation*, *Artificial Intelligence* 40.
 [7] J. Hobbs (1978), *Resolving Pronoun References*, *Lingua* 44, North-Holland Publishing Co.
 [8] Richard Johnson and Gouri Bhatlacharyya (1985), *Statistics: Principles and Methods*, Chapter 6 (Section 4), 9, 10, 11, Published by Hohn Wiley & Sons.
 [9] Michael Lebowitz (1986), *Concept Learning in a Rich Input Domain: Generalization-based Memory*. In R. Michalski, J. Carbonell, and T. Mitchell (ed.) *Machine Learning: An Artificial Intelligence Approach Vol. II*, Chapter 8, Morgan Kaufmann Publishers, Inc.
 [10] Michael Lebowitz (1986), *Integrated Learning: Controlling Explanation*, *Cognitive Science* 10.
 [11] Cherry Ing Li (1985), *Participant anaphora in Mandarin Chinese*, University of Florida Press.
 [12] Mei Du Li (1986), *Anaphoric Structure of Chinese*, Student Book CO., Taipei, Taiwan.
 [13] C. Sidner (1983), *Focusing in the Comprehension of Definite Anaphora*. In *Computational Models of Discourse*, M. Brady and R. Berwick, eds., MIT Press.
 [14] J. H. Tai (1978), *Anaphoric Constraints in Mandarin Chinese Narrative Discourse*. In J. Hinds (ed.) *Anaphora in Discourse*, Linguistic Research, Edmonton, Alberta.
 [15] Ting-Chi Charles Tang (1975), *A Case Grammar Classification of Chinese Verbs*, Hai-Gua Book Company, Taipei, Taiwan.

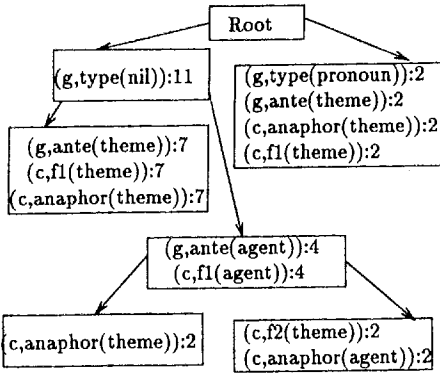


Fig. 1. A g-c-hierarchy of GBM

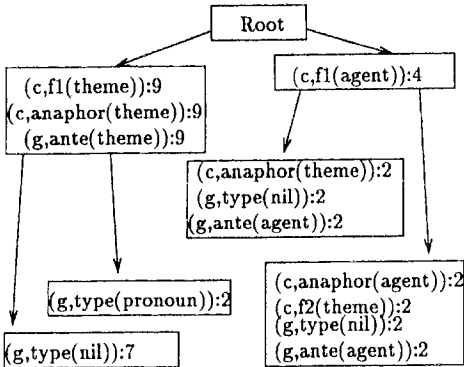


Fig. 2. A c-g-hierarchy of GBM

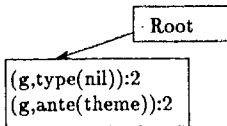


Fig. 3. A GBM with two training instances

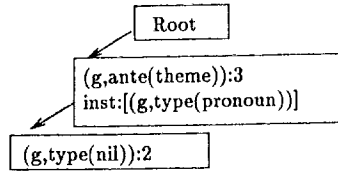


Fig. 4. A new GBM after inserting a new instance [(g,type(pronoun)),(g,ante(theme))]

- 1: [(g,type(nil)),(g,ante(theme))]:- [(c,fl(theme)),(c,anaphor(theme))]
- 2: [(g,type(nil)),(g,ante(agent))]:- [(c,fl(agent)),(c,anaphor(theme))]
- 3: [(g,type(nil)),(g,ante(agent))]:- [(c,fl(agent)),(c,f2(theme)),(c,anaphor(agent))]
- 4: [(g,type(nil)),(g,ante(agent))]:- [(c,fl(agent))]
- 5: [(g,type(nil))]:- []
- 6: [(g,type(pronoun)),(g,ante(theme))]:- [(c,fl(theme)),(c,anaphor(theme))]

Fig. 5. The sample rules generated from Fig. 1.

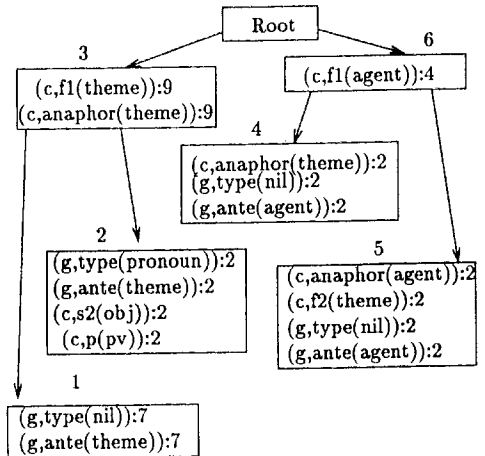


Fig. 6. The c-g hierarchy after conflict resolution from Fig. 2

Table 1: Tai's criteria for choosing anaphora

Co-reference	S -> S	S -> O	O -> S	O -> O
Choose	zero > pronoun	pronoun	pronoun > nominal	pronoun > nominal
Not permitted	nominal	zero	zero	zero

notation S: subject; O: object; ->: co-refer; >: is preferred to.

Table 2: Selected features and references

Class	feature	notation	reference
semantic	antecedent	ante(CASE)	Tang [15], Sidner [13]
	semantic role	f(CASE)	Tang [15], Sidner [13]
	anaphor	anaphor(CASE)	Tang [15], Sidner [13]
	human or nonhuman	h(hm) or hm(nonhm)	Tai [14], Chen [3], Li [11]
syntactic	anaphora	type(X) X= nil(zero); pronoun(pronominal)	Chen [4]
	subject, ...	s2(sub), ...	Tai [14], Chen [3]
	position of anaphora	p(bv) or p(pv) bv: before verb; pv: post-verb	Chen[3]
	definite o nondefinite	d(definite) or d(nondefinite)	Chen [3], Chu [5], Li [11]
	connector	con(s)	Chen [3]

notation : CASE represents a variable for a semantic role and the suffix i of fi indicates the appearing order in a sequence of roles.

Table 3: Comparison of G-UNIMEM using c-g hierarchy against Tai's criteria

Group	Candidate choice	E30	E60	E90	E120	Tai's criteria
rule number		20	31	41	54	42*
Accuracy rate (for resolving)	first	59/62	88/96	104/110	115/120=95.8%	65/120=54.2%
	specific	58/62	87/96	104/110	114/120=95.0%	
	none	58/62	86/96	104/110	111/120=92.5%	30**
Accuracy rate (for choosing)	first	53/60	83/99	95/109	109/120=90.8%	86/120=71.7%
	specific	56/65	83/100	88/110	101/120=84.2%	
	none	55/64	82/99	92/109	102/120=85.0%	
Conflicting number (after resolving)		2	0	4	4	5**

notation: *: for choosing; **: for resolving; accuracy rate = number of success / number of samples that have applicable rules; none in column 2 means no contingent features are used.

Table 4: Comparison of different approaches

Methods	purpose	basis	computability for anaphora
history list	resolve	search	easy
Hobbs's syntactic algorithm	resolve	search	easy
Tai's criteria	choose	predict	somewhat easy
Chen's criteria	choose	predict	not easy
G-UNIMEM & dominant features	choose & resolve	predict	easy

Appendix A. The basic classifier algorithm

input: The current node *N* of the concept hierarchy.

The name *I* of an unclassified instance.

The set of *I*'s unaccounted features *F*.

Results: The concept hierarchy that classifies the instance.

Top-level call: `classifier(Top-node, I, F)`

Variables: *N*, *N'*, *C* and *NC* are nodes in the hierarchy.

G, *H*, and *K* are sets of features.

J is an instance stored on a node.

R is a variable of set.

`Classifier(N, I, F).`

Let *G* be the set of features stores in *N*.

Let *H* be the features in *F* that match features in *G*.

Let *K1* be the features in *F* that do not match features in *G*.

Let *K2* be the features in *G* that do not match features in *H*.

Let *H'*, *K1'* and *K2'* be the sets of features after `Adjust(H,K1,K2,H',K1',K2')`

/* adjust goal and cause features for
g-c-hierarchy or c-g-hierarchy */

if *N* is not the root node,

then if *H* is empty set /* no features match */

then return `False`

else if both *H'* and *K1'* are not empty sets

then { /* split node *N* */

split *N* into *N'* and *NC* where *NC* is a child of *N'*;

N' contains features in *H'* with confidence scores

and *I* as a instance with features *K1'*;

each confidence score in *H'* is increased by 1;

the remaining features and instances belong to *NC*;

return `Split`.

}

else if *H'* and *H* are equal /* all features match */

then increase each confidence score in *N* by 1.

for each child *C* of node *N* /* continue match remaining features */

call `Classifier(C, I, K1')` and collect returns to the set *R*,

if any `Classifier(C, I, K1')` call return `True` or `Split` then break.

if *R* is [`False`] /* All trials fail, try to do generalization */

then for each instance *J* of node *N*

call `Generalize(N, J, I, K1')` and collect returns to the set *R*,

if any `Generalize(N, J, I, K1')` call return `True` then break.

if *R* is [`False`] /* All trials fail, insert *I* as an instance of *N* */

then store *I* as an instance of node *N* with features *K1'*.

return `True`.

Appendix B. Sample rules of regularity with high probability of appearance
in Horn-like clauses

```
[ante(agent), type(nil)]:- [anaphor(agent), f2(theme), f1(agent)]
[ante(agent), type(nil)]:- [anaphor(agent), f1(agent)]
[ante(theme), type(nil)]:- [p(bv), s2(sub), d(nondefinite), anaphor(agent), f1(agent), f2(theme)]
[ante(theme), type(nil)]:- [h(nonhm), anaphor(theme), f1(agent), f2(theme)]
[ante(theme), type(nil)]:- [anaphor(theme), f1(theme)]
[ante(arg), type(pronoun)]:- [anaphor(agent), f2(pred), f1(arg)]
[ante(agent), type(pronoun)]:- [h(hm), d(definite), con(s), anaphor(agent), f2(theme), f1(agent)]
[ante(arg), type(pronoun)]:- [h(hm), anaphor(agent), f2(pred), f1(arg)]
[ante(theme), type(pronoun)]:- [s2(obj), p(pv), d(definite), anaphor(theme), f1(agent), f2(theme)]
[ante(arg), type(pronoun)]:- [h(hm), anaphor(theme), f2(pred), f1(arg)]
```