

A STRUCTURED REPRESENTATION OF WORD-SENSES FOR SEMANTIC ANALYSIS.

Maria Teresa Pazienza
Dipartimento di Informatica e Sistemistica,
Universita' "La Sapienza", Roma

Paola Velardi
IBM Rome Scientific Center

ABSTRACT

A framework for a structured representation of semantic knowledge (e.g. word-senses) has been defined at the IBM Scientific Center of Roma, as part of a project on Italian Text Understanding. This representation, based on the conceptual graphs formalism [SOW84], expresses deep knowledge (pragmatic) on word-senses. The knowledge base data structure is such as to provide easy access by the semantic verification algorithm. This paper discusses some important problem related to the definition of a semantic knowledge base, as depth versus generality, hierarchical ordering of concept types, etc., and describes the solutions adopted within the text understanding project.

INTRODUCTION

The main problem encountered in natural language (NL) understanding systems is that of the trade-off between depth and extension of the semantic knowledge base. Processing time and robustness dramatically get worse when the system is required to deeply understand texts in unrestricted domains.

For example, the FRUMP system [DEJ79], based on *scripts* [SHA77], analyzes texts in a wide domain by performing a superficial analysis. The idea is to capture only the basic information, much in the same way of a hurried newspaper reader.

A different approach was adopted in the RESEARCHER system [LEB83], whose objective is to answer detailed questions concerning specific texts. The knowledge domain is based on the description of physical objects (MPs: Memory Pointers), and their mutual relations (RWs: Relation Words).

A further example is provided by BORIS [LEH83], one of the most recent systems in the field of text understanding. BORIS was designed to understand as deeply as possible a limited number of stories. A first prototype of BORIS can successfully answer a variety of questions on divorce stories; an extension to different domains appears however extremely complex without structural changes.

The current status of the art on knowledge representation and language processing does not offer readily available solutions at this regard. The system presented in this paper does not propose a panacea for

semantic knowledge representation, but shows the viability of a deep semantic approach even in unrestricted domains.

The features of the Italian Text Understanding system are summarized as follows:

- Text analysis is performed in four steps: *morphologic*, *morphosyntactic*, *syntactic* and *semantic* analysis. At each step the results of the preceding steps are used to restrict the current scope of analysis. Hence for example the semantic analyzer uses the syntactic relations identified by the parser to produce an initial set of possible interpretations of the sentence.
- Semantic knowledge is represented in a very detailed form (*word_sense pragmatics*). Logic is used to implement in a uniform and simple framework the data structure representing semantic knowledge and the programs performing semantic verification.

For a detailed overview of the project and a description of morphological and syntactical analyses refer to [ANT87] In [VEL87] a text generation system used for NL query answering is also described.

The system is based on VM/PROLOG and analyzes press_agency releases in the economic domain. Even though the specific application oriented the choice of words to be entered in the semantic data base, no other restrictions were added. Press agency releases do not present any specific morphologic or syntactic simplification in the sentence structure.

This paper deals with definition of knowledge structures for semantic analysis. Basically, the semantic processor consists of:

1. a dictionary of word definitions.
2. a parsing algorithm.

We here restrict our attention to the first aspect: the semantic verification algorithm is extensively described in [PAZ87]

The representation formalism adopted for word definitions is the conceptual graph model [SOW84], summarized in Section 2. According to this model, a piece of meaning (sentence or word definition) is represented as a graph of *concepts* and *conceptual relations*

Section 3 states a correspondence between conceptual categories (e.g. concepts and relations) and word-senses. A dictionary of hierarchically structured conceptual relations is derived from an analysis of grammar cases.

Section 4 deals with concept definitions and type hierarchies. Finally, Section 5 gives some implementation detail.

The present extension of the knowledge base (about 850 word-sense definitions) is only intended to be a test-bed to demonstrate the validity of the knowledge representation scheme and the semantic analyzer. The contribution of this paper is hence in the field of computer science and his objective is to provide a tool for linguistic experts.

THE CONCEPTUAL GRAPH MODEL

The conceptual graph formalism unifies in a powerful and versatile model many of the ideas that have been around in the last few years on natural language processing. Conceptual graphs add new features to the well known *semantic nets* formalism, and make it a viable model to express the richness and complexity of natural language.

The meaning of a sentence or word is represented by a directed graph of *concepts* and *conceptual relations*. In a graph, concepts are enclosed in boxes, and conceptual relations in circles; in the *linear* form, adopted in this paper, boxes and circles are replaced by brackets and parenthesis. Arrows indicate the *direction* of the relations among concepts.

Concepts are the generalization of physical perceptions (MAN, CAT, NOISE) or abstract categories (FREEDOM, LOVE). A concept has the general form:

[NAME: *referent*]

The *referent* indicates a specific occurrence of the concept NAME (for example [DOG: Fido]).

Conceptual relations express the semantic links between concepts. For example, the phrase "John eats" is represented as follows:

[PERSON: John] <--(AGNT)<--[EAT]

where (AGNT) is a diadic relation used to explicit the active role of the *entity* John with respect to the *action* of eating.

In order to describe word meanings, in [SOW84] several types of conceptual graphs are introduced:

1. Type definitions.

The *type* of a concept is the name of the class to which the concept belongs. Type labels are structured in a hierarchy: the expression $C > C'$ means that the type C is more general than C' (for example, ANIMAL $>$ MAN); C is called the *supertype* of C' .

A type C is defined in terms of *species*, that is the more general class to which it belongs, and *differentia*, that is what distinguishes C from the other types of the same *species*. The type definition for MAN is :

[ANIMAL]-- > (CHRC)--> [RATIONAL]

where (CHRC) is the *characteristic* relation.

2. Canonical graphs.

Canonical graphs express the semantic constraints (or *semantic expectations* ruling the use of a concept. For example, the canonical graph for GO is:¹

[GO]-
(AGNT)--> [MOBILE_ENTITY]
(DEST)--> [PLACE]

Many of the ideas contained in [SOW84] have been used in our work. The original contribution of this paper can be summarized by the following items:

- find a clear correspondence between the words of natural language and conceptual categories (concepts and relations).
- provide a lexicon of conceptual relations to express the semantic formation rules of sentences
- use a *pragmatic* rather than *semantic expectation* approach to represent word-senses. As discussed later, the latter seems not to provide sufficient information to analyze not trivial sentences.
- To make a clear distinction between *word-sense concepts* and *abstract types*. It is not viable to arrange word-senses in a type hierarchy and to preserve at the same time the richness and consistency of the knowledge base.

The following sections discuss the above listed items.

Concepts, relations and words.

The problem analyzed in this section concerns the translation of a words dictionary into a concept-relation dictionary. Which words are concepts? Which are relations? Which, if any, are redundant for meaning representation?

Concepts and relations are semantic categories which have been adopted with different names in many models. Besides conceptual graphs, Schank's conceptual dependency

¹ Word definitions in linear form are represented by wrighting in the first line the name of the word W (concept or relation) to be defined, and in the following lines a list of graphs, linked on their left side to W .

[SHA72] and semantic nets in their various implementations [BRA79] [GRI76] represent sentences as a net of concepts and semantic links.

The ambiguity between concepts and relations is solved in the conceptual dependency theory, where a set of *primitive* acts and conceptual dependencies are employed. The use of primitives is however questionable due to the potential loss of expressive power.

In the semantic net model, relations can be role words (father, actor, organization etc.) or verbs (eat, is-a, possess etc.) or position words (on, over, left etc.), depending on the particular implementation.

In [SOW84] a dictionary of conceptual relations is provided, containing role words (mother, child, successor), modal or temporal markers (past, possible, cause etc.), adverbs (until).

In our system, it was decided to derive some clear guidelines for the definition of a conceptual relation lexicon. As suggested by Fillmore in [FIL68], the existence of semantic links between words seems to be suggested by lexical surface structures, such as word endings, prepositions, syntactic roles (subject, object etc.), conjunctions etc. These structures do not convey a meaning *per se*, but rather are used to relate words to each other in a meaningful pattern.

In the following, three correspondence rules between words, lexical surface structures and semantic categories are proposed.

Correspondence between words and concepts.

Words are nouns, verbs, adjectives, pronouns, not-prepositional adverbs. Each word can have synonyms or multiple meanings.

R1: *A biunivocal correspondence is assigned between main word meanings and concept names. Proper names (John, Fido) are translated into the referent field of the entity type they belong to ([PERSON: John]).*

Correspondence between determiners and referents

Determiners (the, a, etc.) specify whether a word refers to an individual or to a generic instance.

R2: *Determiners are mapped into a specific or generic concept referent.*

For example "a dog" and "the dog" are translated respectively into [DOG: *] and [DOG: *x], where * and *x mean "a generic instance" and "a specific instance". The problem of concept instantiation is however far more complex; this will be objective of further study.

Correspondence between lexical surface structures and conceptual relations

The role of prepositions, conjunctions, prepositional adverbs (before, under, without etc.), word endings (nice-st, gold-en) verb endings and auxiliary verbs is to relate words, as in "I go *by* bus", modify the meaning of a name, as in "she is the *nicesst*", determine the tenses of verbs as in "I *was* going", etc.

Like words, functional signs may have multiple roles (e.g. by, to etc.), derivable from an analysis of grammar cases. (The term *case* is here intended in its extended meaning, as for Fillmore).

R3: *A biunivocal correspondence is assumed between roles played by functional signs and conceptual relations.*

Conceptual relations occurrences which have a linguistic correspondent in the sentence (as the one listed above) are called *explicit*. This does not exhaust the set of conceptual relations; there are in fact syntactic roles which are not expressed by signs. For example, in the phrase "John eats" there exist a subject-verb relation between "John" and "eats"; in the sentence "the nice girl", the adjective "nice" is a quality complement of the noun "girl". Conceptual relations which correspond to these syntactic roles are called *implicit*.

A conceptual relation is *only identified by its role* and might have implicit or explicit occurrences. For example, the phrases "a book about history" and "an history book" both embed the argument (ARG) relation:

[BOOK]-->(ARG)-->[HISTORY]

The translation of surface lexical structure into conceptual relations allows to represent in the same way phrases with the same meaning but different syntactic structure, as in the latter example.

Conceptual relations also explicit the meaning of syntactic roles. For example, the subject relation, which expresses the active role of an entity in some action, corresponds to different semantic relation, like agent (AGNT) as in "John reads", initiator (INIT) as in "John boils potatoes" (John starts the process of boiling), participant (PART) as in "John flies to Roma" (John participates to a flight), instrument (INST) as in "the knife cuts". The genitive case, expressed explicitly by the preposition "of" or by the ending "'s", indicates a social relation (SOC_REL) as in "the doctor of John" or in "the father of my friend", part-of (PART-OF) as in "John's arm", a real or metaphorical possession (POSS) as in "John's book" and "Dante's poetry", etc. (see Appendix).

The idea of ordering concepts in a type hierarchy was extended to conceptual relations. To understand the need of a relation hierarchy, consider the following graphs:

[BUILDING]-->(AGE)-->[YEAR: #50]

[BUILDING]-->(EXTEN)-->[HEIGHT: #30]

[BUILDING]-->(PRICE)-->[LIRE: #5.000]

(AGE), (EXTEN) and (PRICE) represent respectively the age, extension and price relations. By

defining a supertype (MEAS) relation, the three statements above could be generalized as follows:

[BUILDING]-->(MEAS)-->[MEASURE: *x]

Appendix 1 lists the set of hierarchically ordered relation types. At the top level, three relation categories have been defined:

1. *Role.* These relations specify the role of a concept with respect to an action (John (AGNT) eats), to a function (building *for* (MEANS) residence) or to an event (a delay *for* (CAUSE) a traffic jam).
2. *Complement.* Complement relations link an entity to a description of its structure (a golden (MATTER) ring) or an action to a description of its occurrence (going *to* (DEST) Roma).
3. *Link.* Links are entity-entity or action-action type of relations, describing how two or more kindred concepts relate with respect to an action or a way of being. For example, they express a social relation (the mother *of* (SOC_REL) Mary), a comparison (John is *more* (MAJ) handsome *than* Bill), a time sequence (the sun *after* (AFTER) the rain), etc.

STRUCTURED REPRESENTATION OF CONCEPTS.

This section describes the structure of the semantic knowledge base. Many natural language processing systems express semantic knowledge in form of *selection restriction* or *deep case constraints*. In the first case, semantic expectations are associated to the words employed, as for canonical graphs; in the second case, they are associated to some abstraction of a word, as for example in Wilk's *formulas* [WIL73] and in Shank's primitive *conceptual cases* [SHA72].

Semantic expectations however do not provide enough knowledge to solve many language phenomena. Consider for example the following problems, encountered during the analysis of our text data base (press agency releases of economics):

1. *Metonymies*

"The state department, the ACE and the trade unions sign an agreement"

"The meeting was held at the ACE of Roma"

In the first sentence, ACE designates a human organization; it is some delegate of the ACE who actually sign the agreement. In the second sentence, ACE designates a plant, or the head office where a meeting took place.

2. *Syntactic ambiguity*

"The Prime Minister Craxi went to Milano for a meeting"

"President Cossiga went to a residence for handicapped"

In the first case, *meeting* is the *purpose* of the act *go*, in the second "handicapped" case specifies the *destination* of a building. In both examples, syntactic rules are unable to determine whether the prepositional phrase should be attached to the noun or to the verb. Semantic expectations cannot solve this ambiguity as well: for example, the canonical graph for GO (see Section 2) does not say anything about the semantic validity of the conceptual relation PURPOSE.

3. *Conjunctions*

"The state department, the ACE and the trade unions sign an agreement"

"A meeting between trade unionists and the Minister of the Interior, Scalfaro"

In the first sentence, the comma links to different human entities; in the second, it specifies the name of a Minister.

The above phenomena, plus many others, like metaphors, vagueness, ill formed sentences etc., can only be solved by adopting a *pragmatic* approach for the semantic knowledge base. Pragmatics is the knowledge about word uses, contexts, figures of speech; it potentially unlimited, but allows to handle without severe restrictions the richness of natural language. The definition of this *semantic encyclopedia* is a challenging objective, that will require a joint effort of linguists and computer scientists. However, we do not believe in short cut solution of the natural language processing problem.

Within our project, the following guidelines were adopted for the definition of a semantic encyclopedia:

1. Each word-sense have an entry in the semantic data base; this entry is called in the following a *concept definition*
2. A concept definition is a detailed description of its semantic expectations *and* of its semantically permitted *uses* (for example, a *car* is included as a possible subject of *drink* as in "my car drinks gasoline", a *purpose* and a *manner* are included as possible relations for *go*)
3. Each word use or expectation is represented by an *elementary graph* :

(1) [W] · · · (REL_CONC) <-> [C]

where W is the concept to be defined, C some other concept type, and <-> is either a left or a right arrow.

Partitioning a definition in elementary graphs makes it easy for the verification algorithm to determine whether a specific link between two words is semantically permitted or not. In fact, given two word-senses W1 and W2, these are semantically related by a conceptual relation REL_CONC if

there exist a concept *W* in the knowledge base including the graph:

[W] <-> (REL_CONC) <-> [C]

where $W \supseteq W1$ and $C \supseteq W2$. To reduce the extent of the knowledge base, *C* in (1) should be the most general type in the hierarchy for which the (1) holds. The problem of defining a concept hierarchy is however a complex one. The following subsection deals with type hierarchies.

Word-senses and Abstract Classes

Many knowledge representation formalisms for natural language order linguistic entities in a type hierarchy. This is used to deduce the properties of less general concepts from higher level concepts (*property inheritance*). For example, if a proposition like the one expressed by graph (1) is true, then all the propositions obtained by substitution of *C* with any of their subtypes must be true. However, generalization of properties is not strictly valid for linguistic entities; for example the graphs:

(2) [GO]-->(OBJ)-->[CONCRETE]
 (3) [WATCH]-->(AGNT)-->[BLIND]

are both false, even though they are specializations respectively of the following graphs:

(4) [MOVE]-->(OBJ)-->[CONCRETE]
 (5) [WATCH]-->(AGNT)-->[ANIMATE]

In fact, the sentences "to go something" and "a blind watches" violate semantic constraints and meaning postulates: generalization does not preserve both completeness and consistency of definitions. In addition, if a pragmatic approach is pursued, one quickly realizes that no word-sense definition really includes some other; each word has its own specific uses and only partially overlap with other words. The conclusion is that it is not possible to arrange word-senses in a hierarchy; on the other side, it is impractical to replace in the graph (1) the concept type *C* with all the possible word-senses *W_i* for which (1) is valid. A compromise solution has been hence adopted. The hierarchy of concepts is structured as follows:

1. There are two levels of concepts: *word-senses* and *abstract classes*;
2. Concepts associated to word-senses (indicated by italic cases) are the leaves of the hierarchy;
3. Abstract conceptual classes, as MOVE_ACTS, HUMAN_ENTITIES, SOCIAL_ACTS etc. (upper cases) are the non-terminal nodes.

In this hierarchy word-sense concepts are never linked by supertype relations to each other, but at most by brotherhood. Definitions are provided only for word-senses; abstract classes are only used to generalize elementary graphs on word uses.

This solution does not avoid inconsistencies; for example, the graph (included in the definition of the word-sense *person*):

(6) [*person*]-->(AGNT)-->[MOVE_ACT]

is a semantic representation of expressions like: John moves, goes, jumps, runs etc. but also states the validity of the expression "John is the agent of flying" which is instead not valid if John is a person. However the definition of *fly* will include:

(7) [*fly*]-->(AGNT)-->[WINGED_ANIMATES]
 (8) [*fly*]-->(PARTICIPANT)-->[HUMAN]

The semantic algorithm (described in [PAZ87]) asserts the validity of a link between two words *W1* and *W2* only if there exist a conceptual relation to represent the meaning of that link. In order for a conceptual relation to be accepted:

1. This relation must be included in some elementary graph of *W1* and *W2*
2. The type constraints imposed by the elementary graphs must be satisfied for both *W1* and *W2*.

In conclusion, it is possible to write general conditions on word uses without get worried about exceptions. The following section gives an example of concept definition.

Concept definitions

Concept definitions have two descriptors: *classification* and *definition*.

1. Classification.

Besides the supertype name, this descriptor also includes a *type definition*, introduced in Section 2. For example, the type definition for *house* is "building for residence", which in terms of conceptual graphs is:

[BUILDING] <-> (MEANS) <-> [RESIDENCE]

where BUILDING represents the *species*, or supertype, and (MEANS) <-> [RESIDENCE] the *differentia*.

2. Definition.

This descriptor gives the structure and functions of a concept. The definition is partitioned in three subareas, corresponding to the three conceptual relation categories introduced in the previous section.

- a. Role. For an *entity*, this field lists the *actions*, *functions* and *events*, and for an *action* the subjects, objects and proposition types that can be related to it by means of role type relations. For example, the role subgraph for *think* would be

(AGNT)-->[HUMAN]
 (OBJ)-->[TOP]

(MEANS)--> [brain]
 (PURPOSE)--> [AIM]

while for *book* would be:

(MEANS) <--[ACT_OF_COMMUNICATION]
 (OBJ) <--[MOVE_POSITION]

b. Complement.

This graph describes the *structure* of an entity or the *occurrence* (place, time etc.) of an action. This is obtained by listing the concept types that can be linked to the given concept by means of complement type relations. A complement subgraph for EAT is:

(STAT)--> [PLACE]
 (TIME)--> [TIME]
 (MANNER)--> [GUSTATORY_SENSATION]
 (QUALITY)--> [QUALITY_ATTRIBUTE]
 (QUANTITY)--> [QUANTITY: *x]

while for *book* is:

(ARG) <--[PROPOSITION: *]
 (MATTER)--> [paper]
 (PART_OF)--> [paper_sheet]

c. Link.

This graph lists the concepts that can be related to a given concept by means of link type relations. A link subgraph for *house* is:

(POSS) <--[HUMAN]
 (INCL)--> [HUMAN]
 (INCL)--> [DOMESTIC_ANIMAL]
 (INCL)--> [FURNITURE]

and for *eat*:

(AND)--> [drink]
 (OPPOSITE) <--[starve]
 (PREC)--> [hunger]
 (AFTER)--> [satiety]

Note that some elementary graph expresses a relation between two terminal nodes (as for example the opposite of *eat*); in most cases however conditions are more general.

AN OVERVIEW OF THE SYSTEM.

This paper focused on semantic knowledge representation issues. However, many other issues related to natural language processing have been dealt with. The purpose of this section is to give a brief overview of the text understanding system and its current status of implementation. Figure 1 shows the three modules of the text analyzer.

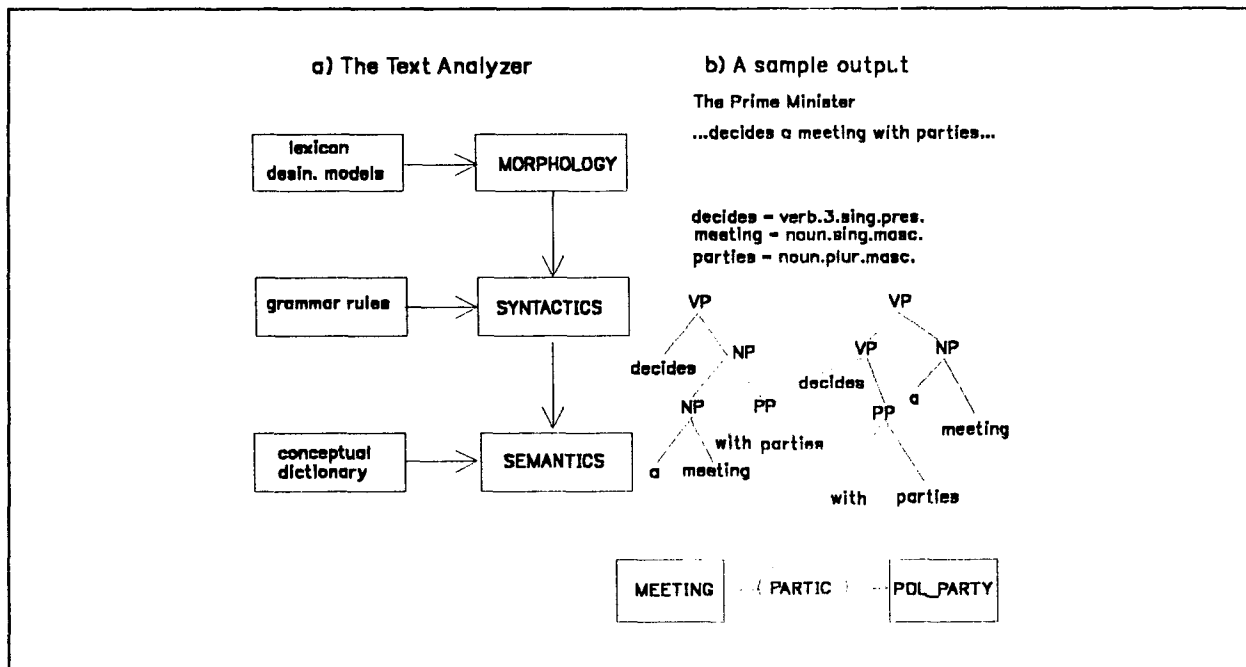


Figure 1. Scheme of the Text Understanding System

All the modules are implemented in VM/PROLOG and run on IBM 3812 mainframe. The morphology associates at least one lemma to each word; in Italian this task is particularly complex due to the presence of recursive

generation mechanisms, such as alterations, nominalization of verbs, etc. For example, from the lemma *casa* (home) it is possible to derive the words *cas-etta* (little home), *cas-ett-ina* (nice little home), *cas-ett-in-accia* (ugly nice little

home) and so on. At present, the morphology is complete, and uses for its analysis a lexicon of 7000 lemmata [ANT87].

The syntactic analysis determines syntactic attachment between words by verifying grammar rules and forms agreement; the system is based on a context free grammar [ANT87]. Italian syntax is also more complex than English: in fact, sentences are usually composed by nested hypotactical phrases, rather than linked paratactical. For example, a sentence like "John goes with his girl friend Mary to the house by the river to meet a friend for a pizza party" might sound odd in English but is a common sentence structure in Italian.

Syntactic relations only reveal the surface structure of a sentence. A main problem is to determine the correct prepositional attachments between words: it is the task of semantics to explicit the meaning of preposition and to detect the relations between words.

The task of disambiguating word-senses and relating them to each other is automatic for a human being but is the hardest for a computer based natural language system. The semantic knowledge representation model presented in this paper does not claim to solve the natural language processing problem, but seems to give promising results, in combination with the other system components.

The semantic processor consists of a semantic knowledge base and a parsing algorithm. The semantic data base presently consists of 850 word-sense definitions; each definition includes in the average 20 elementary graphs. Each graph is represented by a *pragmatic rule*, with the form:

(1) $CONC_REL(W,*x) < -COND(Y,*x)$.

The above has the reading: "*x modifies the word-sense W by the relation CONC_REL if *x is a Y". For example, the PR:

$AGNT(think,*x) < -COND(HUMAN_ENTITY,*y)$.

corresponds to the elementary graph:

[think]-->(AGNT)-->[HUMAN_ENTITY]

The rule $COND(Y,*x)$ requires in general a more complex computation than a simple supertype test, as detailed in [PAZ87]. The short term objective is to enlarge the dictionary to 1000 words. A *concept editor* has been developed to facilitate this task. The editor also allows to visualize, for each word-sense, a list of all the occurrences of the correspondent words within the press agency releases data base (about 10000 news).

The algorithm takes as input one or more parse trees, as produced by the syntactic analyzer. The syntactic surface structures are used to derive, for each couple of

possibly related words or phrases, an initial set of hypothesis for the correspondent semantic structure. For example, a noun phrase (NP) followed by a verb phrase (VP) could be represented by a subset of the LINK relations listed in the Appendix. The specific relation is selected by verifying type constraints, expressed in the definitions of the correspondent concepts. For example, the phrase "John opens (the door)" gives the parse:

NP = NOUN(John)
VP = VERB(opens)

A subject-verb relation as the above could be interpreted by one of the following conceptual relations: AGNT, PARTICIPANT, INSTRUMENT etc. Each relation is tested for semantic plausibility by the rule:

(2) $REL_CONC(x,y) < - (x: REL_CONC(x,*y=y)) \& (y: REL_CONC(*x=x,y))$.

The (2) is proved by rewriting the conditions expressed on the right end side in terms of $COND(Y,*x)$ predicates, as in the (1), and then attempting to verify these conditions. In the above example, (1) is proved true for the relation AGNT, because:

$AGNT(open,person: John) < - (open: AGNT(open,*x=person: John)) \& (person: AGNT(*y=open,person: John))$
 $(open: AGNT(open,*x) < -COND(HUMAN_ENTITY,*x)$
 $(person: AGNT(*y,person) < -COND(MOVE_ACT,*y))$.

The conceptual graph will be

[PERSON: John] <--(AGNT) <--[OPEN]

For a detailed description of the algorithm, refer to [PAZ87]. At the end of the semantic analysis, the system produces two possible outputs. The first is a set of short paraphrases of the input sentence: for example, given the sentence "The ACE signs an agreement with the government" gives:

The Society ACE is the agent of the act SIGN.
AGREEMENT is the result of the act SIGN.
The GOVERNMENT participates to the AGREEMENT.

The second output is a conceptual graph of the sentence, generated using a graphic facility. An example is shown in Figure 2. A PROLOG list representing the graph is also stored in a database for future analysis (query answering, deductions etc.).

As far as the semantic analysis is concerned, current efforts are directed towards the development of a query answering system and a language generator. Future studies will concentrate on discourse analysis.

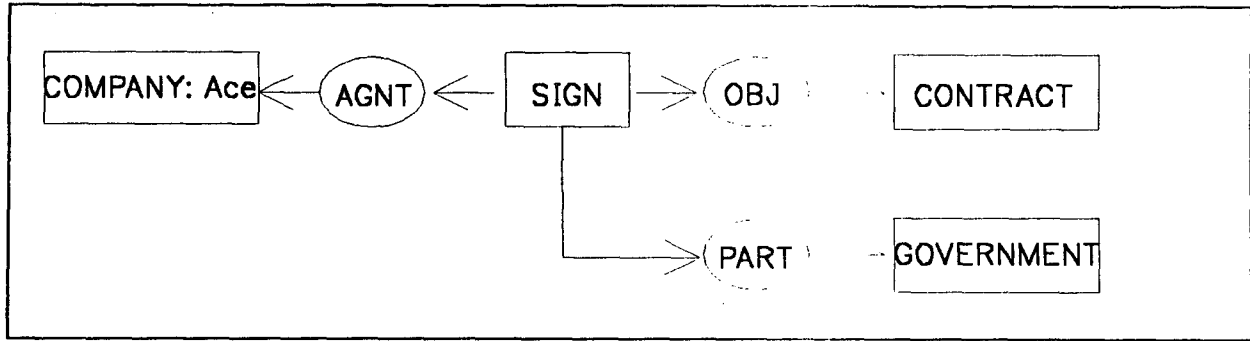


Figure 2. Conceptual graph for the sentence "The ACE signs a contract with the government"

APPENDIX

CONCEPTUAL RELATION HIERARCHY.

This Appendix provides a list of the three conceptual relation hierarchies (role, complement and link) introduced in Section 3. For each relation type, it is provided:

1. The level number in the hierarchy.
2. The complete name.
3. The correspondent abbreviation.

For some of the lower level relation types, an example sentence is also given. In the sentence, the concepts linked by the relation are highlighted, and the relation is cited, if explicit. Bold characters are used for not terminal nodes of the hierarchy.

The set of conceptual relation has been derived by an analysis of Italian grammar cases (the term "case" is here intended as for [FIL68]) and by a careful study of examples found in the analyzed domain. The final set is a trade-off between two competing requirements:

1. A large number of conceptual relations improves the expressiveness of the representation model and allows a "fine" interpretation;
2. A small number of conceptual relations simplifies the task of semantic verification, i.e. to replace syntactic relations between words by conceptual relations between concepts.

Link relations

1. **LINK (LINK)**
2. **HIERARCHY (HIER)**
 3. POSSESSION (POSS) The *house of John*
 3. SOCIAL_RELATION (SOC_REL) The *mother of. John*
 3. KIND_OF (KIND_OF) The *minister of the Interiors*
2. **COMPARISON (COMP)**
 3. MAJORITY (MAJ) *He is nicer than me*
 3. MINORITY (MIN)
 3. EQUALITY (EQ)

3. SIMILARITY (SIMIL)
2. **ORDERING (ORD)**
 3. **TIME_SPACE_ORDERING (POS)**
 4. VICINITY (NEAR) The *house near the lake.*
 4. PRECEDENCE (BEFORE)
 4. ACCOMPANIMENT (ACCOM) *Mary went with John*
 4. SUPPORT (ON) The *book on the table*
 4. INCLUSION (IN)
 3. **LOGIC_ORDERING (LOGIC)**
 4. CONJUNCTION (AND) *I eat and drink.*
 4. DISJUNCTION (OR) *Either you or me.*
 4. CONTRAPPOSITION (OPPOSITE)
 3. **NUMERIC_ORDERING (NUMERIC)**
 4. ENUMERATION (ENUM) *Five political parties*
 4. PARTITION (PARTITION) *Two of us*
 4. ADDITION (ADD) He owns a *pen and also a book.*

Complement relations

1. **COMPLEMENT (COMPL)**
2. **OCCURRENCE (OCCURR)**
 3. **PLACE (PLACE)**
 4. STATUS_IN (STAT_IN) I *live in Roma*
 4. **MOVE (MOVE)**
 5. MOVE_TO (DEST)
 5. MOVE_TROUGH (PATH)
 5. MOVE_IN (MOVE_IN)
 5. MOVE_FROM (SOURCE)
 3. **TIME (TIME)**
 4. DETERMINED TIME (PTIME) I *arrived at five*
 4. TIME LENGTH (TLENGTH) The *movie lasted for three hours*
 4. STARTING TIME (START) The skyscraper *was built since 1940*
 4. ENDING TIME (END)
 4. PHASE (PHASE)
 3. **CONTEXT (CONTEXT)**
 4. STATEMENT (STATEMENT) I *will surely come*
 4. POSSIBILITY (POSSIBLE)
 4. NEGATION (NOT)
 4. QUERY (QUERY)
 4. BELIEF (BELIEF) I *think that she will arrive*
 3. **QUALITY (QUALITY)**
 3. **QUANTITY (QUANTITY)**
 3. **INITIAL_VALUE (IVAL)** The shares *increased their value from 1000 dollars*
 3. **FINAL_VALUE (FVAL)** *to 1500*
2. **STRUCTURE (STRUCT)**
3. **SUBSTANCE (SUBST)**

- 4. MATTER (MATTER) *Wooden window*
- 4. ARGUMENT (ARG)
- 4. PART_OF (PART_OF) *John's arm.*
- 3. **SHAPE (SHAPE)**
- 4. CHARACTERISTIC (CHRC) *John is nice.*
- 4. **MEASURE (MEAS)**
- 5. AGE (AGE)
- 5. WEIGHT (WEIGHT)
- 5. EXTENSION (EXTEN) *A five feet man*
- 5. LIMITATION (LIMIT) *She is good at mathematics.*
- 5. PRICE (PRICE)

Role relations

1. **ROLE (ROLE)**

2. **HUMAN_ROLES (HUM_ROL)**

- 3. AGENT (AGNT) *The escape of the enemies*
- 3. PARTICIPANT (PART) *John flies to Roma.*
- 3. INITIATOR (INIT) *John boils eggs.*
- 3. PRODUCER (PRODUCER) *John's advise*
- 3. EXPERIENCER (EXPER) *John is cold.*
- 3. BENEFIT (BENEFIT) *Parents sacrifice themselves to the sons.*
- 3. DISADVANTAGE (DISADV)
- 3. PATIENT (PATIENT) *Mary loves John*
- 3. RECIPIENT (RCPT) *I give an apple to him.*

2. **EVENT_ROLES (EV_ROL)**

- 3. CAUSE (CAUSE) *He shivers with cold.*
- 3. MEANS (MEANS) *Profits increase investments*
- 3. PURPOSE (PURPOSE)
- 3. CONDITION (COND) *If you come then you will enjoy.*
- 3. RESULT (RESULT) *He was condemned to damages.*

2. **OBJECT_ROLES (OB_ROL)**

- 3. INSTRUMENT (INST) *The key opensthe door.*
- 3. SUBJECT (SUBJ) *The ball rolls.*
- 3. OBJECT (OBJ) *John eats the apple.*

REFERENCES

- [ANT87] Antonacci F., Russo M. **Three steps towards natural language understanding : morphology, morphosyntax, syntax.** submitted 1987
- [BRA79] Brachman P. **On the Epistemological Status of Semantic networks** in *Associative Networks: Representation and use of Knowledge by Computers*, Academic Press, N.Y. 1979
- [DEJ79] De Jong G.F. **Skimming stories in real time: An experiment in integrated understanding.** *Technical Rept. 158, Yale University, Dept. of Computer Science, New Heaven, CT*, 1979
- [FIL68] Fillmore **The case for case** *Universal in Linguistic Theory*, Bach & Harms eds., New York 1968
- [GRI76] Griffith R. **Information Structures** IBM St. Jose, 1976.
- [HEI86] Heidorn G.E. **Augmented Phrase Structure Grammar.** in *Theoretical Issues in Natural Language Processing*, Shank and Nash-Webber eds, Association for Computational Linguistics, 1975
- [HEI86] Heidorn G.E. **PNLP: The Programming Language for Natural Language Processing.** forthcoming
- [LEB83] Lebowitz M., **Researcher: an overview.** *Proc. of AAAI Conference*, 1983.
- [LEH83] Lehnert W.G., Dyer M.G., Johnson P.N., Yang C.J., Harley S. **BORIS- An Experiment in In-Depth Understanding of Narratives.** *Artificial Intelligence, Vol 20*, 1983
- [LEU85] Leuzzi S., Russo M. **Un analizzatore morfologico della lingua Italiana.** *GULP Conference*, Genova 1986
- [MIN75] Minsky M. **A framework for representing Knowledge** in *Psychology for Computer Vision*, Winston, 1975.
- [PAZ87] M.T. Paziienza, P. Velardi **Pragmatic Knowledge on Word Uses for Semantic Analysis of Texts** in *Knowledge Representation with Conceptual Graphs edited by John Sowa, Addison Wesley, to appear*
- [RIE79] Rieger C., Small S. **Word expert parsing.** *IJCAI*, 1979.
- [SHI72] Shank R.C. **Conceptual Dependency: a theory of natural language understanding.** *Cognitive Psychology, vol 3* 1972
- [SHI77] Shank R., Abelson R, **Scripts, Plans, Goals and Understanding.** *L. Erlbaum Associates*, 1977
- [SOW84] Sowa, John F. **Conceptual structures: Information Processing in Mind and Machine.** *Addison-Wesley, Reading*, 1984
- [SOW86] Sowa, John F. **Using a lexicon of canonical graphs in a conceptual parser.** *Computational Linguistics*, forthcoming.
- [VEL87] P. Velardi, M.T. Paziienza, M. De' Giovanetti **Utterance Generation from Conceptual Graphs** submitted
- [WIL73] Y. A. Wilks **Preference Semantics** *Memoranda from the Artificial Intelligence Laboratory, MIT* 1973