Organizational Issues Arising from the Integration of the Lexicon and Concept Network in a Text Understanding System

Pádraig Cunningham, Tony Veale
Hitachi Dublin Laboratory
Trinity College, College Green, Dublin 2, Ireland

Abstract

A knowledge based system for text understanding will incorporate both lexical and encyclopaedic information. The lexical information is the basis of the parsing process while the encyclopaedic information forms the target representation and is used in the knowledge acquisition process. This paper describes TWIG, a text understanding system where these two knowledge bases are integrated into one representation. There is some theoretical justification for this and it has the advantage of reducing duplication of information in the system. This integration also has the advantage of making conceptual information available during the parsing process. Most of all this integration of diverse information forms a natural basis for a blackboard architecture.

1 Introduction

This paper describes some aspects of the work on TWIG,* a system for automatic concept indexing. TWIG is concerned with producing a concept network which indexes the information in a body of text. This concept network is used as a basis for retrieving information from an electronic document database. The concept network forms an object oriented representation of the salient information in the text. As such, using a concept network based information retrieval system, it should be possible to retrieve information without having to guess the keywords under which a particular document was stored. Instead it will be possible to uncover the appropriate indexing concepts by browsing through the concept network. (see [Parsaye '89] on concept indexing)

The approach to information retrieval assumed by this research is that portions of the original document are retrieved as hypertext and this retrieval is based on the concept index, the generation of which is the subject of this paper. Our approach to concept indexing has three main components; parsing, composition, and filtering. These are implemented as knowledge based processes; the knowledge bases being the lexicon on which the parsing is based and the concept network that is accessed/updated by the composition and filtering processes. In TWIG the lexicon and concept network are integrated into a unified representation. This paper discusses the organisational issues arising from such an integration, which, with its emphasis on linking structurally diverse information (concepts and lexemes) forms a natural basis for a blackboard architecture.

The reasons for integrating the concept network and the lexicon are discussed in section 3. First the overall structure of the document storage and retrieval system is described in the next section. The integrated concept network/lexicon facilitates the overall blackboard architecture of the system that is described in section 4. An example of the overall system in operation is shown in section 5. The paper finishes with some conclusions and outlines our future directions of research.

2 System Overview

In order to give this research on concept indexing some context we must describe the overall document storage and retrieval system of which a concept indexing process would be part. The complete intelligent filing system has been described in [Fujisawa '88] and the rôle of hypertext in the knowledge acquisition process has been outlined in [Cunningham '90].

With the advent of optical storage it is possible to realise the idea of an office in which all documents are stored as digital images on optical disk. This has the advantage of reducing space requirements for storage, as tens of thousands of pages of text can be stored on one optical disk. Further benefits derive from the possibilities for flexible retrieval that exist with an electronic document database. Figure 1 shows an outline system architecture for an intelligent filing system based on such an electronic document database.

(i) The first stage of the filing process is the image capture where the document is scanned in order to store the scanned image on optical disk.

(ii) Before the image is stored it is analysed to identify the regions containing pictures, graphics and text. This is necessary because different compaction algorithms are appropriate for the different types of region. This document analysis process produces both the

* The name is based on the colloquialism *twig* which means 'to understand' and derives from the Gaelic word *tuig* meaning 'understand'.

compacted scanned image and a structural representation of the document that acts as a rudimentary hypertext.

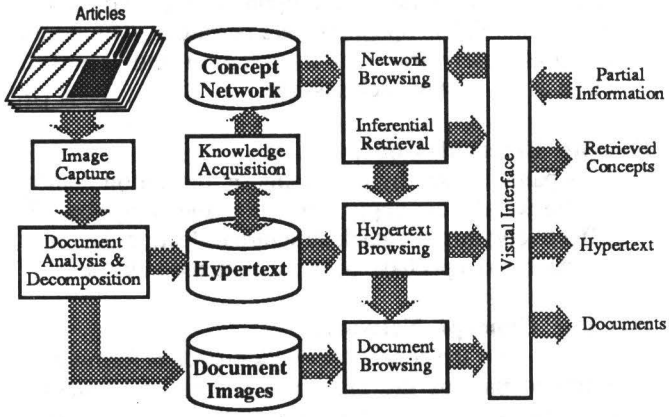

Figure 1. Overall structure of a Concept Network based intelligent filing system.

(iii) This rudimentary hypertext is the input to the knowledge acquisition system. The concept index is produced from this hypertext and the hypertext is fleshed out in the process by the production of reference links.

(iv) This hypertext and concept network form the basis of the information retrieval system. The user identifies interesting concepts in the concept network and the hypertext nodes that reference that concept are retrieved.

(v) A facility exists to both retrieve the scanned image of the original document and to produce a corresponding hard copy if necessary.

The main focus of this paper concerns the knowledge acquisition system, the prototype of which described here is called TWIG. This knowledge acquisition system is blackboard based using object oriented knowledge structures. There are two important knowledge bases associated with the knowledge acquisition system, namely the concept network and the lexicon. The main thesis of this paper concerns the organizational advantages accruing from the integration of both of these knowledge bases and the subsequent blackboard architecture that can be built with this representation as basis. A detailed description of this blackboard structure is presented in section 4. First the object oriented lexicon and concept network that is the basis of the reasoning mechanism will be described.

3 Lexicon & Concept Network

A priori we have an expectation that there is a difference between the concept structure and the lexicon with its associated semantic structure. The expected distinction it is that the concept network reflects encyclopedic information (see [Cunningham '87] for instance) while the lexicon reflect linguistic information.

However, one of the main messages in this paper concerns the integration of the concept network and the lexicon. We have two motivations for doing this. The first is the theoretical support given to this approach from Jackendoff (see [Jackendoff '83]). He argues that both semantic and conceptual structure denote the same level of representation. Dahlgren also argues for an approach that does not differentiate between the form of representation for word meanings and concepts [Dahlgren '88], and draws upon psycholinguistic research to back her claim. Our view is that, while not being identical, the concept network and the lexicon are best represented at the same level of abstraction. Thus, they can be integrated in an application such as TWIG.

The second motivation stems from our early experiments with a system based on a separate lexicon and concept network. Our experience with that system indicated that that approach resulted in considerable duplication of information. This experience supports the theoretical arguments outlined above.

This integration has resulted in some important benefits, especially the manner in which an integrated concept network and lexicon facilitates the blackboard architecture for knowledge acquisition that will be described in section 4. In addition, when a new concept is discovered by the system and entered into the concept network the system can in the future generate a semantic representation of that concept/lexeme for use in parsing.

Figure 2 shows an example concept from the concept network. It represents the class PERSONAL COMPUTER and has subclasses MACINTOSH and IBM COMPATIBLE The Status slot shows that this object is browsable and that subclasses and instances of this concept that are discovered in text will be indexed. This index information will be held in the References slot, which effectively contains pointers to text nodes in the hypertext representation of the document. The slots Synonyms and Lexical contain information that is used in the parsing process while other slots contain information that would conventionally be considered conceptual or encyclopaedic in nature. The Lexical slot contains syntactic information about the concept, represented as a directed acyclic graph (DAG); this is augmented with semantic/conceptual information when required by the parser (see section 4 for more details).

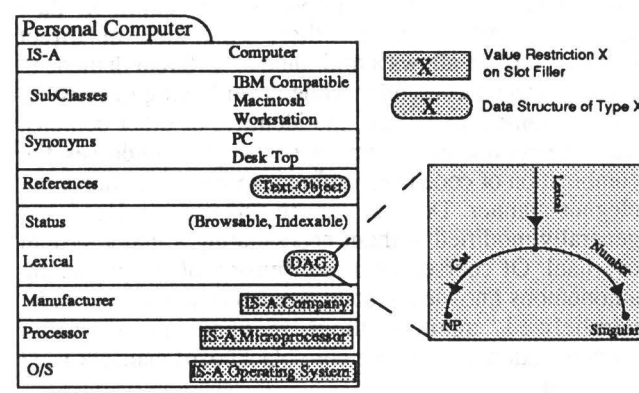

Figure 2. The frame for the concept PERSONAL COMPUTER.

Figure 3 offers a convenient overview of the integrated concept network and lexicon. Since we have decided that the form of representation for word meanings and concepts

can be the same the lexical and conceptual information forms an IS-A hierarchy as shown in the figure.

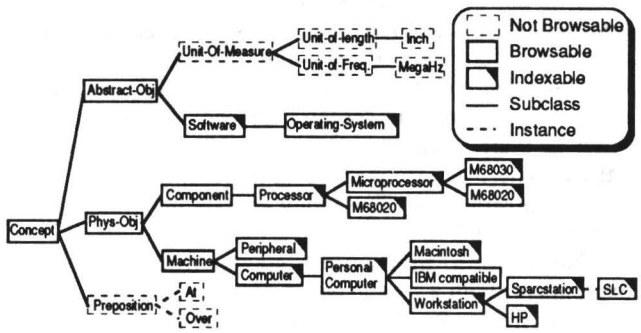

Figure 3. The integrated concept network & lexicon represented as an IS-A hierarchy

This integrated network serves two rôles; it is the conceptual representation of the information in the document database, and it is the source of semantic information for the parsing process. In the current version of the system every concept contains lexical information so every concept can be viewed as a lexeme. However some concepts, for instance the preposition "over" or the unit of measure "inch", are not useful as conceptual information so these are not designated browsable concepts. In addition only some of the browsable concepts are designated indexable; this means that references will only be attached to occurrences of these concepts in the concept indexing process.

4 Knowledge Acquisition Architecture

The entire process of knowledge acquisition, from parsing the initial input text to producing the final conceptual structure, is governed by the use of a blackboard system. Such a control metaphor, which encourages the division of labour among opportunistic and autonomous knowledge agents, allows for a great degree of modularity in the overall system organization. The blackboard serves as a a common reservoir of heterogeneous information, necessitating the individual agents acting upon it to define their interface solely in terms of this common data structure, and not each other. This lack of direct interdependence allows a multitude of different demons to communicate successfully while never having to consider the mechanics (or indeed, the identity) of other demons. Such a network of cooperative *strangers* endorses the combination of code from different sources in a simple but elegant manner. Demons can be updated, replaced and generally modified without necessitating a major system overhaul. Of course, the clean division of labour among autonomous knowledge sources enables the system to be parallelized, again without recourse to a system reorganization, as only the core blackboard manager need be altered.

4.1 Blackboard Structure

Fundamental to the system architecture is a three-panel blackboard which assumes the following format (see Figure 4):-

(i) **The Scratchpad**
This panel is used as a common data store of heterogeneous knowledge, upon which read, write, update and delete operations are permitted. This panel contains solely transient information and is wiped clean at the start of each session.

(ii) **The Referent Context List**
This panel is used by the anaphoric resolution demon to track all concepts entering the blackboard. Effectively, this is a stack structure reflecting the order in which concepts have been encountered in the text.

(iii) **The Concept Network**
We can view the concept network proper as forming an intrinsic part of the blackboard structure; existing concept structures are accessed during the parsing process, the composition/filtering processes update these concepts, and the concept formation process creates and inserts new concepts. Only read, write and update operations are permitted upon this third panel of the blackboard, which is never wiped.

4.2 Demon Organization

The TWIG blackboard system utilizes five distinct knowledge sources, each implemented as an autonomous demon as shown in Figure 4. What follows is a brief overview of the mechanics of these agents.

(i) **Parser**
The Parser demon produces semantic structure representations of individual input sentences which are subsequently written onto the scratch pad.

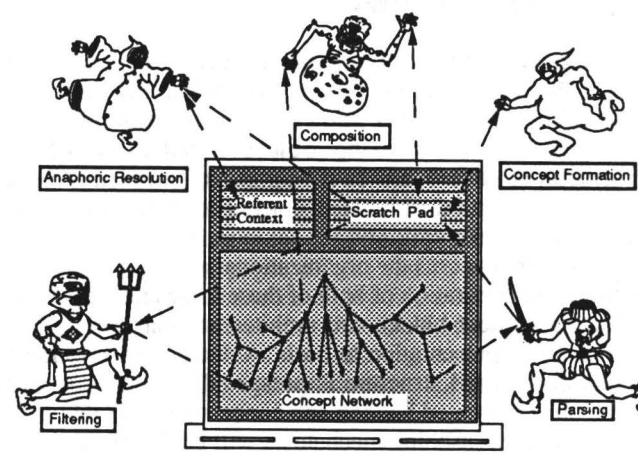

Figure 4. The blackboard architecture of the knowledge acquisition system. The demons reflect the autonomous knowledge sources acting upon the Blackboard.

(ii) **Concept Formation**
This demon takes the semantic structures produced by the parser and creates the corresponding conceptual structures (a collection of related frames) which are then written onto the scratch pad.

(iii) **Anaphoric Resolution**
All concept frames added to the blackboard are compared with concepts currently stored in the referent context list; if a match is found, both referent concept and new concept are merged and the resulting concept is both written onto the scratch pad and pushed onto the top of the referent context stack.

(iv) **Concept Composition**
The process of composition creates links between related concept frames, by putting the value of one into the appropriate slot of another. For example, a newly formed SCSI concept frame might be added to the peripheral slot of the recently created Macintosh SE concept frame. The newly composed frames are written onto the scratch pad.

(v) **Filter**
The filtering demon retrieves composed domain salient concepts from the scratch pad and inserts them into the concept network. As such, the filter demon acts as a functional interface between the scratch pad and concept network panels of the blackboard. Driven by the indexable concepts in the concept network, it takes the composed concepts from the scratch pad, selects concepts that can be related to indexable concepts in the concept network and merges the new concept into the concept network. Actual indexing is performed by including a pointer to the referenced text node on the indexed concept. The user is the final arbiter of what the system should consider salient, and by tagging specific concepts (and thus their subclasses) as either interesting or uninteresting, one can personalize the document indexing process.

4.3 Parser Organization

Fundamental to the implementation of any text understanding system is the correct choice of both parsing model and grammar formalism. The current TWIG system utilizes a combination of chart parser (operating in bottom-up fashion) and unification grammar (the PATR II formalism, utilizing DAG representations of concepts) to create the initial semantic structure representations of the incoming text. These components were chosen based on the following requirements as visualized in the initial system specification:-

(i) **Robustness**
The parser should be capable of producing a set of partial parses from difficult text, and a set of alternative parses when dealing with ambiguous text. It is not expected that the system understand each sentence in its entirety, for this would make unrealistic demands upon the grammar and the corresponding lexicon of the system.

(ii) **Efficiency**
The parser should perform requirement 1 above efficiently and elegantly. Use should be made of existing constituent structures and partial hypotheses, such that the system never performs duplicate analysis of the same input text.

(iii) **Semantic Representation at Concept level**
The mapping between deep structure and semantic structure representations of the input text should be encoded within the conceptual structures of the constituent concepts, and not within the grammar itself. Such a mapping necessitates a directed acyclic graph (DAG) representation (see Figure 6).

(iv) **Expressive Power**
The grammar formalism should be capable of expressing the mapping required by 3 above in a declarative and perspicuous fashion.

4.4 Concept Organization

How does one reconcile the fundamental view of the concept network as an object oriented frame hierarchy with the need to represent semantic information via DAGs, as expressed by requirements 3 and 4 above? Our solution to this problem is simple but effective - simply build the DAG from the concept frame when requested by the parser. A cache slot within the frame can be used to store this newly created DAG to avoid duplication of effort at a later stage; the inherited procedural attachment used to set slot values can be instructed to clear this cache slot whenever the concept is modified or extended, ensuring that the stored dag is also updated when next it is required. Specialized mapping information is also stored within each concept frame to indicate which slots share values, effectively endowing a concept with the expressive abilities of a DAG. As this information is stored within the Lexical slot of the concept frame, it is inherited by all subclasses and instances of that concept (see Figure 2).

A <slot : filler > <feature : value> equivalence is employed when creating a DAG from the corresponding concept frame. Clearly, this implies that no slot have more than one filler if it is to contribute to the construction of the DAG. However, the problem remains of how best to encode the IS-A chain of the concept in the DAG, for such conceptual information is needed by the parser to both curb the analysis of incongruous hypotheses and to build the final semantic structure. TWIG adopts a level number scheme which attributes a different semantic level (feature name) to each superclass of the concept, corresponding to the depth of that class in the concept network. Figure 5 demonstrate the IS-A encoding scheme used upon three different concepts.

In adhering to the object oriented philosophy of the system, our encoding scheme also caters for the class/instance distinction instrinsic to the organization of the concept network. Effectively, this distinction guides the knowledge acquisition phase in the creation of new

concepts, for only classes can serve as the basis of new instances.

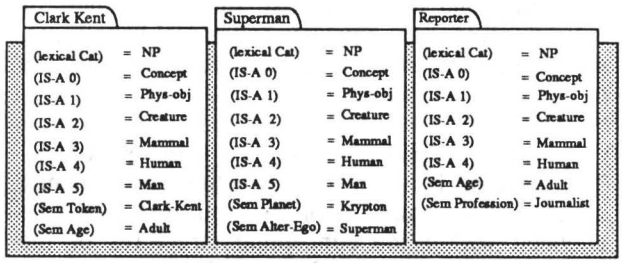

Figure 5. DAG representation of different concepts

Words encountered by the parser which have no corresponding entry in the concept network are initially defined as instances of the class Concept. Such shallow definitions can later be specified further during the knowledge acquisition phase. For instance, the new concept Lois-Lane is promoted to an instance of the class Woman, qualified by the descriptor class Reporter, when the noun phrase "Lois Lane, ace woman reporter" is parsed and subsequently composed.

4.5 Noun Phrase Preprocessing

To minimize the initial chart size and hence the resultant parse time, TWIG utilizes a preprocessor to scan the input text and perform as much initial noun phrase (NP) grouping as possible. Adjacent DAG-compatible concepts are unified to produce a single DAG representation, which is then placed in the chart. For instance, the complex NP "Bruce Wayne, millionaire playboy philantropist" can be simplified by the preprocessor such that it is represented as a single lexical *chunk* (and effectively, a single chart edge).

Clearly, the effectiveness of such a grouping heuristic would be greatly improved if the preprocessor were extended to also handle adjectives. Twig achieves this end through the homogeneous representation of adjectives and nouns. Essentially, adjectives are defined as descriptor classes of the noun types they qualify, thus making adjective noun pairs DAG compatible and hence fair game for the NP preprocessor. For example, phrases such as "small footprint high-resolution multisynch RGB monitor" can subsequently be reduced by the preprocessor into a single *chunk*, considerably reducing the size of the final chart.

4.6 Grammar Organization

The grammar supports a three case system, namely surface, deep and semantic, to facilitate a one pass analysis of the input text which produces as output the corresponding semantic structure representation. The mapping between surface and deep cases is performed explicitly by the grammar, while the mapping between deep and semantic cases is performed internally by the constituent concepts. This adheres to the object oriented philosophy central to our system - essentially each concept defines and governs its own semantic behaviour while presenting a black box interface to the parser. Deep cases are assigned by the grammar and sent to the associated concepts, which perform internal mapping from deep to semantic cases (actual slots in the final conceptual representation). This case hierarchy frees the semantic structure from simply *mirroring* the syntactic structure of the text.

- **Surface Case**
  Dictated by the surface ordering of the input text:.
  First NP → Subject and Second NP → Object
- **Deep Case**
  Dictated by the syntactic structure of the input text.
  Active Voice assigns Subject→Agent
  and Object→Patient
  Passive Voice assigns Object→Agent
  and Subject→Patient.
  The set of deep cases can be an arbitrary open-ended set chosen by the grammar designer to reflect the requirements of the system. For instance, prepositions and other function words may map onto different deep cases.
- **Semantic Case**
  Dictated by the semantics of the host concept, as each concept may employ a different set semantic cases (which are equivalent to slot names).

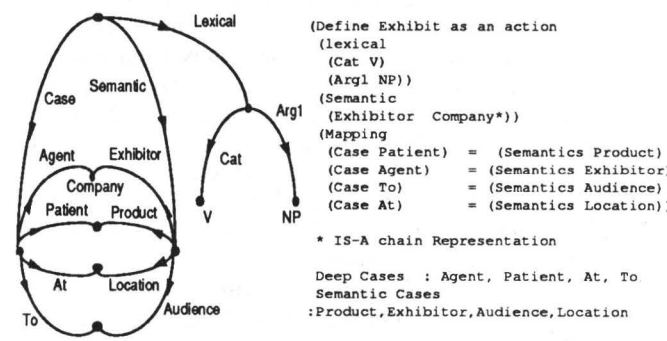

Figure 6. The DAG for the action 'Exhibit'

The above diagram presents the concept Exhibit as seen both by the system (as a DAG) and the lexicon designer (as a frame with additional mapping rules). For purposes of clarity, neither inherited features nor semantic level numbers are displayed in the DAG above. The grammar places deep case assignments into the case feature of the concept, and the mapping rules cause those assignments to be mapped onto the appropriate semantic cases.

5 TWIG in Action: An Example

We present here a brief example of the TWIG KA system at work in our current domain of interest, that of product reviews. The short two sentence narrative, taken from a trade press review of the new SUN SPARCstation SLC (a concept previously unknown to TWIG), serves to illustrate the combined actions of anaphoric resolution, concept formation, composition and filtering.

The parser generates a semantic structure for each input sentence, which is subsequently given a conceptual representation as a collection of related concept frames. In turn, these skeletal concepts are *chunked* into more

consolidated concepts by the composition demon - for instance, the processor concept is integrated into the SLC frame. The necessity for anaphoric resolution arises when dealing with the patient of Incorporate ("It", sentence 2), causing the SLC frame to be linked to that of Incorporate. We use the filing cabinet metaphor to represent the mechanics of the filtering demon, where the cabinet corresponds to the concept network. Only concepts with an associated folder in the cabinet are placed into the concept network, while others are either discarded or have their contents composed into some slot of a relevant concept. Extending the metaphor, we see that allocating a folder for a class of relevant concepts is equivalent to marking that concept as interesting in the concept network.

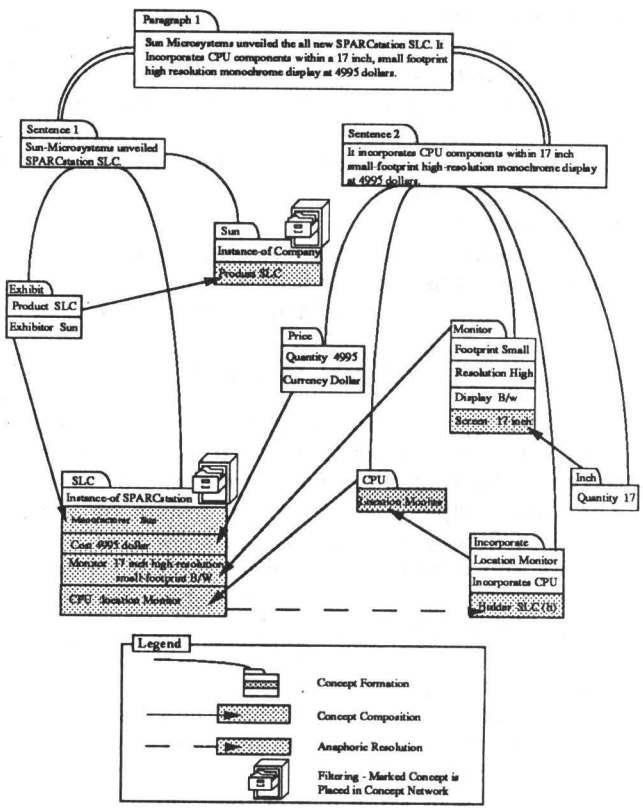

Figure 7. The concept indexing process using a short passage from a product review.

6 Conclusion

The main conclusion from the research described in this paper is that, in natural language understanding systems, the lexicon and the concept network can and should be integrated into one representation. This is supported by the theoretical arguments presented by Jackendoff and Dahlgren and by our experience with TWIG.

This integration has three important advantages:-

- It eliminates the duplication of information that occurs when the concept network and lexicon are maintained separately.
- It allows for the use of the conceptual information in the parsing process.
- It forms the basis of a blackboard for the total knowledge acquisition system. The blackboard architecture is an ideal architecture for a task of this type [Clancey '90].

Our current work concentrates on the representation of the integrated concept network and lexicon, as this clearly is the key to the effectiveness of the system.

7 References

[Clancey '90] Clancey W. J., 'Implications of the system-model-operator metaphor for knowledge acquisition', in Proceedings of the First Japanese Knowledge Acquisition for Knowledge-Based Systems Workshop, ed. H. Motoda, R. Mizoguchi, J. Boose, B. Gaines, pp65-80, 1990.

[Cunningham '87] Cunningham P., Brady M., "Qualitative reasoning in electronic fault diagnosis", IJCAI-'87, pp443-445, 1987.

[Cunningham '90] Cunningham P., Fujisawa H., Hederman L., Cummins F., 'A combined approach to text retrieval using concept networks linked to hypertext', in Proceedings of the First Japanese Knowledge Acquisition for Knowledge-Based Systems Workshop, ed. H. Motoda, R. Mizoguchi, J. Boose, B. Gaines. pp235-248, 1990.

[Dahlgren '88] Dahlgren K., Naïve Semantics for Natural Language Understanding, Kluwer Academic Publishers, 1988.

[Fujisawa '88] Fujisawa H., Hatakeyama A., "Intelligent filing system with knowledge-base", Hitachi Review, pp323-328, Vol. 37, No.5, 1988.

[Jackendoff '83] Jackendoff R. S., Semantics and Cognition, MIT Press, 1983.

[Parsaye '89] Parsaye K., Chignell M., Khoshafian S., Wong H., Intelligent Databases, Wiley, 1989.

[Shieber '86] Shieber S.M., An introduction to unification approaches to grammar, CSLI Lecture Notes Number 4. 1986.

[Sowa '84] Sowa J.F., Conceptual Structures, Addison Wesley, 1984.

[Winograd '83] Winograd T., Language as a cognitive process: Volume 1, Syntax, Addison Wesley, 1983.