Layer Structures and Conceptual Hierarchies in Semantic Representations for NLP

Hermann Helbig, Ingo Glöckner, and Rainer Osswald

Intelligent Information and Communication Systems
FernUniversität in Hagen, Germany

Abstract. Knowledge representation systems aiming at full natural language understanding need to cover a wide range of semantic phenomena including lexical ambiguities, coreference, modalities, counterfactuals, and generic sentences. In order to achieve this goal, we argue for a multidimensional view on the representation of natural language semantics. The proposed approach, which has been successfully applied to various NLP tasks including text retrieval and question answering, tries to keep the balance between expressiveness and manageability by introducing separate semantic layers for capturing dimensions such as facticity, degree of generalization, and determination of reference. Layer specifications are also used to express the distinction between categorical and situational knowledge and the encapsulation of knowledge needed e.g. for a proper modeling of propositional attitudes. The paper describes the role of these classificational means for natural language understanding, knowledge representation, and reasoning, and exemplifies their use in NLP applications.

1 Introduction

Although envisaged since the early days of automated natural language processing (NLP), there are currently only a few implemented systems that aim at a full semantic analysis of unrestricted natural language. One of the reasons for this situation may be the diversification in formal semantics with its highly elaborate but specialized theories focusing on specific semantic phenomena such as presuppositions, generics, pluralities or modalities (see e.g. [1]), whereas building language understanding systems calls for a uniform and concise formalism covering all aspects of natural language semantics up to a certain degree of granularity. A second reason may be the current dominance of shallow NLP even in areas where traditionally deep semantic analysis has been taken as a *sine qua non*. For instance, many approaches to open-domain textual question answering rest mainly on text retrieval techniques with no or little analysis of the underlying documents. However, this approach will fail if the relevant information is mentioned only once in the text collection, has no lexical overlap with the question and uses other syntactic constructions, or is distributed over several sentences linked by anaphora.

In this paper, we argue for a concept-centered representation of natural language semantics that employs a moderate amount of reification and represents semantic aspects like plurality or facticity by ontological features whenever appropriate. The proposed approach is explicated by a slightly simplified version of the *MultiNet* knowledge representation formalism, which is described in detail in [2]. MultiNet has been designed

A. Gelbukh (Ed.): CICLing 2008, LNCS 4919, pp. 171–182, 2008.

for representing the semantics of unregimented text (*universality criterion*) while being applicable at all stages of linguistic analysis, knowledge representation, inference processes, and natural language generation (*interoperability criterion*).

The MultiNet formalism has been employed for building a large semantically-based computational lexicon [3] as well as a syntactico-semantic parser [4] that generates MultiNet representations of natural language input. The lexicon and the parser are the core component of several NLP systems, ranging from question answering systems [5,6] over natural language interfaces [7] to systems for checking text readability [8].

2 Sorts, Relations, and Conceptual Hierarchies

This section briefly introduces part of the sortal and relational inventory of the MultiNet formalism in order to provide a basis for explaining the use of layer structures in Sections 3 and 4.

2.1 Ontological Sorts and Semantic Relations

The definition of a formalism for knowledge representation and natural language semantics, usable in all parts of an NLP system, requires a clear characterization of its basic categories. On the one hand, one needs a classification of the concepts to be dealt with (yielding the sorts of conceptual entities shortly described in this subsection). On the other hand, one has to define the fundamental relations and functions holding between these entities, using these sorts for domain and range specifications. The most basic means for conceptual classification used in MultiNet is a tree shaped hierarchy of *ontological sorts*, which can be regarded as the backbone of an upper ontology. Part of the MultiNet sort hierarchy is shown in Fig. 1 (see [2, Chap. 17] for the full hierarchy). The use of these sorts for specifying the signatures of relations shown in Table 1 explains some of the proposed distinctions. Consider, for example, the discernment of ordinary properties (*p*) vs. relational qualities (*rq*). Concepts of sort *rq*, like equivalent, are different from regular properties in that they can only qualify collections. Thus ordinary properties will be expressed by PROP and relational qualities by a relation PROPR governed by different axioms (see Table 2). The refinement of a sort into subsorts sometimes reflects differences in the inheritance patterns. As shown by axiom (b) in Table 2, gradable properties (sort gq) and total properties (sort tq) are distinguished for that reason. In contrast to upper domain ontologies like DOLCE [9], the hierachy of ontological sorts proposed here does not cover concepts like *sets*, *collections* or *aggregates*, because building pluralities and collections is possible across different sorts and should thus be seen as orthogonal to the basic sortal hierarchy. Therefore, plurality is expressed by an *ontological feature* as a separate classificational dimension (see Sect. 3.3).

Based on general requirements like universality, homogeneity and granularity (see [2, Chapt. 1]), to be fulfilled by a formalism for natural language semantics, MultiNet comes up with a system of more than hundred relations and functions for modeling relationships between conceptual entities [2]. These comprise basic structuring relations (for subordination and meronymy), argument roles (for describing participants of a situation), temporal and local relations (for describing spatio-temporal aspects) and finally

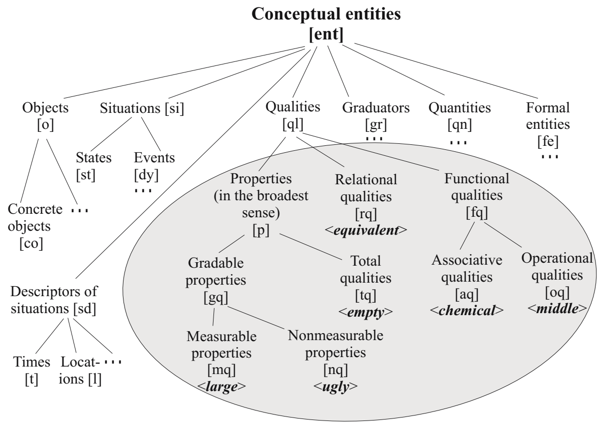

Fig. 1. Part of the MultiNet sort hierarchy (with emphasis on qualities)

intersituational relations (for describing causes, circumstantials and modal embeddings), see Table 1.[1] The commitment to a fixed catalogue of relations and functions fosters the long-term development of linguistic resources and knowledge bases (e.g. of the computational lexicon HaGenLex [3]), as well as the interoperability of applications [7].

Fig. 2 illustrates the use of these relations for describing the meaning of an example sentence, automatically generated by the WOCADI parser [4].[2] In particular, the example shows that all nodes which represent instances (i.e. $c1, c2, \dots$) are described in terms of the pre-defined relations and the concept constants (like *analyst*, *meeting*...) defined in the lexicon. This warrants a tight coupling between knowledge representation and the computational lexicon, on the one hand (homogeneity property), and a smooth interfacing between syntactic semantic analysis and following inference processes in a question-answering system on the other hand (interoperability property). The figure also displays the ontological features FACT, GENER, ETYPE of nodes and the characterization of edges explained in Sections 3 and 4.

2.2 Conceptual Hierarchies

There is a general consensus that the *subordination* of concepts is the most important relation governing ontologies and hierarchies of concepts. In Description Logics [10], this relation is extensionally interpreted as an inclusion of extensions and called subsumption. In our formalism, we assume a single subordination relation SUB for conceptual

[1] Some of the signature definitions given in the table make use of ontological features, which is indicated by the following notational conventions: The notation $\ddot{\sigma}$ restricts an argument to collections of sort σ. It is needed to describe the combinatorics of PROPR. The specification $\overline{\sigma}$ indicates generic, i.e. non-instantiated concepts of sort σ. It is used in the signature of SUB and SUBS to ensure that the second argument is a generic concept (see Sect. 3.2).

[2] Notice that some of the relationships, e.g. SUB$(c_1, \textit{analyst})$, are not displayed as directed edges but rather listed below the node name.

Table 1. Simplified description of relations used in this paper

Relation	Signature	Short Characteristics
AGT	$si \times co$	Agent
ATTR	$[o \cup l \cup t] \times at$	Specification of an attribute
CIRC	$si \times si$	Relation between situation and circumstance
COND	$si \times si$	Conditional relation
ELMT	$ent \times \ddot{ent}$	Membership (of extensions)
EXP	$si \times o$	Experiencer
HSIT	$si \times si$	Composition of situations
LOC	$co \times l$	Location of an object
MCONT	$si \times [si \cup o]$	Mental or informational content
MEXP	$st \times co$	Mental experiencer
ORIGM	$co \times co$	Relation of material origin
PARS	$co \times co$	Part-whole relation for concrete objects
PROP	$o \times p$	Relation between object and property
PROPR	$\ddot{o} \times qr$	Relation between plurality and relational property
SUB	$o \times \overline{o}$	Relation of conceptual subordination (for objects)
SUBM	$\ddot{ent} \times \ddot{ent}$	Subsumption (inclusion of extensions)
SUBS	$si \times \overline{si}$	Relation of conceptual subordination (for situations)
TEMP	$si \times [si \cup t]$	Relation of temporal containment
VAL	$at \times [qn \cup p \cup fe]$	Relation between attribute and value

Table 2. Selected axioms related to subordination

(a)	$\mathrm{SUB}(o_1, o_2) \wedge \mathrm{SUB}(o_2, o_3) \rightarrow \mathrm{SUB}(o_1, o_3)$
(b)*	$\mathrm{SUB}(o_1, o_2) \wedge \mathrm{PROP}(o_2, p) \wedge p \in \mathrm{tq} \rightarrow \mathrm{PROP}(o_1, p)$
(c)	$\mathrm{SUB}(o_1, o_2) \wedge \mathrm{ATTR}(o_2, a) \rightarrow \exists a'[\mathrm{SUB}(a', a) \wedge \mathrm{ATTR}(o_1, a')]$
(d)*	$\mathrm{SUB}(o_1, o_2) \wedge \mathrm{ORIGM}(o_2, s_2) \rightarrow \exists s_1[\mathrm{SUB}(s_1, s_2) \wedge \mathrm{ORIGM}(o_1, s_1)]$

objects, which we characterize axiomatically in order to avoid this assumption of extensional interpretation. We are well aware of the necessity to separate instantiation and specialization of concepts as pointed out by Brachman [11]. In our system, instantiation corresponds to a subordination relationship SUB(o_1, o_2) where o_1 is a specific entity and o_2 is generic. Specialization, by contrast, is modelled by a subordination relationship SUB(o_1, o_2) where both participants are generics. In this way, we cleanly distinguish the two flavours of subordination, while avoiding a duplication of axioms valid in both cases. Table 2 lists basic axioms related to subordination. (Axioms marked by an asterisk have default status only). Axiom (a) expresses the transitivity of SUB. Inheritance of properties is described by (b). The term $p \in \mathrm{tq}$ restricts applicability to total properties. This prevents a wrong conclusion that an object which is both a small animal and an ant, can be described as a small ant. Axiom (c) formalizes inheritance of attributes. For example, the attribute of eye color is inherited by all instances of *human*. Finally, (d) expresses inheritance of material origin: if ORIGM(*window*, *glass*), i.e. windows are made of glass, then a concrete window is also made of glass. From the SUB relation we distinguish another subordination relation (in MultiNet called SUBS) which carries

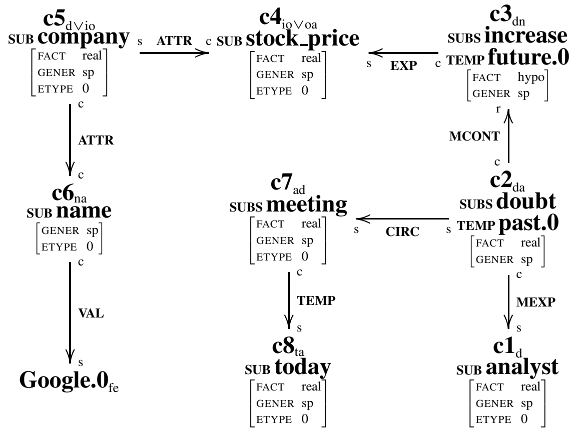

Fig. 2. MultiNet representation of the sentence "*On today's meeting, the analyst doubted that Google's stock price would increase*"

Table 3. Selected axioms related to meronymy relations

(a)*	$\mathrm{PARS}(k_1, k_2) \wedge \mathrm{PARS}(k_2, k_3) \rightarrow \mathrm{PARS}(k_1, k_3)$
(b)	$\mathrm{SUB}(d_1, d_2) \wedge \mathrm{PARS}(d_3, d_2) \rightarrow \exists d_4[\mathrm{SUB}(d_4, d_3) \wedge \mathrm{PARS}(d_4, d_1)]$
(c)*	$\mathrm{PARS}(k_1, k_2) \wedge \mathrm{LOC}(k_2, \ell) \rightarrow \mathrm{LOC}(k_1, \ell)$
(d)*	$\mathrm{PARS}(k_1, k_2) \wedge \mathrm{ORIGM}(k_2, s) \rightarrow \mathrm{ORIGM}(k_1, s)$
(e)	$\mathrm{ELMT}(k_1, k_2) \wedge \mathrm{ELMT}(k_2, k_3) \rightarrow \neg\, \mathrm{ELMT}(k_1, k_3)$
(f)	$\mathrm{HSIT}(k_1, k_2) \wedge \mathrm{HSIT}(k_2, k_3) \rightarrow \mathrm{HSIT}(k_1, k_3)$
(g)*	$\mathrm{HSIT}(s, p) \wedge \mathrm{LOC}(s, \ell) \rightarrow \mathrm{LOC}(p, \ell)$
(h)	$\mathrm{HSIT}(s, p) \wedge \mathrm{COND}(c, s) \rightarrow \mathrm{COND}(c, p)$
(i)	$\mathrm{HSIT}(s, p) \wedge \mathrm{CIRC}(e, p) \rightarrow \mathrm{CIRC}(e, s)$

the conceptual hierarchy for situational entities. This relation, for instance, connects the activities *go* and *move (oneself)*. The main characteristics of SUBS is the *inheritance of argument structure* of superconcepts by subconcepts.

Besides subsumption or conceptual subordination, *part-whole relations* traditionally play an important role in knowledge representation. The proper way of reasoning along part-whole hierarchies has been discussed extensively in the literature [12]. In particular, it has been observed that transitivity violations are often due to mixing up different types of part-whole relations. We thus discern several mereological relations which capture the different ways in which a part can be related to a whole, viz PARS (physical parts of concrete objects), ELMT (membership of an element in a collection), HSIT (relationship between a complex situation and its parts), and the loosely related SUBM (containment of a collection in another collection) and ORIGM (which relates an object and the material it is made of). See Table 3 for basic characteristics of these relations. Axiom (a) asserts the limited transitivity of PARS. Due to the default status of the axiom, transitive chains of PARS relationships can only be followed if no predefined epistemic or functional borderlines are transgressed. Axiom (b) describes the inheritance of parts in a SUB-hierarchy, axiom (c) asserts that a part is normally co-located with the whole. It is also plausible to assume that the part consist of the same material as the whole, see axiom (d). Axiom (e) asserts the intransitivity of membership. Axioms (f) through (j) describe the properties of situational composition HSIT(s_1, s_2), expressing the embedding of a subsituation s_2 in an enclosing situation s_1. Axiom (f) states that HSIT be transitive (i.e. scoring a goal in the final match of a soccer championship is also part of the championship), axiom (g) asserts that if the soccer world championship takes place in Germany, then the final of this championship also takes place in Germany, and axiom (h) states that if c is a sufficient condition for s to take place (expressed by the relation COND), then c is also sufficient for situational parts of s to take place. Finally, axiom (i) describes a widening of circumstantials for a given situation: If the goalkeeper becomes injured in the final match of the 2006 world championship (subsituation), then he becomes injured in the 2006 world championship (enclosing situation).

3 Layer Structures and Their Utility to NLP

The classification of concepts in terms of sorts, subordination, and meronymy is too weak to cope with complex linguistic phenomena like quantification, scope, modal embeddings and propositional attitudes, presuppositions etc. Here we focus on the aspects of facticity (real vs. nonreal or hypothetical entities), genericity (concrete instances vs. generic abstractions) and extensionality or plurality (i.e. individuals vs. collections). Due to the independence of these aspects from the hierarchy-forming relations of subordination and meronymy, we treat these aspects as *ontological features* defined for all conceptual entities of suitable sort. This annotation effects a stratification of the conceptual entities into *layers* of those entities which share the same value of a given layer feature. The feature-based representation lends itself to the compositional construction of a semantic representation by means of unification and is thus suited for automated semantic analysis. Moreover these representations are simple and robust enough to provide an immediate benefit for NLP applications.

3.1 Facticity

One important aspect of NL meaning which is not accounted for by the relational representation is concerned with the *facticity* of the entities mentioned in the discourse. Within natural language we can discuss the existence or non-existence of objects of a certain kind – like ghosts, aliens or black holes. It is also possible to talk about situations or events regardless of facticity. Consequently, Hobbs [13] emphasizes the importance of modeling non-existing objects for knowledge representation. In predicate logic, however, there is an inherent existence assumption for all individuals (i.e. individual constants) because the model-theoretic semantics demands that all constants denote in the universe. The ontological feature FACT therefore expresses the facticity of a considered entity. The FACT feature discerns three cases: real entities [FACT *real*], non-existing entities [FACT *nonreal*] and hypothetical entities [FACT *hypo*]. The third value *hypo* is needed e.g. to describe states of affairs in conditionals (relation COND), e.g. "*If*

Table 4. Selected axioms related to facticity

(a)	$R(x, x') \wedge \mathrm{FACT}(x) = \mathit{real} \rightarrow \mathrm{FACT}(x') = \mathit{real}$
	where $R \in \{\mathrm{AGT}, \mathrm{EXP}, \mathrm{MEXP}, \ldots\}$ (most argument roles)
(b)*	$\mathrm{MCONT}(x, x') \rightarrow \mathrm{FACT}(x') = \mathit{hypo}$
(c)	$\mathrm{SUBS}(s, \mathit{know}) \wedge \mathrm{MCONT}(s, s') \wedge \mathrm{FACT}(s) = \mathit{real} \rightarrow \mathrm{FACT}(s') = \mathit{real}$
(d)	$R(x, x') \wedge \mathrm{FACT}(x) = \mathit{real} \rightarrow \mathrm{FACT}(x') = \mathit{real}$
	where $R \in \{\mathrm{PARS}, \mathrm{ELMT}, \mathrm{HSIT}\}$ (meronymy relations)
(e)*	$\mathrm{COND}(s, s') \rightarrow \mathrm{FACT}(s) = \mathit{hypo} \wedge \mathrm{FACT}(s') = \mathit{hypo}$

John returns late, he will be tired." In this case, there is obviously no commitment as to the facticity of premise and conclusion, which are thus classified *hypo*.[3]

Table 4 presents some basic axioms related to facticity. Axiom (a) states that the facticity of a situational concept demands the facticity of its argument roles. For example, Maria can only believe in something if she is a real person. The same can also be said about LOC and TEMP, i.e. a real situation takes place at a real location and has a temporal extent in the real world. An exception to axiom (a) is the MCONT role which expresses mental content or modal embedding. Here, Maria's belief in the existence of the Yeti gives no indication as to the actual existence of the Yeti, which is classified *hypo* by axiom (b). However, as shown by (c), there are exceptional cases (e.g. the mental state of knowing something) which justify this conclusion. Axiom (d) states that a real aggregate can only be composed of real parts. Finally, axiom (e) expresses the above observation that if-then-relationships involve hypothetical situations.

Utilizing Facticity Information for NLP. There is growing interest in supporting question answering and, more generally, systems for recognizing textual entailment by an explicit modeling of facticity and facticity-related inferences. Successful examples of integrating facticity-related information are de Marneffe et al [14], whose system checks if an embedded proposition has a non-factive verb as a parent, and Hickl and Bensley [15], whose discourse commitment-based approach considers presuppositions including conversational implicatures. Bobrow et al [16] propose a logic for textual inference which uses 'instantiability conditions' to express facticity. Nairn et al [17] present a typology of factive and implicative English verbs which was formalized in that logic.

Klutsch [18] presents a typology of factive and implicational verbs which is a refinement of the proposal of Nairn et al. The typology was used for a classification of the German verbs represented in the computational lexicon HaGenLex [3]. Moreover a system of facticity propagation rules was developed which makes it possible to derive unknown facticity values in complex sentences from known FACT values of other entities, from facticity projection properties of the involved participant roles, and from the presence of implicative or factive verbs of a certain type.

This facticity annotation based on the FACT feature is used in the MAVE answer validator, which is part of the IRSAW question answering system [6]. Suppose the system

[3] Based on the hypothetical/real distinction, one can also approach the modelling of counterfactuals, see [2, Chap. 12.3].

Table 5. Selected axioms related to genericity

(a)	$R(x, x') \rightarrow \text{GENER}(x) = \text{GENER}(x')$ where $R \in \{\text{AGT}, \text{EXP}, \text{MEXP}, \dots\}$ (most argument roles)
(b)	$\text{MCONT}(s, s') \wedge \text{GENER}(s') = \textit{sp} \rightarrow \text{GENER}(s) = \textit{sp}$
(c)	$R(s, d) \wedge \text{GENER}(s) = \textit{sp} \rightarrow \text{GENER}(d) = \textit{sp}$, $R \in \{\text{LOC}, \text{TEMP}\}$
(d)	$R(x, x') \rightarrow \text{GENER}(x) = \text{GENER}(x')$ where $R \in \{\text{PARS}, \text{ELMT}, \text{HSIT}\}$ (meronymy relations)
(e)	$R(x, x') \rightarrow \text{GENER}(x') = \textit{ge}$, for $R \in \{\text{SUB}, \text{SUBS}\}$

is asked *"Who escaped Mozambique jail?"* and suppose that the answer candidate *"the Cardoso killer"* was found in a text passage *"The Cardoso killer again tried to escape Mozambique jail."* Then the answer is not validated by the text since the question relates to situations of successful escaping marked by [FACT *real*]. There is no evidence for this facticity value in the text (assuming conventional use of "*try*", it was probably not successful). MAVE accounts for that by decreasing the rank of an answer if it depends on hypothetical information or information about non-existing entities [6].

3.2 Degree of Generality

Another representational challenge to be mastered is concerned with the genericity (degree of generality) of a conceptual entity. A possible starting point for discussing genericity is the T-Box/A-Box distinction known from Description Logics [10], which essentially separates assertions about individual instances (in the A-Box) from knowledge which is universally valid for arbitrary instances of a concept (terminological knowledge in the T-Box). However, consider the example "*Quicksilver was known to the ancient Chinese*". The sentence does not express general knowledge about instances of *quicksilver* but rather about an abstraction which represents quicksilver in a more general sense. We thus consider generic entities in addition to the familiar specific entities. By introducing the genericity attribute GENER which divides the world of conceptual entities into generics [GENER *ge*] and specific individuals [GENER *sp*], assertions about the generic concept and about instances of that concept can be cleanly separated. Generic entities also serve to model prototypical knowledge. Consider the sentence "*The squirrel likes eating nuts*". Universal quantification over all instances of squirrels would be inadequate because the sentence expresses default knowledge only. In our formalism, however, one can use a generic entity to model a squirrel which acts as a prototype. Table 5 lists axioms related to genericity. Axiom (a) asserts that a situation and its agent or (mental) experiencer must have the same genericity type. This property of co-genericity is shown by most argument roles of situations, but MCONT (mental content) is an exception. The example "*Peter does not know that the squirrel likes nuts*" shows that a specific propositional attitude (Peters knowledge about squirrels eating nuts) can embed a generic sentence "*the squirrel likes eating nuts*". Still, MCONT satisfies axiom (b), i.e. if a mental content is specific, then the embedding mental situation is also specific. The converse pattern (c) is shown by LOC and TEMP. In fact, generic sentences like "*The polar bear lives in the arctic*" or "*The dinosaur died out at the end*

Table 6. Selected axioms related to the extensionality type

(a)	$\text{ELMT}(e, c) \rightarrow \text{ETYPE}(c) = \text{ETYPE}(e) + 1$
(b)	$\text{SUBM}(c, c') \rightarrow \text{ETYPE}(c) > 0$
(c)	$\text{SUBM}(c, c') \rightarrow \text{ETYPE}(c) = \text{ETYPE}(c')$
(d)	$\text{PROPR}(c, p) \rightarrow \text{ETYPE}(c) > 0$
(e)	$\text{SUBS}(s, \langle \textit{die out} \rangle) \wedge \text{EXP}(s, e) \rightarrow \text{ETYPE}(e) > 0 \vee \text{GENER}(e) = \textit{ge}$

of the mesozoikum" refer to specific regions in time or space. However, a concrete situation cannot have a generic location or temporal extension. Axiom (d) summarizes that the meronymic relations (PARS, ELMT, HSIT) entail co-genericity. In general, if there is any interaction or physical coupling between two participants of a situation, then these participants must be co-generic. Still, there are a few more relations which may cross genericity levels, including the prominent cases SUB and SUBS. Axiom (e) rephrases the signature constraint expressed by the notation $\overline{o}$ and $\overline{si}$ for generics in Table 1.

Utilizing the Degree of Generality for NLP. The GENER feature has proven useful in NLP applications like question answering and automated knowledge acquisition: (a) Knowledge integration or 'assimilation' [19], in particular coreference resolution [4], can profit from the GENER markup, since co-referring mentions are always co-generic. This information eliminates ambiguities concerning the proper antecedent. (b) The MAVE answer validator uses GENER for avoiding wrong results of the textual entailment check. Basically, when the question requests information about a concrete object or event, as expressed by the constraint [GENER *sp*] on the queried variable, then the variable can only be bound to an entity in the supporting background knowledge which is also specific. In this way GENER improves validation results by eliminating a source of false positives.

In perspective, the GENER annotation makes it possible to automatically extract general knowledge from analyzed text, since it clearly separates specific and generic fractions of the represented document content. Rules for projecting GENER assignments, like those sketched in Table 5, are very important in this respect since they can infer unknown GENER values from GENER values of other conceptual entities and the information provided by the semantic representation of the surrounding text.

3.3 Extensionality Type

The MultiNet formalism allows a simplified extensional characterization of specific concepts with [GENER *sp*] by means of the ontological feature ETYPE (extensionality type), which serves to distinguish individuals from pluralities. For example, ⟨*the best player of the soccer world championship 2006*⟩ denotes an individual [ETYPE 0], ⟨*the German soccer team*⟩ denotes a collection of individuals [ETYPE 1], and ⟨*the teams at the soccer world championship 2006*⟩ denotes a group of groups [ETYPE 2]. Table 6 lists basic axioms related to the ETYPE. Axioms (b) and (d) rephrase the $\ddot{o}$-notation for collections used in Table 1, while (a) and (c) specify the admissible arguments of ELMT and SUBM with more detail compared to the signature table. Thus, c can only be a subset of c' if both collections share the same ETYPE. Axiom (e) applies the ETYPE for describing the combinatory potential of situational concepts: The proposition "*x has died out*" can only be asserted of generic entities or collections.

Applications of the Extensionality Type. As shown by the latter example, the ETYPE can be used for a differentiated description of lexical entries, which in turn reduces ambiguities in syntactic and semantic analysis. In particular, knowing the ETYPE is important for PP interpretation. An example for that is given by the preposition "*in*", which can mean local containment ("*the ball in the bowl*") or membership in a collection ("*the politician in the party*"). The WOCADI parser which automatically generates MultiNet representations from German text uses corresponding PP interpretation rules involving ETYPE constraints [4]. The ETYPE is also used in disambiguation rules for coreference resolution and parse refinement rules which elaborate the result of syntactic-semantic analysis after finishing the main parse. Apart from these uses during parsing, the ETYPE is also available in the axiom syntax. As shown by Table 6, this is useful for integrating set-theoretic notions (like membership) into the KR formalism. Moreover the ETYPE can be used to restrict application of an axiom to individuals or to collections.

4 Classifying Relational Knowledge

So far we considered the classification of conceptual entities by sorts, relations, and ontological features. However, consider the sentences (a) "*The old man is wise*" and (b) "*The wise man is old.*" In both cases, the man of interest (say m) is described by the same elementary relationships SUB(m, man), PROP(m, old) and PROP(m, wise). To capture the difference between the sentences, we analyse the relationships as to their role for describing the involved arguments. A relationship is considered defining or 'categorical' for an argument node if it must necessarily be known in order to understand what the node stands for. In sentence (a), m is characterized as an old man, i.e. SUB(m, man) and PROP(m, old) are considered defining, while m being wise, which merely adds further knowledge about m, is said to be assertional or 'situational' knowledge. In (b), by contrast, m is described as a wise man, i.e. SUB(m, man) and PROP(m, wise) is defining while PROP(m, old) is assertional. The annotation by knowledge types 'c' (categorical) and 's' (situational) thus captures the difference between (a) and (b). The role of these KTYPEs for logical answer finding is explained in more detail in [20].

The KTYPE assignment is also needed for a proper treatment of propositional attitudes. In our example of an analyst's expectations on Google's stock price (see Fig. 2), the analyst's doubt becomes a modal operator which embeds the future increase in Google's stock price represented by node c_3. We may then ask which *proposition* the modal operator applies to. The point is that the sort hierarchy knows situational concepts, but does not avail us with (reifications of) propositions. The solution rests on the KTYPE annotation: A modally restricting edge like MCONT (labelled by 'r' in the figure) will always pick up the proposition formed by the literals in the *capsule* of the restricted node, which comprises all literals marked as categorical. The propositional content of the analyst's doubt is therefore given by SUBS$(c_3, \mathit{increase}) \wedge$ TEMP$(c_3, \mathit{present.0}) \wedge$ EXP(c_3, c_4), which is the proper description of the embedded situation c_3.

Utility of Knowledge Types for NLP. In coreference resolution, one must distinguish between known information about a mention (i.e. occurrence of an NP denoting an entity of discourse), which helps identifying the antecedent, and novel information about the mention, which corresponds to the assertion made by the new sentence. The following example will illustrate this distinction: "*Peter bought (a bungalow)$_i$. (The house)$_i$ is on the hill.*" Knowing that the mention "*the house*" in the second sentence refers to a house makes it possible to identify the proper antecedent i (i.e. the bungalow) assuming the background knowledge that $\mathrm{SUB}(\mathit{bungalow}, \mathit{house})$. This information is contained in the (defining) capsule of the referring node and thus marked as categorical. The new information (location of the house) makes an assertion about the known object (regardless of how it was identified). Thus being located at the hill is treated as assertional (outside capsule) and marked as situational, i.e. $[\mathrm{KTYPE}\, s]$.

The KTYPE is of special importance to NL generation since it separates the core characteristics of a concept (i.e. the interior of the capsule with $[\mathrm{KTYPE}\, c]$), and the contingent knowledge with $[\mathrm{KTYPE}\, s]$. Thus, Kintzel [21] uses the KTYPE annotation for content selection in an NL generation component for German. In advanced question answering (QA) systems which involve NL generation, we can distinguish a first phase of determining the 'answer core', i.e. the queried entity, and a second phase of determining the best way of describing the found entity. The second problem does not call for logic but rather for some form of retrieval which selects the relevant material. The QA system InSicht [5] exemplifies such a combination of semantic network matching (for identifying the conceptual entity of interest) and NL generation.

5 Conclusion

In this paper, we outlined an integrated approach to knowledge representation and natural language semantics suitable for question answering and similar NLP tasks. While incorporating advanced features like generic entities and pluralities, the approach tries to keep the balance between expressiveness and manageability.

References

1. van Benthem, J., ter Meulen, A. (eds.): Handbook of Logic and Language. North-Holland, Amsterdam (1997)
2. Helbig, H.: Knowledge Representation and the Semantics of Natural Language. Springer, Berlin (2006)
3. Hartrumpf, S., Helbig, H., Osswald, R.: The semantically based computer lexicon HaGenLex – Structure and technological environment. Traitement automatique des langues 44(2), 81–105 (2003)
4. Hartrumpf, S.: Hybrid Disambiguation in Natural Language Analysis. Der Andere Verlag, Osnabrück, Germany (2003)
5. Hartrumpf, S.: Question answering using sentence parsing and semantic network matching. In: Peters, C., Clough, P., Gonzalo, J., Jones, G.J.F., Kluck, M., Magnini, B. (eds.) CLEF 2004. LNCS, vol. 3491, pp. 512–521. Springer, Heidelberg (2005)
6. Glöckner, I., Hartrumpf, S., Leveling, J.: Logical validation, answer merging and witness selection – a case study in multi-stream question answering. In: Proc. of RIAO 2007, Pittsburgh, PA (2007)

7. Leveling, J.: Formale Interpretation von Nutzeranfragen für natürlichsprachliche Interfaces zu Informationsangeboten im Internet. Der andere Verlag, Tönning (2006)
8. Hartrumpf, S., Helbig, H., Leveling, J., Osswald, R.: An architecture for controlling simple language in web pages. eMinds: Int. J. on Human-Computer Interaction 1(2), 93–112 (2006)
9. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L.: Sweetening ontologies with DOLCE. In: Gómez-Pérez, A., Benjamins, V.R. (eds.) EKAW 2002. LNCS (LNAI), vol. 2473, pp. 166–181. Springer, Heidelberg (2002)
10. Baader, F., Horrocks, I., Sattler, U.: Description logics. In: Staab, S., Studer, R. (eds.) Handbook on Ontologies, pp. 3–28. Springer, Heidelberg (2004)
11. Brachman, R.: What IS-A is and isn't: An analysis of taxonomic links in semantic networks. IEEE Computer 16, 30–36 (1983)
12. Artale, A., Franconi, E., Guarino, N., Pazzi, L.: Part-whole relations in object-centered systems: An overview. Data & Knowledge Engineering 20(3), 347–383 (1996)
13. Hobbs, J.: Ontological promiscuity. In: Proc. 23rd Annual Meeting ACL, pp. 61–69 (1985)
14. de Marneffe, M.C., MacCartney, B., Grenager, T., Cer, D., Rafferty, A., Manning, C.D.: Learning to distinguish valid textual entailments. In: Proc. of the 2nd Pascal RTE Challenge Workshop (2006)
15. Hickl, A., Bensley, J.: A discourse commitment-based framework for recognizing textual entailment. In: Proc. Workshop on Textual Entailment and Paraphrasing, Prague, ACL, pp. 171–176 (2007)
16. Bobrow, D., Condoravdi, C., Crouch, R., de Paiva, V., Kaplan, R., Karttunen, L., King, T., Zaenen, A.: A basic logic for textual inference. In: Proceedings of the AAAI Workshop on Inference for Textual Question Answering, Pittsburgh, PA (2005)
17. Nairn, R., Condoravdi, C., Karttunen, L.: Computing relative polarity for textual inference. In: Proceedings of ICoS-5 (Inference in Computational Semantics), Buxton, UK (2006)
18. Klutsch, A.: Axiomatische Behandlung von Faktizität in MultiNet unter besonderer Berücksichtigung faktiver und implikativer Verben. Master's thesis, FernUniversität Hagen (2007)
19. Glöckner, I., Hartrumpf, S., Helbig, H.: Automatic knowledge acquisition by semantic analysis and assimilation of textual information. In: Proc. of KONVENS 2006, Konstanz, Germany, pp. 36–43 (2006)
20. Glöckner, I., Helbig, H.: Wissensrepräsentation und Antwortfindung mit Mehrschichtigen Erweiterten Semantischen Netzen. KI 2, 49–55 (2005)
21. Kintzel, F.: Entwurf und Implementierung einer natürlichsprachlichen Generierungskomponente für mehrschichtige erweiterte semantische Netze. Master's thesis, FernUniversität in Hagen (2007)