

Focusing and Reference Resolution in PUNDIT

Deborah A. Dahl

Research and Development Division
SDC -- A Burroughs Company
PO Box 517
Paoli, PA 19301

Science Track

Natural Language

ABSTRACT

This paper describes the use of focusing in the PUNDIT text processing system.¹ Focusing, as discussed by [Sidner1979] (as well as the closely related concept of centering, as discussed by [Grosz1983]), provides a powerful tool for pronoun resolution. However, its range of application is actually much more general, in that it can be used for several problems in reference resolution. Specifically, in the PUNDIT system, focusing is used for *one*-anaphora, elided noun phrases, and certain types of definite and indefinite noun phrases, in addition to its use for pronouns. Another important feature in the PUNDIT reference resolution system is that the focusing algorithm is based on syntactic constituents, rather than on thematic roles, as in Sidner's system. This feature is based on considerations arising from the extension of focusing to cover *one*-anaphora. These considerations make syntactic focusing a more accurate predictor of the interpretation of *one*-anaphoric noun phrases without decreasing the accuracy for definite pronouns.

¹ This work is supported in part by DARPA under contract N00014-85-C-0012, administered by the Office of Naval Research.

1. Background

1.1. Focusing

Linguistically reduced forms, such as pronouns, are typically used in texts to refer to the entity or entities with which the text is most centrally concerned.² Thus, keeping track of this entity, (the *topic*, of [Gundel1974], the *focus* of [Sidner1979], and the *backward-looking center* of [Grosz1983, Kameyama1985]) is clearly of value in the interpretation of pronouns. However, while 'pronoun resolution' is generally presented as a problem in computational linguistics to which focusing can provide an answer (See for example, the discussion in [Hirst1981]), it is useful to consider focusing as a problem in its own right. By looking at focusing from this perspective, it can be seen that its applications are more general than simply finding referents for pronouns. Focusing can in fact play a role in the interpretation of several different types of noun phrases. In support of this position, I will show how focus is used in the PUNDIT (Prolog UNDERstander of Integrated Text) text processing system to interpret a variety of forms of anaphoric reference; in particular, pronouns, elided noun phrases, *one*-anaphora, and context-dependent full noun phrase references.

A second position advocated in this paper is that surface syntactic form can provide an accurate guide to determining what entities are in focus. Unlike previous focusing algorithms, such as that of [Sidner1979], which used thematic roles (for example, *theme*, *agent*, *instrument* as described in [Gruber1976]), the algorithm used in this system relies on surface syntactic structure to determine which entities are expected to be in focus. The extension of the focusing mechanism to handle *one*-anaphora has provided the major motivation for the choice of syntactic focusing.

The focusing mechanism in this system consists of two parts--a **FocusList**, which is a list of entities in the order in which they are to be considered as foci, and a focusing algorithm, which orders the **FocusList**. The implementation is discussed in detail in Section 5.

1.2. Overview of the PUNDIT System

I will begin with a brief overview of the PUNDIT system, currently under development at SDC. PUNDIT is written in Quintus Prolog 1.5. It is designed to integrate syntax, semantics, and discourse knowledge in text processing for limited domains. The system is implemented as a set of distinct interacting components which communicate with each other in clearly specified and restricted ways.

The syntactic component, Restriction Grammar, [Hirschman1982, Hirschman1985], performs a top-down parse by interpreting a set of context-free BNF definitions and enforcing context-sensitive restrictions associated with the BNF definitions. The grammar is generally modelled after that developed by the NYU Linguistic String Project [Sager1981]. Restrictions which enforce context-sensitive constraints on the parse are associated with the bnf rules

² I am grateful for the helpful comments of Lynette Hirschman, Marcia Linebarger, Martha Palmer, and Rebecca Schiffman on this paper. John Dowding and Bonnie Webber also provided useful comments and suggestions on an earlier version.

Some semantic filtering of the parse is done at the noun phrase level. That is, after a noun phrase is parsed, it is passed to the noun phrase semantics component, which determines if there is an acceptable semantics associated with that parse. If the noun phrase is acceptable, the semantics component constructs a semantic representation. If the noun phrase is not semantically acceptable, another parse is sought.

At the conclusion of parsing, the sentence-level semantic interpreter is called. This interpreter is based on Palmer's Inference Driven Semantic Analysis system, [Palmer1985], which analyzes verbs into their component meanings and fills their thematic roles. In the process of filling a thematic role the semantic analyzer calls reference resolution for a specific syntactic constituent in order to find a referent to fill the role. Reference resolution instantiates the referent, and adds to the discourse representation any information inferred during reference resolution.

Domain-specific information is available for both the noun phrase and clause level semantic components through the knowledge base. The domain currently being modelled by SDC is that of computer maintenance reports. Currently the knowledge base is implemented as a semantic net containing a part-whole hierarchy and an *isa* hierarchy of the components and entities in the application domain.

Following the semantic analysis, a discourse component is called which updates the discourse representation to include the information from the current sentence and which runs the focusing algorithm.

2. Uses of Focusing

Focusing is used in four places in PUNDIT -- for definite pronouns, for elided noun phrases, for *one*-anaphora, and for implicit associates.

As stated above, reference resolution is called by the semantic interpreter when it is in the process of filling a thematic role. Reference resolution proposes a referent for the constituent associated with that role. For example, if the verb is *replace* and the semantic interpreter is filling the role of **agent**, reference resolution would be called for the surface syntactic subject. After a proposed referent is chosen for the subject, any specific selectional restrictions on the agent of *replace* (such as the constraint that the agent has to be a human being) are checked. If the proposed referent fails selection, backtracking into reference resolution occurs and another referent is selected. Cooperation between reference resolution and the semantic interpreter is discussed in detail in [Palmer1986]. The semantic interpreter itself is discussed in [Palmer1985].

2.1. Pronouns and Elided Noun Phrases

Pronoun resolution is done by instantiating the referent of the pronoun to the first member of the **FocusList** unless the instantiation would violate syntactic constraints on coreferentiality.³ (As noted above, if the proposed referent fails selection,

³ At the moment, the syntactic constraints on coreferentiality used by the system are very simple. If the direct object is reflexive it must be instantiated to the same referent as the subject. Otherwise it must be a different referent. Obviously, as the system is extended to cover sentences with more complex structures, a more sophisticated treatment of syntactic constraints on

backtracking occurs, and another referent is chosen.)

The reference resolution situation in the maintenance texts however, is complicated by the fact that there are very few overt pronouns. Rather, in contexts where a noun phrase would be expected, there is often elision, or a zero-np as in *Won't power up* and *Has not failed since Hill's arrival*. Zeroes are handled exactly as if they were pronouns. The hypothesis that elided noun phrases can be treated in the same way as pronouns is consistent with previous claims in [Gundel1980] and [Kameyama1985] that in languages such as Russian and Japanese, which regularly allow zero-np's, the zero corresponds to the focus. If these claims are correct, it is not surprising that in a sub-language like that found in the maintenance texts, which also allows zero-np's, the zero should correspond to the focus.

Another kind of pronoun (or zero) also occurs in the maintenance texts, which is not associated with the local focus, but is concerned with global aspects of the text. For example, the field engineer is a default agent in the maintenance domain, as in *Thinks problem is in head select area*. This is handled by defining *default elided referents* for the domain. The referent is instantiated to one of these if no suitable candidate can be found in the **FocusList**.

2.2. Implicit Associates

Focusing is also used in the processing of certain full noun phrases, both definite and indefinite, which involve *implicit associates*. The term implicit associates refers to the relationship between *a disk drive* and *the motor* in examples like *The field engineer installed a disk drive. The motor failed*. It is natural for a human reader to infer that the motor is part of the disk drive. In order to capture this intuition, it is necessary for the system to relate the motor to the disk drive of which it is part. Relationships of this kind have been extensively discussed in the literature on definite reference. For example, implicit associates correspond to *inferrable* entities described by [Prince1981], the *associated use definites* of [Hawkins1978], and the *associated* type of implicit backwards specification discussed by [Sidner1979]. Sidner suggests that implicit associates should be found among the entities in focus. Thus, when the system encounters a definite noun phrase mentioned for the first time, it sequentially examines each member of the **FocusList** to determine if it is a possible associate of the current noun phrase. The specific association relationships (such as part-whole, object-property, and so on) are defined in the knowledge base.

This mechanism is also used in the processing of certain indefinite noun phrases. In every domain, it is claimed, there are certain types of entities which can be classified as *dependent*. By this is meant an entity which is not typically mentioned on its own, but which is referred to in connection with another entity, on which it is dependent. In the maintenance domain, for example, parts such as keyboards, motors, and printed circuit boards are dependent, since when they are mentioned, they are normally mentioned as being part of something else, such as a console, disk drive, or

coindexing using some of the insights of [Reinhart1976], and [Chomsky1981] will be required.

printer.⁴ In an example like *The system is down. The field engineer replaced a bad printed circuit board*, it seems clear that a relationship between the printed circuit board and the system should be represented. Upon encountering a reference to a dependent entity like the printed circuit board, the system looks through the **FocusList** to determine if any previously mentioned entities can be associated with a printed circuit board, and if so, the relationship is made explicit. If no associate has been mentioned, the entity will be associated with a default defined in the knowledge base. For example, in the maintenance domain, parts are defined as dependent entities, and in the absence of an explicitly mentioned associate, they are represented as associated with the system.

2.3. One-Anaphora

PUNDIT extends focusing to the analysis of *one*-anaphora following [Dahl1984], which claims that focus is central to the interpretation of *one*-anaphora. Specifically, the referent of a *one*-anaphoric noun phrase (e.g., *the blue one, some large ones*) is claimed to be a member or members of a set which is the focus of the current clause. For example, in *Installed two disk drives. One failed*, the set of two disk drives is assumed to be the focus of *One failed*, and the disk drive that failed is a member of that set. This analysis can be contrasted with that of [Halliday1976], which treats *one*-anaphora as a surface syntactic phenomenon, completely distinct from reference. It is more consistent with the theoretical discussions of [1976], and [Webber1983].⁵ These analyses advocate a discourse-pragmatic treatment for both *one*-anaphora and definite pronouns. The main computational advantage of treating *one*-anaphora as a discourse problem is that, since definite pronouns are treated this way, little modification is needed to the basic anaphora mechanism to allow it to handle *one*-anaphora. In contrast, an implementation following the account of Halliday and Hasan would be much more complex and specific to *one*-anaphora.

The process of reference resolution for *one*-anaphora occurs in two stages. The first stage is resolution of the anaphor, *one*, and this is the stage that involves focusing. When the system processes the head noun *one*, it instantiates it with the *category* of the first set in the **FocusList** (*disk drive* in this example).⁶ In other words, the referent of the noun phrase must be a member of the previously mentioned set of disk drives. The second stage of reference resolution for *one*-anaphora assigns a specific disk drive as the referent of the entire noun phrase, using the same procedures that would be used for a full noun phrase, *a disk drive*.

The extension of the system to *one*-anaphora provides the clearest motivation for the choice of a syntactic focus in PUNDIT. Before I discuss the kinds of examples

⁴ There are exceptions to this generalization. For example, in a sentence like *field engineer ordered motor*, the motor on order is not part of anything else (yet). In PUNDIT, these cases are assumed to depend on the verb meaning. In this example, the object of *ordered* is categorized as *non-specific*, and reference resolution is not called. See [Palmer1986] for details.

⁵ Although not Webber's analysis in [Webber1978], which advocates an approach similar to Halliday and Hasan's.

⁶ Currently the only sets in the **FocusList** are those which were explicitly mentioned in the text. However, as pointed out by [Dahl1982], and [Webber1983, Dahl1984], other sets besides those explicitly mentioned are available for anaphoric reference. These have not yet been added to the system.

which support this approach, I will briefly describe the relevant part of the focusing algorithm based on thematic roles which is proposed by [Sidner1979]. After each sentence, the focusing algorithms order the elements in the sentence in the order in which they are to be considered as potential foci in the next sentence. Sidner's ordering and that of PUNDIT are compared in Figure 1.

The idea that surface syntax is important in focusing comes from a suggestion by [Erteschik-Shir1979], that every sentence has a *dominant* syntactic constituent, which provides a default topic for the following utterance⁷. Intuitively, the dominant constituent can be thought of as the one to which the hearer's attention is primarily drawn. Operationally the dominance of a constituent is tested by seeing if a referent with that constituent as the antecedent can be cooperatively referred to with an unstressed pronoun in the following sentence.

The feature of *one*-anaphora which motivates the syntactic algorithm is that the availability of certain noun phrases as antecedents for *one*-anaphora is strongly affected by surface word order variations which change syntactic relations, but which do not affect thematic roles. If thematic roles are crucial for focusing, then this pattern would not be observed.

Consider the following examples:

- (1) A: I'd like to plug in this lamp, but the bookcases are blocking the electrical outlets.
 B: Well, can we move one?
- (2) A: I'd like to plug in this lamp, but the electrical outlets are blocked by the bookcases.

Sidner	PUNDIT
Theme	Sentence
Other thematic roles	Direct Object
Agent	Subject
Verb Phrase	Objects of Prepositional Phrases

Figure 1: Comparison of Potential Focus Ordering in Sidner's System and PUNDIT

⁷ As discussed in [Dahl1984] there are problems with Erteschik-Shir's definition of dominance and slightly different definition is proposed. However the details of this reformulation do not concern us here.

B: Well, can we move one?

In (1), most informants report an initial impression that B is talking about moving the electrical outlets. This does not happen for (2). This indicates that the expected focus following (1) A is the outlets, while it is the bookcases in (1) B. However, in each case, the thematic roles are the same, so an algorithm based on thematic roles would predict no difference between (1) and (2).

Similar examples using definite pronouns do not seem to exhibit the same effect. In (3) and (4), *they* seems to be ambiguous, until world knowledge is brought in. Thus, in order to handle definite pronouns alone, either algorithm would be adequate.

(3) A: I'd like to plug in this lamp, but bookcases are blocking the electrical outlets.

B: Well, can we move them?

(4) A: I'd like to plug in this lamp, but the electrical outlets are blocked by the bookcases.

B: Well, can we move them?

(5) and (6) illustrate another example with *one*-anaphora. In (5) but not in (6), the initial interpretation seems to be that a bug has lost its leaves. As in (1) and (2), however, the thematic roles are the same, so a thematic-role-based algorithm would predict no difference between the sentences.

(5) The plants are swarming with the bugs. One's already lost all its leaves.

(6) The bugs are swarming over the plants. One's already lost all its leaves.

In addition to theoretical considerations, there are a number of obvious practical advantages to defining focus on constituents rather than on thematic roles. For example, constituents can often be found more reliably than thematic roles. In addition, thematic roles have to be defined individually for each verb.⁸ Since thematic roles for verbs can vary across domains, defining focus on syntax makes it less domain dependent, and hence more portable. While in principle focus based on thematic roles does not have to be domain-dependent, a general algorithm based on thematic roles would have to rely on a general, domain-neutral specification of all possible thematic roles and their behavior in focusing. Until such a specification exists, a thematic-role based focusing algorithm must be redefined for each new domain as the domain requires the definition of new thematic roles, and because of this, will continue to be less portable than an approach based on syntax.

⁸ Of course, some generalizations can be made about how arguments map to thematic roles. For example, the basic definition of the thematic role *theme* is that, for a verb of motion, the theme is the argument that moves. More generally, the theme is the argument that is most affected by the action of the verb, and its typical syntactic manifestation is as a direct object of a transitive verb, or the subject of an intransitive verb. However, even if these generalizations are accurate, they are no more than guidelines for finding the themes of verbs. The verbs still have to be classified individually.

3. Implementation

3.1. The FocusList and CurrentContext

The data structures that retain information from sentence to sentence in the PUNDIT system are the **FocusList** and the **CurrentContext**. The **FocusList** is a list of all the discourse entities which are eligible to be considered as foci, listed in the order in which they are to be considered. For example, after a sentence like *The field engineer replaced the disk drive*, the following **FocusList** would be created.

```
[[event1],[drive1],[engineer1]]
```

The members of the **FocusList** are unique identifiers that have been assigned to the three discourse entities -- the disk drive, the field engineer, and the event. The **CurrentContext** contains the information that has been conveyed by the discourse so far. After the example above, the **CurrentContext** would contain three types of information:

- (1) *Discourse id's*, which represent classifications of entities. For example, `id(field^engineer,[engineer1])` means that `[engineer1]` is a field engineer.⁹
- (2) Facts about part-whole relationships (**hasparts**). In the example in Figure 2, notice that the lack of a representation of time results in both drives being part of the system, which they are, but not at the same time. Work to remedy this problem is in progress.
- (3) Representations of the events in the discourse. For example, if the event is that of a disk drive having been replaced, the representation consists of a unique identifier (`[event1]`), the surface verb (`replace(time(_))`), and the decomposition of the verb with its (known) arguments instantiated¹⁰. The thematic roles involved are **object1**, the replaced disk drive, **object2**, the replacement disk drive, **time** and **instrument** which are uninstantiated, and **agent**, the field engineer. (See [Palmer1986], for details of this representation). Figure 2 illustrates how the **CurrentContext** looks after the discourse-initial sentence, *The field engineer replaced the disk drive*.

3.2. The Focusing Algorithm

The focusing algorithm used in this system resembles that of [Sidner1979], although it does not use the actor focus and uses surface syntax rather than thematic roles, as discussed above. The focusing algorithm is illustrated in Figure 3. Removing candidates from the **FocusList** when they are no longer eligible to be the referents of pronouns is not currently done in this system. The conditions determining this have not been fully investigated, and since the texts involved are short, few problems are created in practice. This problem will be addressed by future research.

⁹ `field^engineer` is an example of the representation used in PUNDIT for an idiom.

¹⁰ `_8176` is an uninstantiated variable representing the time of the replacement. It appears in several places, such as `included(object2([drive2]),time(_8176))`, and `missing(object1([drive1]),time(_8176))`.


```

id(field ^ engineer,[engineer1]),
id(disk ^ drive,[drive1]),
id(system,[system1]),
id(disk ^ drive,[drive2]),
id(event,[event1]),

haspart([system1],[drive1]),
haspart([system1],[drive2])

event([event1],
  replace(time(_8176),
    [included(object2([drive2]),time(_8176)),
      missing(object1([drive1]),time(_8176)),
      use(instrument(_8405),
        exchange(object1([drive1]),object2([drive2]),time(_8176))),
      cause(agent([engineer1]),
        use(instrument(_8405),
          exchange(object1([drive1]),object2([drive2]),time(_8176))))))

```

Figure 2: CurrentContext after *The field engineer replaced the disk drive.*

(1) **First Sentence of a Discourse:**

Establish expected foci for the next sentence (order FocusList): the order reflects how likely that constituent is to become the focus of the following sentence.

Sentence
Direct Object
Subject
Objects of Prepositional Phrases

(2) **Subsequent Sentences (update FocusList):**

If there is a pronoun in the current sentence, move the focus to the referent of the pronoun. If there is no pronoun, retain the focus from the previous sentence. Order the other elements in the sentence as in (1).

Figure 3: The Focusing Algorithm

4. Summary

Several interesting research issues are raised by this work. For example, what is the source of the focusing algorithm? Is it derivable from theoretical considerations about how language is processed by human beings, or is it simply an empirical observation about conventions used in particular languages to bring discourse entities into prominence? Evidence bearing on this issue would be to what extent the focusing mechanism carries over to other, non-related languages. Kameyama's work on Japanese suggests that there are some similarities across languages. To the extent that such similarities exist, it would suggest that the algorithm is derivable from other theoretical considerations, and is not simply a reflection of linguistic conventions.

This paper has described the reference resolution component of PUNDIT, a large text understanding system in Prolog. A focusing algorithm based on surface syntactic constituents is used in the processing of several different types of reduced reference: definite pronouns, *one*-anaphora, elided noun phrases, and implicit associates. This generality points out the usefulness of treating focusing as a problem in itself rather than simply as a tool for pronoun resolution.

REFERENCES

- [1976]
 Jorge Hankamer and Ivan Sag, Deep and Surface Anaphora. *Linguistic Inquiry* 7(3), 1976, pp. 391-428.
- [Chomsky1981]
 Noam Chomsky, *Lectures on Government and Binding*. Foris Publications, Dordrecht, 1981.
- [Dahl1982.]
 Deborah A. Dahl, Discourse Structure and one-anaphora in English, presented at the 57th Annual Meeting of the Linguistic Society of America, San Diego, 1982..
- [Dahl1984]
 Deborah A. Dahl, The Structure and Function of One-Anaphora in English, PhD Thesis; (also published by Indiana University Linguistics Club, 1985), University of Minnesota, 1984.
- [Erteschik-Shir1979]
 Nomi Erteschik-Shir and Shalom Lappin, Dominance and the Functional Explanation of Island Phenomena. *Theoretical Linguistics*, 1979, pp. 41-86.
- [Grosz1983]
 Barbara Grosz, Aravind K. Joshi, and Scott Weinstein, Providing a Unified Account of Definite Noun Phrases in Discourse. *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, 1983, pp. 44-50.
- [Gruber1976]
 Jeffery Gruber, *Lexical Structure in Syntax and Semantics*. North Holland, New York, 1976.
- [Gundel1974]
 Jeanette K. Gundel, Role of Topic and Comment in Linguistic Theory, Ph.D. thesis, University of Texas at Austin, 1974.
- [Gundel1980]
 Jeanette K. Gundel, Zero-NP Anaphora in Russian. *Chicago Linguistic Society Parasession on Pronouns and Anaphora*, 1980.
- [Halliday1976]
 Michael A. K. Halliday and Ruqaiya Hasan, *Cohesion in English*. Longman, London, 1976.
- [Hawkins1978]
 John A. Hawkins, *Definiteness and Indefiniteness*. Humanities Press, Atlantic Highlands, New Jersey, 1978.

[Hirschman1982]

L. Hirschman and K. Puder, Restriction Grammar in Prolog. In *Proc. of the First International Logic Programming Conference*, M. Van Caneghem (ed.), Association pour la Diffusion et le Developpement de Prolog, Marseilles, 1982, pp. 85-90.

[Hirschman1985]

L. Hirschman and K. Puder, Restriction Grammar: A Prolog Implementation. In *Logic Programming and its Applications*, D.H.D. Warren and M. VanCaneghem (ed.), 1985.

[Hirst1981]

Graeme Hirst, *Anaphora in Natural Language Understanding*. Springer-Verlag, New York, 1981.

[Kameyama1985]

Megumi Kameyama, Zero Anaphora: The Case of Japanese, Ph.D. thesis, Stanford University, 1985.

[Palmer1985]

Martha S. Palmer, Driving Semantics for a Limited Domain, Ph.D. thesis, University of Edinburgh, 1985.

[Palmer1986]

Martha S. Palmer, Deborah A. Dahl, Rebecca J. Schiffman, Lynette Hirschman, Marcia Linebarger, and John Dowding, Recovering Implicit Information, to be presented at the 24th Annual Meeting of the Association for Computational Linguistics, Columbia University, New York, August 1986.

[Prince1981]

Ellen F. Prince, Toward a Taxonomy of Given-New Information. In *Radical Pragmatics*, Peter Cole (ed.), Academic Press, New York, 1981.

[Reinhart1976]

Tanya Reinhart, The Syntactic Domain of Anaphora, Ph.D. thesis, Massachusetts Institute of Technology, 1976.

[Sager1981]

N. Sager, *Natural Language Information Processing: A Computer Grammar of English and Its Applications*. Addison-Wesley, Reading, Mass., 1981.

[Sidner1979]

Candace Lee Sidner, Towards a Computational Theory of Definite Anaphora Comprehension in English Discourse, MIT-AI TR-537, Cambridge, MA, 1979.

[Webber1978]

Bonnie Lynn Webber, *A Formal Approach to Discourse Anaphora*. Garland, New York, 1978.

[Webber1983]

Bonnie Lynn Webber, So What Can We Talk About Now?. In *Computational Models of Discourse*, Michael Brady and Robert C. Berwick (ed.), 1983.