Lean Semantic Interpretation

Martin Romacker KatjaMarkert UdoHahn
Text Understanding Lab, Computational Linguistics Division
Freiburg University
D-79085 Freiburg, Germany
http://www.coling.uni-freiburg.de

Abstract

We introduce two abstraction mechanisms for streamlining the process of semantic interpretation. Configurational descriptions of dependency graphs increase the linguistic generality of interpretation schemata, while interfacing them to lexical and conceptual inheritance hierarchies reduces the amount and complexity of semantic specifications.

1 Introduction

Natural language processing methodology has matured in the recent years considering the design of grammar formalisms and parsing algorithms. The development of computationally feasible semantic theories and, in particular, their linkage to the grammar level by an adequate semantic interface has not witnessed a comparable level of consolidation. So, only a low degree of consensus has emerged concerning the formulation of semantic interpretation rules by which syntactic structures are mapped onto semantic (or conceptual) representations.

Hie lack of a mature and commonly shared methodology for semantic interpretation is not only deplorable from a theoretical standpoint but has immediate practical consequences for the design of NLP systems dealing with the diversity of real-world input on a larger scale. In such an environment, rule sets tend to be designed and updated in a less than systematic manna- to accomodate to the many complex phenomena in the language data, rule specifications confound different description layers, have heterogeneous formats and sometimes even ad hoc extensions— in the end, they become nonportable. Often this also leads to an unwieldy growth of the number of rules for large-scale NLP systems. As a result, the rules' compatibility, mutual interactions, side effects, order constraints, etc. are likely to get out of hand.

In order to avoid these negative effects, we introduce two abstraction mechanisms by which the process of semantic interpretation can be streamlined. The first abstraction aims at increasing the linguistic generality of descriptions for semantic interpretation. The criteria we use address *configurations* within dependency graphs rather than hook on particular language phenomena. These configurations have a natural graph-theoretical reading in terms of "minimal" connected subgraphs of a syntactic dependency *graph.* This way, we are able to cover a variety of linguistic phenomena by instantiation of few and general interpretation *schemata. The* second abstraction relates to the way how these schemata interact with grammar and domain knowledge. By interfacing them to lexical and conceptual *inheritance hierarchies,* we further increase descriptional economy and supply a parsimonious semantic interpretation system.

Our methodology crucially depends on a strict separation between linguistic grammar knowledge and conceptual domain knowledge, and their linkage via a lean semantic interface. Granting such an organization of knowledge, portability of semantic interpretation schemata across different domains and applications is made feasible — a feature of advanced system design that was hard to achieve so far.

2 Framework for Semantic Interpretation

We now sketch the knowledge sources required for semantic interpretation. *Grammatical knowledge* for syntactic analysis is based on a fully lexicalized dependency grammar [Hahn *et al.*$_y$ 1994]. A dependency grammar captures binary constraints between a syntactic head (e.g., a noun) and one of its possible modifiers (e.g., a determiner or an adjective). Our preference for dependency structures is motivated, among other things, by the observation that the correspondence of dependency relations (between lexical items) to conceptual relations (between the concepts they denote) is much closer than for any constituent-based grammar [Hajicova, 1987]. Hence, a dependency-based approach eases the description of the regularities underlying semantic interpretation.

In the dependency framework we have chosen, lexeme specifications form the leaf nodes of a lexicon tree, which are further abstracted in terms of word class specifications at different levels of generality. This leads to a word class hierarchy, which consists of word class names $\mathcal{W}$:= {VERB, VERBTRANS, DET, ARTICLE, ...} and a subsumption relation $isa_{\mathcal{W}}$ = {(VERBTRANS, VERB), (ARTICLE, DET), ...} $\subset \mathcal{W} \times \mathcal{W}$. Inheritance of grammatical knowledge is based on the idea that constraints are attached to the most general word classes to which they apply, leaving room for more and more specific (possibly, even idiosyncratic) grammatical specifications when one descends this hierarchy.

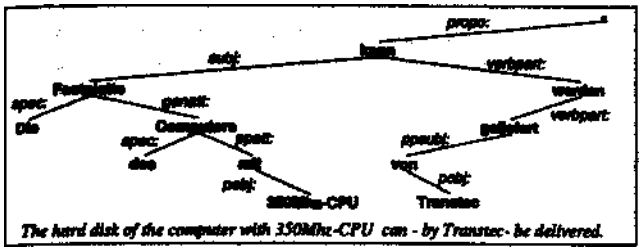

Figure 1: A Sample Dependency Graph

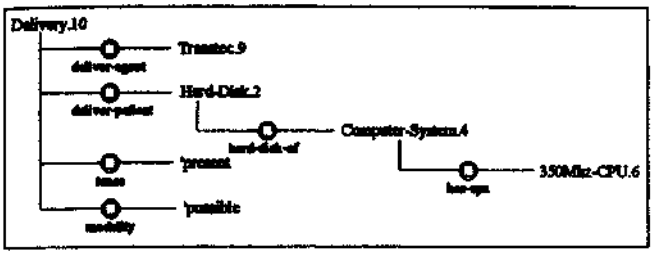

Figure 2: The Corresponding Concept Graph

In order to establish a dependency relation $\mathcal{D}$:= {specifier, *subject, dir-object,...*} between a head and a modifier, the corresponding constraints on word order, compatibility of morphosyntactic features as well as semantic criteria have to be fulfilled. Fig. 1 depicts a sample dependency graph in which word nodes are given in bold face and dependency relations are indicated by labeled edges. For example, the syntactic head *"Festplatte" (hard disk)* governs its modifier *"Computers"* via the *gen[itive]att[ribute]* dependency relation.

Conceptual knowledge is expressed in terms of a KL-ONE-like knowledge representation language [Woods and Schmolze, 1992]. This choice is motivated by the requirement to express domain knowledge in terms of a formally sound representation language. A domain ontology consists of a set of concept names $\mathcal{F}$:= {COMPANY, HARD-DISK, ...} and a subsumption relation $isa_{\mathcal{F}}$ = {(HARD-DISK, STORAGEDEVICE), (TRANSTEC, COMPANY), ...} $\subset \mathcal{F} \times \mathcal{F}$. The set of relation names $\mathcal{R}$:= {HAS-PART, DELIVER-AGENT, ...} contains the labels of conceptual relations which are also organized in a subsumption hierarchy $isa_{\mathcal{R}}$ = {(HAS-HARD-DISK, HAS-PHYSICAL-PART), (HAS-PHYSICAL-PART, HAS-PART), ...}.[1] Concept names are assigned conceptual roles, taken from the repertoire of conceptual relations, as part of the concept definition process. Unless defined as being primitive, the terminological classifier computes *subsumption* relations between these concept definitions in order to generate a domain's concept hierarchy (similarly, for relations).

In our approach, the representation languages for semantics and domain knowledge coincide (for arguments supporting this view, cf. Allen [1993]). The basic mechanism for linking lexical items and conceptual entities proceeds as follows: Upon entering the parsing process, each lexical item *w* that has a conceptual correlate *C* in the domain knowledge

[1] All subsumption relations, *isaw, isa$_f$,* and *isan,* are considered to be transitive and reflexive.

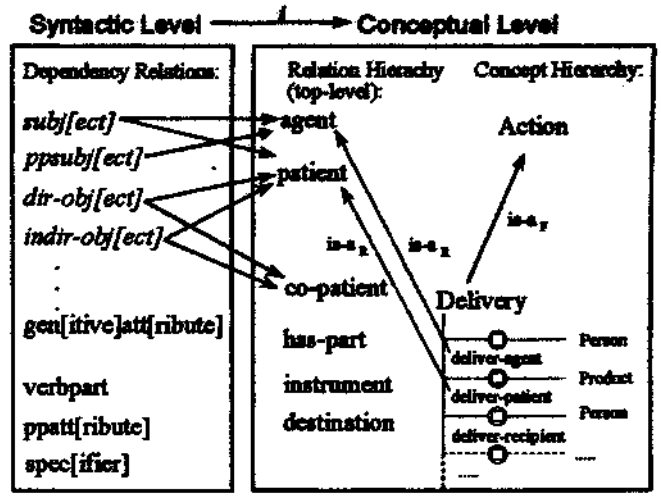

Figure 3: Relating Grammatical and Conceptual Knowledge

base, $w.C \in \mathcal{F}$ (mostly verbs, nouns and adjectives), gets immediately instantiated in the knowledge base, such that for any instance I_w, initially,[2] $type(I_w) = w.C$ holds (e.g., w = *"Festplatte"*, I_w = HARD-DISK.2, *W.C* = type(HARD-DISK.2) = HARD-DISK). In case several conceptual correlates exist (either due to homonymy or polysemy) each lexical ambiguity is processed independently.

Conceptual relations between instances are determined by different types of dependency relations that are established between their corresponding lexical items. *Semantic interpretation* mediates between both levels in a way as abstract and general as possible. First, *static semantic constraints* relate grammatical and conceptual knowledge via fixed mappings which take inheritance at both knowledge levels into account The illustration in Fig. 3 (left side) depicts a subset of the dependency relations contained in $\mathcal{D}$ at the syntactic level proper. Dependency relations that have a hard-wired mapping to conceptual relations are shown in italics. For instance, whenever the dependency relation *dir-obj[ect]* is established it must conceptually be interpreted in terms of PATIENT or CO-PATIENT including all their subrelations (e.g., DELIVER-PATIENT). *gen[itive]att[ributel* however, has no fixed conceptual counterpart as this dependency relation does not restrict conceptual interpretation at all.

Second, *semantic interpretation schemata* then *dynamically* incorporate these static constraints when translating a dependency graph (such as Fig. 1) into a corresponding conceptual representation (as depicted in Fig. 2). Schemata are high-level generalizations that abstract away from many linguistic details one encounters in specific dependency graphs (concrete lexical items, word classes, dependency relations). These abstract schemata get instantiated in the course of semantic interpretation by reference to the grammatical and conceptual hierarchies previously introduced. So, in our example, instantiating a general interpretation schema by con-

2Textual phenomena, figurative language, etc. might necessitate changes of this initial reference assignment, cf. Hahn *et al.* [1999].

crete lexical *materia* like 'harddisk"* and *"deliver"* automatically specializes high-level conceptual constraints like PATIENT (from the schema) to low-level ones such as DELIVER-PATIENT (from the domain knowledge) by exploiting the hierarchy of conceptual relations (cf. Fig. 3).

3 Semantically Interpretable Subgraphs

In die dependency parse tree from Fig. 1, we can distinguish lexical nodes that have a conceptual correlate (e.g., *"Festplatte" (hard disk), "geliefert" (delivered))* from others that do not have such a correlate (e.g., *"mit" (with), "von" (by)).* The basic configurations for semantic interpretation are based on this distinction:

- Direct linkage: If two word nodes with conceptual correlates are linked by a *single* dependency relation, a *direct* linkage is given. Such a subgraph can immediately be interpreted in terms of a corresponding conceptual relation. This is illustrated in Fig. 1 by the direct linkage between *"Festplatte" (hard disk)* and *"Computers"* via the *gen[itive]att[ribute]* relation, which gets mapped to the HARD-DISK-OF role linking the corresponding conceptual correlates, *viz,* HARD-DISK.2 and COMPUTER-SYSTEM.4, respectively (see Fig. 2). This interpretation uses knowledge about die conceptual correlates and die linking dependency relation, only.
- Mediated Linkage: If two word nodes with conceptual correlates are linked via a *series* of dependency relations and none of the intervening nodes have a conceptual correlate, a *mediated* linkage is given. Such a "minimal" subgraph can only indirectly be interpreted in terms of a conceptual relation. In this case, we include lexical information from intervening nodes in addition to the knowledge about the conceptual correlates and dependency relations. In Fig. 1 this is illustrated by the syntactic linkage between *"Computers"* and *"350Mhz-Cpu"* via the intervening node *"mit" (with)* and the *ppatt[ribute]* and *pobj[ect]* relations, the result of which is a conceptual linkage between COMPUTER-SYSTEM.4 and 350MHZ-CPU.6 via the relation HAS-CPU in Fig. 2.

In order to increase the generality and to preserve the simplicity of semantic interpretation we introduce a generalization of the notion of dependency relation such that it incorporates direct as well as indirect linkage: Two content words (nouns, adjectives, adverbs or full verbs) stand in a *mediated syntactic relation,* if one can pass from one word to the other along the connecting edges of the dependency graph without traversing word nodes other than prepositions, modal or auxiliary verbs (i.e., elements of closed word classes). In Fig. *1, e.g.,the tuples("Festplatte", "Computers"), ("Computers", "350Mhz~Cpu"), ("Festplatte", "geliefert")* and *("geliefert", "Transtec")* stand in mediated syntactic relations, whereas, e.g., the tuple (*"Transtec ", "Festplatte ")* does not, since the connecting path contains *"geliefert" (delivered),* a content word.

This leads us to the following definition: *Let* w and w' be two content words in a sentence *S.* In addition, let $w_2, \ldots, w_{n-1} \in S$ $(n \geq 2)$ be prepositions, auxiliary or modal verbs, and let $w_1 := w$ and $w_n := w'$. Then we say that *w* and *w'* stand *in q mediated syntactic relation,* iff there exists an index $i \in \{1, \ldots, n\}$ so that the following two conditions hold:

1. w_i is modifier of w_{i+1} for $i \in \{1, \ldots, i\text{-}1\}$;
2. w_i is head of w_{i+1} for $i \in \{i, \ldots, n\text{-}1\}$.

We call such a series $w_1, \ldots, w_n$ a *semantically interpretable subgraph* of the dependency graph of *S.*

The definition of a mediated syntactic relation encompasses the notion of a direct linkage ($n := 2$, so that an empty set of intervening nodes emerges). The special cases $i := 1$ and $i := n$ yield an ascending and descending series of head-modifier relations, respectively.[3]

The cases of direct and mediated linkage are the two relevant configurational settings for *semantic interpretation.* Thus, this paper is concerned with the translation of dependency graphs, like the one in Fig. 1, to a concept graph as in Fig. 2. The translation is achieved in a strictly compositional way, i.e., by linking the conceptual representations of the semantically interpretable subgraphs within the entire dependency graph.

We do not, however, consider *conceptual interpretation,* the mechanisms of which reside at the conceptual level only and, therefore, are merely indirectly affected by syntactic structures. As an example of such an inference consider Fig. 4, with the DELIVERS relation linking TRANSTEC.9, a hardware supplier, and HARD-DISK.2. Note that the corresponding lexical items, *"Transtec"* and *"Festplatte",* are not linked via a mediated syntactic relation in Fig. 1. Hence, we may clearly discern semantic interpretation, as the interpretation of semantically interpretable subgraphs, and conceptual interpretation, where the interpretation of relationships among *different* subgraphs come into play.

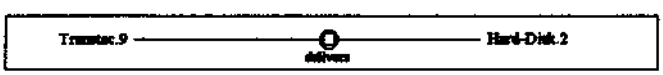

Figure 4: A Sample Conceptual Interpretation

4 A Model of Lean Semantic Interpretation

In the following, we shall describe a model of semantic interpretation that seamlessly integrates the processing of dependency relations either directly or indirectly linking conceptually interpretable word nodes. We distinguish two levels of semantic interpretation —first, static constraints for semantic interpretation derived from mapping dependency relations to conceptual roles, and, second, a search of the knowledge base when conducting the semantic interpretation, which dynamically takes these static constraints into account.

[3]Note that not every such graph may arise in a special dependeney grammar. This need not concern us in this paper.

Static Constraints on Semantic Interpretation, interpretation procedures for semantically interpretable subgraphs may inherit restrictions from the type of dependency relations (or even from the lexical material) occurring in these subgraphs. Constraint knowledge from the grammar level comes in two varieties, viz. via a positive list, D_+^{lexval}, and a negative D_-^{lexval}, of *dependency relations,* from which admitted as well as excluded *conceptual relati*R_+s, and H_, respectively, are derived by a simple static symbol mapping.

Knowledge about D_+^{lexval} and D_-^{lexval} is part of the valency specifications. It is encoded at the level of word dasses $\mathcal{W}$, such that *lexvai* $\in \mathcal{W} \times \mathcal{D}$, and, thereby, it is inherited by all subsumed lexical instances. For instance, the word class of transitive verbs, VERBTRANS $\in \mathcal{W}$, defines for its subject valenc $D_+^{(verbtrans,\ subject)} := \{subject\}$ and $D_-^{(verbtrans,\ subject)} := \emptyset$; for an optional prepositional phrase valency, *ppopt,* we require $D_+^{(verbtrans,\ ppopt)} := \emptyset$ and $D_-^{(verbtrans,\ ppopt)} := \{subject, dir\text{-}object, indir\text{-}object\}$. We may then distinguish three basic cases:

1. Knowledge available from syntax *determines* the semantic interpretation, if $D_+^{lexval} \neq \emptyset$ and $D_-^{lexval} = \emptyset$ (e.g., the subject of a verb).
2. Knowledge available from syntax *restricts* the semantic interpretation, if $D_+^{lexval} = \emptyset$ and $D_-^{lexval} \neq \emptyset$ (e.g., most prepositional phrases).
3. If $D_+^{lexval} = \emptyset$ and $D_-^{lexval} = \emptyset$, no syntactic constraints apply and semantic interpretation proceeds *entirely concept-driven,* i.e., it relies on the domain knowledge only (e.g., for genitive attributes).[4]

In order to transfer syntactic constraints to the conceptual level, we define $i: \mathcal{D} \rightarrow 2^{\mathcal{R}}$, a mapping from dependency relations onto sets of conceptual relations. This mapping generalizes the illustration depicted in Figure 3 (e.g., *((subject)* := {AGENT, PATIENT}). For dependency relations r that cannot be linked a priori to a conceptual relation (e.g. *gen{itive]att[ribute]*), we $i(r) := \emptyset$. The conceptual restrictions, R_+ and R_-, must be computed D_+^{lexval} and D_-^{lexval}, respectively, by applying the interpretation functus to $R_+ := \{y \mid x \in D_+^{lexval} \wedge y \in i(x)\}$ and $R_- := \{y \mid x \in D_-^{lexval} \wedge y \in i(x)\}$.

Basic Schema for Dynamic Semantic Interpretation. Semantic interpretation is done via a search in the domain knowledge base which takes the just mentioned constraints into account. Two sorts of knowledge have to be combined—first, a pair of concepts for which a connecting relation path has to be determined; second, conceptual constraints on the kinds of permitted and excluded conceptual relations when connected relations are being computed. The first constraint type incorporates the content words linked at the dependency level within the semantically interpretable subgraph, the tetter accounts for the particular dependency relation(s)betwi^ them. In technical tems, schema (1) describes a mapping from the conceptual con$h.C_{from}$ and $m.C_{to}$, in $\mathcal{F}$ of the two syntactically linked lexical items, *h* and *m,* respectively, to connected relation pathR_{con}.

$$si : \begin{cases} \mathcal{F} \times 2^{\mathcal{R}} \times 2^{\mathcal{R}} \times \mathcal{F} \rightarrow 2^{R_{con}} \\ (C_{from},\ R_+,\ R_-,\ C_{to}) \mapsto \widetilde{R_{con}} \end{cases} \quad (1)$$

A connected relation path $rel_{con} \in R_{con}$ is defined by:

$$rel_{con}((r_1, ..., r_n)) :\Leftrightarrow \forall i \in \{1, ..., n-1\} : isa_{\mathcal{F}}(type(range(r_i)), type(domain(r_{i+1})))$$

A relation path is called *connected,* if for all its n constituent relations the concept type of the domain of the relation r_{i+1} umes the concept type of the range of the relation r_i.[5]

In order to compute a semantic interpretation, st triggers a search through the concept graph of the domain knowledge base and identifies all connected relation paths C_{from} to C_{to}. Due to potential conceptual ambiguities in interpreting syntactic relations more than one such path can exist (hence, we map to the power set oR_{con}).

As a filter for constraining connectivity, *si* takes into consideration all conceptual re$R_+ \subset \mathcal{R}$ a priori permitted for semantic interpretation, as well as all relations $R_- \subset \mathcal{R}$ a priori excluded from semantic interpretation. Both of them reflect the constraints set up by particular dependency relations or non-content words figuring as lexical relators of content words (for examples, cf. Section 5). Thus, $rel \in \widetilde{R_{con}}$ holds, if *rel* is a connected relation path C_{from} to C_{to}, obeying the restrictions imposed R_+ and R_-.

If the function *si* returns the empty set (i.e., no valid interpretation can be computed), no dependency relation will be established. Otherwise, for all resulting relation path$\mathbf{REL}_i \in \widetilde{R_{con}}$ an assertional axiom is added to the knowledge base by asserting the $(h.C_{from}\ \mathbf{REL}_i\ m.C_{to})$ where REL* denotes the i^{th} reading. If $i > 1$, conceptual ambiguities occur (not an issue here).

To map a concept definition *C* against the constraints imposed bR_+ and R_-, we define the function *get-rcles(C)* =: *CR,* which returns the set of all conceptual roles *CR* associated v $get\text{-}roles(C_{from})$ extracts the roles that are used as starting points for the path search. As, for ease and generality of specification, R+ and R_ consist of the most general conceptual relations only, the concrete conceptual roles *CR* and the general ones in *R+* and *R_* may not always be compatible. Prior to semantic interpretation, we therefore expand R_+ and R_- into their transitive closures, incorporating all their subrelations in the relation hierarchy. Thus, $R_+^* := \{ r^* \in \mathcal{R} \mid \exists\ r \in R_+ : r^*\ isa_{\mathcal{R}}\ r \}$. Correspondingly, R_-^* is defined. *R+* restricts the search to relations contained in *CR* $\cap R_+^*$, *iff* R_+ is not empty (otherwise, all *CR* are allowed), whereas R_- allows only for relations in $CR \setminus R_-^*$.

[4] We have currently no empirical evidence for the fourth possible case, where $D_+^{lexval} \neq \emptyset$ and $D_-^{lexval} \neq \emptyset$.

[5] A number of additional constraints must hold, e.g., composed relations must be acyclic (cf. Markert ami Hahn [1997]).

5 Lean Semantic Interpretation at Work

Whenever a semantically interpretable subgraph is complete semantic interpretation is carried out immediately. Several configurations of these semantically interpretable subgraphs will be discussed subsequently. We start from the interpretation of direct linkage, and then turn to mediated linkage patterns by considering increasingly complex configurations in dependency graphs as given by prepositional phrases and passive clauses.

Interpreting direct linkage. Consider a fragment of our sample sentence, *"die Festplatte des Computers" (the hard disk of the computer)* in Fig. 1. *"Festplatte"* and *"Computers"* are directly linked by the dependency relation *gen[itive]-att[ribute]*$_y$ *"Festplatte"* being the syntactic head of its genitive modifier *"Computers"*. Both have conceptual correlates, whose type is given by type(HARD-DiSK.2) = HARD-DISK and t^pe(CoMPUTER-SYSTEM.4) = COMPUTER-SYSTEM. To relate the two instances conceptually, a slight specialization of the basic semantic interpretation schema (1), the direct linkage schema (2), will be applied (substituting C_{from} by C_{head} and C_{to} by C_{mod}).

$$si_{dir} : (C_{head}\ ,\ R_+\ ,\ R_-\ ,\ C_{mod}) \mapsto \widetilde{R_{con}} \qquad (2)$$

"Festplatte" belongs to the word class of nouns. It inherits $D_+^{\langle noun,\ genatt\rangle} := \emptyset$ and $D_-^{\langle noun,\ genatt\rangle} := \emptyset$. Thus, syntax does not at all restrict the search for valid conceptual relations, i.e., no static constraints apply. Hence, $R_+ = \emptyset$ and $R_- = \emptyset$. In order to dynamically incorporate constraints the function get-roles(HARD-DISK) extracts all roles related to HARD-DISK at the domain knowledge level (e.g., HAS-ACCESS-TIME, HAS-NOISE-LEVEL, HARD-DISK-OF, etc.). It is iteratively checked for each of these roles whether die conceptual correlate of the modifier *"Computers"*, i.e., COMPUTER-SYSTEM, is a legal filler (this is only the case for HARD-DISK-OF, cf. Kg. 2).

Interpreting mediated linkage. When interpreting mediated syntactic relations, additional information about the intervening nodes becomes available such that further static constraints are imposed on R_+ (and R_-) in terms of a list $R_{lex} \subset \mathcal{R}$ of permitted conceptual relations, which is specified at the lexeme level. Note that R_{lex} relates to closed-class items only, so the number of specifications required can be kept manageable. Also, such a specification can be done, once and for all, in a *domain-independent* way.

In our example (cf. Fig. 1), *"Festplatte" (hard disk)* and *"geliefert" (delivered)* are linked by a mediating modal verb *("kann")* and a passive auxiliary *("werden")*. The semantic interpretation schema for passive auxiliaries (3) addresses the concept type of the instance for their syntactic *subj[ect]*$_f$ $C_{subj} = type(\mathbf{I}_{subj})$ (i.e., HARD-DISK), and that for their *verbpart*, $C_{verbpart} = type(\mathbf{I}_{vrbpart})$ (i.e., DELIVERY). The relation between these two, however, is determined by $R_{passaux} :=$ {PATIENT, CO-PATIENT}, constraint knowledge which resides in die lexeme specification for *"werden" (be)* as passive auxiliary.

$$si_{aux} : (C_{verbpart}\ ,\ R_{passaux}\ ,\ \emptyset\ ,\ C_{subj}) \mapsto \widetilde{R_{con}} \qquad (3)$$

This leads to si_{aux}(DELIVERY, {PATIENT, CO-PATIENT}, $\emptyset$, HARD-DISK) which yields the conceptual relation DELIVER-PATIENT (cf. Fig. 3), since HARD-DISK is subsumed by PRODUCT and, thus, a legal filler of DELIVER-PATIENT $\in R^*_{passaux}$.

We will now turn to the attachment of prepositional phrases. Unlike conceptually neutral auxiliaries, prepositions serve as relators carrying conceptual constraints for the corresponding instances of their syntactic head and modifier. The "meaning" of a preposition is encoded by a set $R_{Prep} \subset \mathcal{R}$, for each preposition *Prep,* holding all permitted relations in terms of high-level conceptual relations. As an example, consider *"von" (by, from, of),* with $R_{von} :=$ {AGENT, SOURCE, STARTING-TIME-POINT, HAS-DEGREE …}.

In order to infer a valid semantic interpretation for each semantically interpretable subgraph containing a preposition a precompilation step for integrating static constraints from different sources is carried out These static constraints arise from the preposition itself and from the syntactic head of the preposition (e.g., *"geliefert" (delivered)* governs the preposition *"von" (by)* via the *ppsubj[ect]* dependency relation). The information provided by the syntactic relation between the preposition and its head is exploited such that the positive list D_+^{lexval} and negative D_-^{lexval} of the preposition's head are consulted prior to a semantic check in the knowledge base. By applying the function *i* to all elements contained in D_+^{lexval}, a new set $P_+ := \{y \mid x \in D_+^{lexval} \wedge y \in i(x)\}$ containing all conceptually permitted relations is created. By analogy, the corresponding set of conceptual relations P_- is excluded from interpretation. Expanding appropriately, we get $P_+^* := \{\, p^* \in \mathcal{R} \mid \exists\, p \in P_+ : p^*\ isa_{\mathcal{R}}\ p \,\}$ (correspondingly, we deP_-^*). We say for non-empty P_+: If $(R^*_{Prep} \setminus P^*_-) \cap P^*_+ = \emptyset$ (and, similarly, for empty P_+: If $(R^*_{Prep} \setminus P^*_-) = \emptyset$), then further semantic interpretation is obsolete. This is due to the fact that the static constraints given by the syntax (derived by the interpretation function % applied to D_+^{lexval} and D_-^{lexval}, respectively) are incompatible with the set of constraints for the preposition, R_{Prep}. Else, we determine $R_+ := (R^*_{Prep} \setminus P^*_-) \cap P^*_+$ for non-empty P_+ and $R_+ := (R^*_{Prep} \setminus P^*_-)$ for empty P_+, and define a specialized interpretation schema for prepositions si_{prep}:

$$si_{prep} : (C_{head}\ ,\ R_+\ ,\ \emptyset\ ,\ C_{mod}) \mapsto \widetilde{R_{con}} \qquad (4)$$

As far as the example *"geliefert von Transtec" (delivered by Transtec)* in Fig. 1 is concerned, the syntactic restrictions given bv the syntactic head of the preposition *("geliefert")* are $D_+^{\langle verbpartpass,\ ppsubj\rangle} := \{ppsubj\}$ and $D_-^{\langle verbpartpass,\ ppsubj\rangle} := \emptyset$ for the *ppsubj* valency, which is specified at the word class level for passive participles *verbpartpass* like *"geliefert" (delivered)*. Since syntactic constraints are encountered, the precompilation step becomes necessary in order to compute P_+^* and P_-^*. Since *ppsubj[ect]* statically maps to AGENT (cf. Fig. 3) and AGENT is also contained in R_{von}, AGENT and all its subrelations are contained in R_+ (i.e., the intersection of P_+^* and R^*_{von}). Since

$R_+ \neq \emptyset$ after integrating all static constraints, the semantic interpretation schema for prepmitions gets instantiated with li^numr, {AGENT*}, , TRANSTBC).[6] The roles to be considered are thus the roles contained in the intersection of the concept roles of DELIVERY (extracted by *get*-roJe$(DBLIVBRY) with R+. Given the underlying ontology, this set of relations melt down to a single element, DELIVER-AGENT isa_R AGENT. Thus, satisfaction of dynamic constraints reduces to the check whether TRANSTBC is a legal filler of DELIVER-AGENT. Since TRANSTBC is subsumed by LEGAL-PERSON which IS-A PERSON, the sortal constraints are met and TRANSTEC.9 is asserted to be the DELIVER-AGENT of DELIVERY. 10 (cf. Fig. 2).

6 Evaluation of Semantic Interpretation

In this section, we want to discuss, for a particular type of language phenomenon, the adequacy of our approach in the light of concrete language data taken from the corpora we work with. This part of the enterprise, the empirical assessment of semantic interpretation, is almost entirely neglected in the literature (for a notable exception, cf. Bean *et al.* [1998]).

The semantic interpretation task has to be clearly distinguished from the information extraction task and its standard evaluation setting [MUC-6,1995]. In the IE task, a *subset* of the templates from the entire domain is selected into which information from the texts are mapped. Also, the design of these templates focus a priori on particularly *interesting* facets (roles, in our terminology), so that an IE system does not have to deal with the full range of qualifications that can be attributed even to a relevant concept that might occur. The semantic interpretation task, however, is far less constrained as we evaluate the adequacy of the conceptual representation structures relating to the *entire* domain of discourse, with *all* qualifications mentioned in a text.

This increased complexity of the task leads to many unsolved methodological questions (for a discussion, cf. Friedman and Hripcsak [1998]). We may illustrate this claim within the framework of our approach. Basically, a triple division of test scenarios have to be distinguished. The first class relates to checks whether each static constraint, effected by the mapping from a single dependency relation to one or more conceptual relations is valid. Second, one may investigate the appropriateness of the results from the search of the domain knowledge base, i.e., whether a relation between two concepts can be determined at all, and, if so, whether that relation (path) is adequate. Third, interactions between static constraints and dynamic constraint propagation may occur, as illustrated by the interpretation of prepositions.

In the small-scale evaluation study we performed, we started from a domain ontology that is divided into an upper generic part (containing about 1,100 concepts and relations) and various domain-specific parts. In the study we report on two specialized domains were dealt with — an information technology (IT) model and an ontology covering parts of anatomical medicine (MED) (each domain model, in addition, contributes 1,100 concepts and relations). OMT-responding lexeme entries in the lexicon provide linkages to the ontology. We also assume a correct parse to be deliveied for the semantic interpretation process.

We then took a random selection of 54 texts (comprising 18,500 words) from our two text corpora. For evaluation purposes we concentrated on the interpretation of genitives (direct linkage), prepositional phrase attachments and auxiliary as well as modal verbs (both variants of mediated linkage). In the following, we will focus on the discussion of the results from the semantic interpretation of genitives (cf. Table 1).

At first glance, the choice of genitives may appear somewhat trivial. From a syntactic point of view, genitives are directly linked and, indeed, constitute an easy case to deal with at the dependency level. From a conceptual perspective, however, they provide a real challenge. Since *no static* constraints are involved in the interpretation of genitives, an unconstrained search (apart from connectivity conditions) of the domain knowledge base is started. Hence, the main burden rests on the *dynamic* constraint processing part of semantic interpretation, i.e., the path finding procedure muddling through the complete domain knowledge base in order to select the adequate reading(s). Therefore, genitives make a strong case for the second test scenario mentioned above.

The following criteria guided the evaluation of genitives. We considered a semantic interpretation to be a *correct* one, if the conceptual relation between the two concepts involved was considered adequate by introspection (otherwise, *incorrect).* This qualification is not as subjective as it may sound, since we applied really strict conditions.[7] Correct interpretations were those that contain exactly one relation, as well as cases of ambiguities (up to three readings, the most), where the relation set contained the correct one. A special case of incorrectness, called *nil,* occurs when no relation path can be determined though the two concepts under scrutiny are contained in the domain knowledge base.

We also classified the cases where the system failed to produce an interpretation due to at least one concept specification missing (with respect to the two linked content words in a semantically interpretable subgraph). In those cases *without* interpretation, insufficient coverage of the *generic model* was contrasted with that of the two *domain models* and with cases in which concepts referred to *other domains,* e.g., fashion or food. Subareas that could neither be assigned to the generic model nor to particular domains were denoted by phrases re-

[6] Note that due to the precompilation step there is no need to expand R_+ to its transitive closure.

[7] The majority of cases were easy to judge. For instance, *"the infiltration of the stroma"* resulted in a correct reading - STROMA being the PATIENT of the INFILTRATION -, as well as in an incorrect one - being the AGENT of the INFILTRATION. Among the incorrect semantic interpretations we also categorized, e.g., the interpretation of the expression *"the prices of the manufacturers" in* terms of a conceptual linkage from PRICE via PRICE-OF to PRODUCT via HAS-MANUFACTURER to MANUFACTURER, since it did not account for the interpretation that MANUFACTURERS fix PRICES as part of their marketing stragegies. Correct interpretations always boiled down to evident cases, eg., HARD-DISK PART-OF COMPUTER.

	MED	TT
# texts	29	25
# words	4,300	14,200
recall	57%	31%
precision	97%	94%
# genitives...	100	147
... with interpretation	59(59%)	49 (33%)
	53 (53%)	28(19%)
.....correct (multiple readings)	4 (4%)	18(12%)
.....incorrect	0	3
nil	2	0
... without interpretation	41 (41%)	98 (67%)
.....domain model	23 (23%)	57(39%)
.....generic model	3	23
	0	4
.....abstracta	11	12
.....space	7	8
.....time	0	15
	1	17
.....evaluative	0	8
	0	1

Table 1: Empirical Results for the Interpretation of Genitives

ferring to *time* (e.g., "the beginning of the year"), *space* (e.g., "the surface of the storage medium"), and *abstract* notions (e.g., "the acceptance of IT technology"). These areas can be distinguished from *evaluative* expressions (e.g., "the advantages of plasma display") *and figurative* language, including idiomatic expressions (e.g., "the heart of the notebook").

We considered a total of almost 250 genitives in all these texts, from which about 59%/33% (MED/IT) received an interpretation. Of the total loss due to incomplete conceptual coverage, 56%/58% (23 of 41 genitives/57 of 98 genitives) can be attributed to insufficient coverage in the domain model. Only the remaining 44%/42% are due to the many other factors we have listed in Table 1.

The results we present here are just a preliminary sketch of what has to be done as part of a more serious evaluation. The concrete values we present, disappointing as they may be for recall (57%/31%), encouraging, however, for precision (97%/94%), can only be interpreted relative to other data still lacking on a broader scale. One should keep in mind, however, that we neither narrowed semantic interpretation down to a very limited range of spatial relations in anatomy [Bean *et al,* 1998], nor did we bias the result by preselecting only those phrases that were already covered by our domain models [Gomez *et al.,* 1997], Judged from the poor figures of our recall data, there is no doubt, whatsoever, that conceptual coverage of the domain constitutes *the* bottleneck for any knowledge-based approach to NLR Sublanguage differences sore also mirrored in these data, since medical texts adhere more closely to well-established concept taxonomies than magazine articles in the IT domain.

In our evaluation, we certainly have only scratched the surface. Besides many open methodological questions related to the evaluation of conceptual structures, a lot of empirical material related to different syntactic structures still has to be considered. In our system, we currendy supply semantic interpretation schemata for declaratives, relatives, and passives at the clause level, complement subcategorization via PPs, auxiliaries, tenses at the verb phrase level, pre- and and postnominal modifiers at the noun phrase level, and anaphoric expressions. We currently do not cover control verbs, coordination and quantification.

7 Related Work

The standard way of deriving a semantic interpretation is to assign each syntactic rule one or more semantic interpretation rules (e.g., van Eijck and Moore [1992]), and to determine the meaning of the syntactic head from its constituents. There are no constraints on how to design and organize this rule set despite those that are implied by the choice of the semantic theory. In particular, abstraction mechanisms (going beyond the level of sortal taxonomies for semantic labels, cf., e.g., Creary and Pollard [1985]), such as property inheritance, defaults, are lacking. Accordingly, the number of rules increases rapidly and easily reaches orders of several hundreds in a real-world setting [Bean *et al,* 1998]. As an alternative, we provide a small set of interpretation schemata instead of assigning specific interpretation rules to each grammar item (in our case, single lexemes), and incorporate inheritance-based abstraction in the use of these schemata during the interpretation process in the knowledge base.

Sondheimer *et al* [1984] and Hirst [1988] treat semantic interpretation as a direct mapping from syntactic to conceptual representations. They also share with us the representation of domain knowledge using KL-ONE-style terminological languages, and, hence, they make heavy use of property inheritance (or typing) mechanisms. The main difference to our approach lies in the status of the semantic information rules. Sondheimer *et al* attach single interpretation rules to each *role {filler)* and, hence, have to provide utterly detailed specifications reflecting the idiosyncrasies of each semantically relevant (role) attachment. Property inheritance comes only into play when the selection of alternative semantic rules is constrained to the one(s) inherited from the most specific case frame. In a similar way, Hirst uses strong typing at the conceptual *object* level only, while we use it simultaneously at the grammar and the domain knowledge level for the processing of semantic schemata.

Charniak and Goldman [1988] and Jacobs [1991] already specify semantic *rules* in the context of inheritance hierarchies. So, they achieve a similar gain in generality as we do (e.g., Charniak and Goldman's rule for case relations corresponds to our schema (2)). We differ, however, in that we specify semantic interpretation *schemata* rather than *rules.* This additional level of generality becomes evident, e.g., in terms of schema (2), which also incorporates the interpretation of genitives (and many other direct linkage phenomena). Indeed, Charniak and Goldman have to supply an extra semantic interpretation rule for every syntactic phenomenon as

they base semantic interpretation on syntactic rules and not on abstract configurations in dependency graphs that cover more than one syntactic phenomenon. Jacobs [1991] goes even further and completely ties syntactic role specifications into conceptual ones. Such an approach, however, mixes knowledge levels at the cost of clean modularization.

8 Conclusions

We proposed a principled approach to the design of compact, yet highly expressive semantic interpretation schemata. They derive their power from two sources. First, the organization of grammar and domain knowledge, as well as semantic interpretation mechanisms, are based on inheritance principles. Second, interpretation schemata abstract from particular linguistic phenomena in terms of general configuration patterns in dependency graphs.

Underlying these design decisions is a strict separation of linguistic from conceptual knowledge. A clearly defined interface is provided which allows these specifications to make reference to fine-grained hierarchical knowledge, no matter whether it is of linguistic or conceptual origin. In addition, the interface is clearly divided into two levels. One makes use of static, high-level constraints supplied by the mapping of syntactic to conceptual roles or supplied as the meaning of closed word classes. The other uses these constraints in a dynamic search through a knowledge base, that is parametrized by few and simple schemata.

It should be clearly noted, however, that the power of this approach is, to a large degree, dependent on the fine granularity of the knowledge sources we incorporate, the domain knowledge base, in particular. Given such an environment, the formulation of the regularities at the semantic description level can be kept fairly general. The need for detailed ontologies is by far not restricted to semantic interpretation. Indeed, one cannot do without sophisticated domain representations for many other challenging understanding tasks such as the resolution of reference relations in texts [Hahn *et al,* 1999], the acquisition of concepts from texts [Hahn and Schnattinger, 1998], or the interpretation of evaluative assertions [Staab and Hahn, 1997].

Also since the number of schemata at the semantic description layer remains rather small, their execution is easy to trace and thus supports the maintenance of large-scale NLP systems. Hie high abstraction level provided by inheritance-based semantic specifications allows easy porting across different application domains. Our experience rests on reusing the set of semantic schemata once developed for the IT domain in the medical domain *without* further changes.

Acknowledgements. K. Maikert was a member of the Graduate Program *Human and Machine Intelligence* funded by DFG, while M. Romacker is supported by a grant from DFG (Ha 2097/5-1).

References

[Allen, 1993] James E Allen. Natural language, knowledge representation, and logical form. In M. Bates and R. M. Weischedel, editors, *Challenges in Natural Language Processing,* pages 146-175. Cambridge: Cambridge University Press, 1993.

[BesuketaL, 1998] C. A. Bean, T. C. Rindflesch, and C. A. Sneiderman. Automatic semantic interpretation of anatomic spatial relationships in clinical text. In *Proceedings of the 1998 AMIA Annual Fall Symposium,* pages 897-901. Hanley & Belfus, 1998.

[Charniak and Goldman, 1988] Eugene Charniak and Robert Goldman. A logic for semantic interpretation. *In Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics,* pages 87-94,1988.

[Creary and Pollaid, 1985] Lewis G. Creary and Caii J. Pollard A computational semantics for natural language. In *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics, pages* 172-179, 1985.

[Friedman and Hripcsak, 1998] C. Friedman and G. Hripcsak. Evaluating natural language processors in the clinical domain. *Methods of Information in Medicine,* 37(4/5):334-344,1998.

[Gomez *et al.,* 1997] F. Gomez, C. Segami, and R. Hull. Determining prepositional attachment, prepositional meaning, verb meaning and thematic roles. *Computational Intelligence,* 13(1): 1-31, 1997.

[Hahn and Schnattinger, 1998] Udo Hahn ami Klemens Schnattinger. Towards text knowledge engineering. In *AAAI'9MAAI'98 - Proceedings of the 15th National Conference on Artificial Intelligence & 10th Conf on Innovative Applications of Artificial Intelligence,* pages 524-531. AAAI Press & MTT Press, 1998.

[Hahn *et aL,* 1994] Udo Hahn, Susanne Schacht, awl Norbert Broker. Concurrent, object-oriented natural language parsing: the PARSETALK model. *International Journal of Human-Computer Studies,* 41(1/2): 179-222, 1994.

[Hahn *et aL,* 1999] Udo Hahn, Martin Romacker, and Stefan Schulz. Discourse structures in medical repents - watch out! The generation of referentially coherent and valid text knowledge bases in the MEDSYNDIKATE system. *International Journal of Medical Informatics,* 53(I):I-28,1999.

[Hajicova, 1987] E. Hajicova. Linguistic meaning as related to syntax and to semantic interpretation. In M. Nagao, editor, *Language and Artificial Intelligence.,* pages 327-351. North-Holland, 1987.

[Hirst, 1988] Graeme Hirst. Semantic interpretation and ambiguity. *Artificial Intelligence,* 34(2): 131-177,1988.

[Jacobs, 1991] Paul S. Jacobs. Integrating language and meaning in structured inheritance networks. In John F. Sowa, editor, *Principles of Semantic Networks. Explorations in the Representation of Knowledge,* pages 527-542. Morgan Kaufmann, 1991.

[Markert and Hahn, 1997] Katja Markert and Udo Hahn. On the interaction of metonymies and anaphora. In *IJCAI'97 - Proceedings of the 15th International Joint Conference on Artificial Intelligence,* pages 1010-1015. Morgan Kaufmann, 1997.

[MUC-6,1995] MUC-6. MUC-6 - *Proceedings of the 6th Message Understanding Conference.* Morgan Kaufmann, 1995.

[Sondheimer et al., 1984] Norman K. Sondheimer, Ralph M. Weischedel, and Robert J. Bobrow. Semantic interpretation using KL-ONE. In *COUNG'84 - Proceedings of the 10th International Conference on Computational Linguistics $ 22nd Annual Meeting of the ACL,* pages 101-107,1984.

[Staab and Hahn, 1997] Steffen Staab and Udo Hahn. 'Tall**, "good", "high" - compared to what? In *IJCAI'97 - Proceedings of the 15th International Joint Conference on Artificial Intelligence,* pages 996-1001. Morgan Kaufmann, 1997.

[van Eijck and Moore, 1992] Jan van Eijck and Robert C. Moore. Semantic rules for English. In Hiyan Alshawi, editor, *The Core Language Engine,* pages 83-115. MTT Press, 1992.

[Woods and Schmolze, 1992] W. Woods and J. Schmolze. The KL-ONE family. *Computers & Mathematics with Applications,* 23(2/5): 133-177, 1992.