Using exceptions in a semantic network for a natural language application

Brigitte BIEBOW Pascal COUPEY Sylvie SZULMAN

LIPN-CNRS
Av. J.B. Clément 93430 VILLETANEUSE
tel: (33 1) 49 40 36 09

ABSTRACT

This paper presents the advantages of using a semantic network with default and exception abilities for a specific problem in natural language processing. More precisely, we show that the features of this semantic network provide an adequate representation of texts expressing a general rule and its exceptions. General statements are usually expressed in natural language by the presence of some specific words, and they occur frequently in our application domain. Our research is effectively integrated in a whole project which is an interactive user-aided tool for the description of requirements including the use of graphics and textual comments. The general cases of treatments are represented by semantic rules which are not strict and admit exceptions. The inferences supported by the network are justified by an interpretation in default logic.

1 INTRODUCTION

The "understanding" of natural language by computers needs a translation of sentences into some formal representation and the use of some commonworld knowledge represented in a knowledge base. Semantic networks have often been used in natural language processing since 70's (13). As knowledge base for syntactic and semantic processing, a KL-ONE (5) semantic network was used in the RUS project of (4). More recently, many research works lead to add expression of exceptions in semantic networks (8). On the practical point of view, L. Rector (11) showed the interest of defaults and exceptions in a semantic network to represent medical knowledge. The system of treatment of NL that we present is based on a semantic network.

Our research takes place in a whole project which uses artificial intelligence techniques to help the acquisition of requirements in software engineering. The project wants to offer an "intelligent" environment for standard methodologies describing requirements, including the use of graphics and textual comments on the objects defined in the graphics. These comments are analyzed syntactically and semantically, then translated into the semantic network. Previous knowledge on general and specific domain is represented as the initial state of the network. This prior knowledge eases the parsing and allows a partial validation of new texts. The semantic analysis permits a coherence control of the specification and a gradual enrichment of the knowledge base. The final result is a reliable specification in which the most common inconsistencies have been avoided.

Our semantic network includes default, exception and negation links. Default links are the most usual in our application. Indeed, all along the enrichment process of a specification, the user describes roughly the principal features and then refines them. Often, the particular cases are in contradiction with the general one. But it is important to represent both of them, and it is possible only with exception link to keep a coherent network. The only other possibility would be to know all the particular cases at the very beginning (6), which is unrealistic, and to prohibit general descriptions, which is unfortunate.

First of all, we describe briefly our system and its application domain. Then we give the syntax of our semantic network, and the representation of natural language expressions into the semantic network language, with special interest to default and exceptions links. Finally we give an interpretation of the network objects in default logic.

2 THE NATURAL LANGUAGE SYSTEM

The system of natural language analysis and representation includes six modules.

- the editor: it gives a tool that is for a user to express his/her specifications in some rigid structure. After a study of the language used in software specification domain, we have defined a few sentence types sufficient to describe requirements. This typology has led us to restrict the allowed syntax to some "templates" which contain some semantic information. Each template corresponds to both a syntactic structure and a rule for network construction. These templates are the sentence units of requirements (for instance, function definition, function explanation, data definition, by examples or by decomposition).

- the syntactic analyzer: based on ATN (15), it checks for syntactic correctness of the template content and sends some syntactic relations and translates them into the network language. It uses a semantic lexicon in the spirit of (14) and specific translation rules some of which are predefined in the lexicon. The resulting interpretation is given to the system user who can either request the system to perform another interpretation, or rewrite his/her text.

- the semantic analyzer: it checks for semantic validity of the syntactic structures, builds semantic relations and translates them into the network language. It uses a semantic lexicon in the spirit of (14) and specific translation rules some of which are predefined in the lexicon. The resulting interpretation is given to the system user who can either request the system to perform another interpretation, or rewrite his/her text.

- the semantic network: the network is KL-ONE-like, augmented with default, exception and negation links. It includes the knowledge base of the application and a base controller. This base contains a minimal set of general knowledge and domain knowledge in a long term memory, and the semantic translation of the texts written by the user in a working memory. The first is not called in question when the second is enriched. After the enrichment, the network controller performs a validation to verify the global structural correctness of the new network.

- the semantic enrichment module: it communicates with the semantic network and determines the validity of an integration and the possible network modifications to be performed before. A provisional version of the modified portion of the network is built allowing for effective enrichment only after a first step of validation. This local validation is performed by the network controller, and by the enrichment module. This latter uses
semantic rules on the domain which are not represented in the network. Once this first step of validation finished, the system proposes to the user an interpretation of the sentence. If he/she is satisfied, the semantic representation is integrated into the network.

- the validation module: it verifies that the modifications of the network are not in contradiction with the interpretation of the network entities. Some possible new contradictions could appear, or some subtle incoherences like a disrespect of constraints of non circularity on some particular relations.

When an inconsistency is detected and not resolved, or when ambiguity is not solved, the system informs the user. The semantic network does not integrate ambiguity.

3 NETWORK SYNTAX AND INFERENCE RULES

We present the only part of (7) used in this paper.

3.1 Syntax

The initial kernel of the semantic network includes general knowledge: general semantic information about cases (in a case grammar terminology (9) (3)) e.g. the action concept and its roles, and about the world, e.g. the concepts entity, physical object, person, and application-specific knowledge e.g. for the specification of the clocking system of a company, the concepts to clock, work place, hour's overtime. Once this kernel defined, it is gradually enriched from analysis of sentences.

Our unit of work for the enrichment is a semantic relation. It can be represented in the network by a new concept which would be sub-concept of an old one, or by a strict or default link (ISKINDOF, EXCLUSION or ROLE) between two concepts, or by a restriction on a link (value or cardinality restriction on role, or exception on any kind of default link). In the following examples we present the use of strict, default, and exception links.

4 HOW TO USE NETWORK OBJECTS TO REPRESENT NL?

Let be a sentence \( P = [D] NPl \) be \( [ADV] \) \( SA \) where:

- \( D \) is one of the indefinite adjectives (tout, tous, chaque) and ADV is an adverb (toujours, jamais).
- at least one of \( D \) or ADV is present in the sentence.
- \( SA \) is a subject attribute represented by a concept named here \( NPl \). It can be either a \( NP \) or an adjective.
- \( NP1 \) is represented by a concept named \( NP1 \).

P will be represented by a strict ISKINDOF (or EXCLUSION if ADV is a strengthened negative adverb as never) link between \( NP1 \) and \( NP2 \):

4.1 What to express by strictness?

A strict link does not support exception. We use it only when the statement is strengthened by the presence of some adverb as (toujours, jamais, always, never) or some determiner as (tous, chaque, all, every). As example, we explain the most current use of ISKINDOF or EXCLUSION strict links.

Let be a sentence \( P = [D] NPl \) be \( [ADV] \) \( SA \) where:

- \( D \) is one of the indefinite adjectives (tout, tous, chaque) and ADV is an adverb (toujours, jamais).
- at least one of \( D \) or ADV is present in the sentence.
- \( SA \) is a subject attribute represented by a concept named here \( NPl \). It can be either a \( NP \) or an adjective.
- \( NP1 \) is represented by a concept named \( NP1 \).

P will be represented by a strict ISKINDOF (or EXCLUSION if ADV is a strengthened negative adverb as never) link between \( NP1 \) and \( NP2 \):
use default link. This avoids any further difficult restructuring of the network in case of enrichment by some new contradictory information. We illustrate this with a rule for creating default roles.

Let be a sentence \( P = \text{NP} \), on which the semantic analysis gives as result:
- \( \text{NP} \) is the agent of the verb \( V \).
- \( \text{NP} \) and \( V \) are represented by concepts named respectively \( \text{NP} \) and \( V \).

The semantic relation between \( \text{NP} \) and \( V \) is represented in the semantic network by a default role named agent-1 from the concept \( \text{NP} \) to the concept \( V \). This role expresses that to play the role agent for \( V \) is a definitional property\(^1\) of \( \text{NP} \). The reverse agent role from \( V \) to \( \text{NP} \) is deductible; it is explicitly created if there is something more to express on it.

The negative form of the sentence would be represented by a cardinality \((0,0)\).

\[ \begin{align*}
\text{manager agent-1} & \quad \text{to clock} \\
(0,0) & \\
\text{"les cadres ne pointent pas"} & \\
\text{(managers do not clock)}
\end{align*} \]

4.3 What to express by an exception link?

When the network detects a contradiction with a default statement, it can be easily resolved by an exception link. Most often, the enrichment module decides to build the necessary exception link after asking the user a confirmation of this interpretation. We give examples of affirmative statements which are then negated; the treatment is similar for the inverse case.

4.3.1 When to create an exception link on an ISKINDOF [EXCLUSION] link?

Let suppose that the network contains the representation of the sentence \( P = \text{NP1} \). Let a new sentence \( P' = D \text{NP1} \) be not \( \text{NP2} \). where \( D \) is one of the indefinite adjectives (certains, quelques, some, any).

\( P' \) will be represented by a new subconcept \( \text{NP1-2 of NP1} \), with strict links ISKINDOF to \( \text{NP1} \) and EXCLUSION to \( \text{NP2} \). The network detects a contradiction: \( \text{NP1-2 ISKINDOF NP2 by transitivity from P and P', and NP1-2 EXCLUSION NP2 from P'} \). The enrichment resolves it by creating an exception link from the concept \( \text{NP1-2} \) to the default link from the concept \( \text{NP1} \) to the concept \( \text{NP2} \).

\[ \begin{align*}
\text{secretary} & \quad \text{woman} \\
\text{sec-not-woman}
\end{align*} \]

\( \text{les secrétaires sont des femmes; quelques secrétaires ne sont pas des femmes (secretaries are women; some secretaries are not women)} \]

\( \text{employees clock} \)

4.3.2 When to create an exception on a default role?

An exception link upon a default role inhibits the inheritance of the role cardinality or role value for a subconcept. Let a sentence be \( P = \text{NP} \), in the network by a role agent-1 expressing the property for \( \text{NP} \) to play the role agent for \( V \).

Let a new sentence \( P1 = \text{NP1 be NP} \). \( \text{NP1} \) inherits the role agent-1 from \( \text{NP} \).

Let now the sentence \( P2 = \text{NP1 not V} \). The network detects the contradiction; it is resolved by creating an exception link on the cardinality of the inherited role from \( \text{NP1} \) to agent-1. The cardinality of the restricted role is \((0,0)\) to express the negation. The idea is equivalent for an exception upon a value.

\[ \begin{align*}
\text{employees agent-1} & \quad \text{to clock} \\
(1,n) & \\
\text{"les employés pointent"} & \\
\text{(employees clock)}
\end{align*} \]

5 INTERPRETATION IN DEFAULT LOGIC

For details on default logic, see (12).

A default is written like: \( X : Y \)

\( X \) is the prerequisite (that is necessary to apply the default), \( Y \) is the justification (from which the negation prevents the application of the default), \( Z \) is the consequent (that is inferred when the default is applied).

A normal default is like: \( X : Y \)

A semi normal default is like: \( X : Y \wedge P \)

We used "semi-normal" defaults derived from research of (10).

For each default link ("iskindof" and "exclusion") they attribute an unary predicate. The advantage of these defaults is the modularity. For the interpretation of the modus tollens we are inspired by (1) who proposed a default called "free" because it has no prerequisite.

'ISKINDOF' default link:

\( A \wedge B \quad \Rightarrow \quad (A(x) \Rightarrow B(x) \wedge \text{Lab}(x)) \)

For the modus tollens interpretation: If the default is applicable we infer \( A(x) \Rightarrow B(x) \), so if \( A(x) \) is true we infer \( B(x) \) else if \( -B(x) \) is true we infer \( -A(x) \).

'EXCLUSION' default link:

\( A \wedge B \quad \Rightarrow \quad (A(x) \Rightarrow -B(x) \wedge \text{Lab}(x)) \)

'GENERIC ROLE' default link:

\( A \quad \Rightarrow \quad (R(x,y) \Rightarrow B(y)) \quad (R(x,y) \Rightarrow B(y)) \)

Default (*) interprets the role cardinality and default (**) the role "value". Our defaults justify the rules used by our algorithms (cf. 3.2.). Under some constraints on the network, its net theory has a single extension. That means that there is no ambiguity in our knowledge base.

---

\( ^1 \)Definitional properties are inherited, they correspond to starting links; the roles arriving to a concept express only the ability for an instance of the concept to play this role; the reverse role is necessary when to play a role is part of the definition of a concept, as it is the case here.

EUROSPED '89, Paris, France, September 1989
6 CONCLUSION

We have presented the basic rules for using a semantic network with exception ability for the representation of natural language. The existence of exception and default links allows to express together general rules and particular cases which do not respect them. Without exception, the network had to be modified to avoid contradiction. The general rule must be suppressed to keep only the particular ones. This requires to reorganize the network, which is costly and difficult. Moreover, in our application of enrichment, this happens often, and anyway general rules with their particular cases are both needed. This work results in a knowledge base of the application in which most of the inconsistencies and all ambiguities have been removed. Moreover, the deduction mechanism is justified by an interpretation in default logic. This paper focus on exception use without details on natural language analysis or on the enrichment part which are studied thoroughly in (2). But we only treat a restricted natural language without grappling hard difficulties. An implementation in Smalltalk80 is in progress on Apollo workstations.

7 REFERENCES

(1) BESNARD P.
Une Procédure de Décision en Logique non monotone
Thèse de 3ème Cycle, Rennes I, France, 83

(2) BIEBOW B., SZULMAN S.
Enrichment of semantic network for requirements expressed in natural language
IFIP World Computer Congress, San Francisco, USA, 89

(3) BINOT J.L.
SABA: Vers un système portable d'analyse du français écrit
PhD thesis, University of Liège, Belgique, 86

(4) BOBROW R.J.
Knowledge Representation for Syntactic/Semantic Processing
Proc. of AAAI, Stanford, August 80

(5) BRACHMAN R.J., SCHMOLZE J.G.
An Overview of the KL-ONE Knowledge Representation System
Cognitive Science, Vol. 9, n°2, 85

(6) BRACHMAN R.J.
I lied about the trees
A.I. Magazine, Fall 85

(7) COUPEY P.
Etude d'un réseau sémantique avec gestion des exceptions
PhD thesis, University of Paris-Nord, France, January 89

(8) Etherington D.W., REITER R.
On inheritance hierarchies with exceptions
Proc. of AAAI, Washington, USA, August 83

(9) FILLMORE C.J.
The case for case
in "Universals in Linguistic Theory", BACH and HARMS eds,
Holt, Rinehart and Winston, Inc., New-York, 68

(10) FROIDEVAUX C., KAYSER D.
Inheritance in Semantic Networks and in Default Logic
Non-Standard Logic for Automated Reasoning
(P.Smet, A.Mandani, D.Dubois, H.Prade eds.) Academic Press 88

(11) RECTOR A.L.
Inheritance and Ambiguity in Semantic Networks for Medical Knowledge and Representation
Proc. of ECAI, Brighton, England, July 86

(12) REITER R.
A logic for Default Reasoning
Artificial Intelligence vol.13 n°1-2, April 80

(13) SIMMONS R.F., BRUCE B.C.
Some relations between predicate calculus and semantic net representation of discourse

(14) SMALL S., RIEGER C.
Parsing and Comprehending with Word Experts (A Theory and its Realization)
TR-1039, University of Maryland, April 81

(15) WOODS W.A.
Transition Network grammars for natural language analysis
CACM, Vol. 13, n°10, 70