

Interpretation as Exception Minimization

J. Joachim Quantz

Technische Universität Berlin,
Projekt KIT-BACK, FR 5-12,
Franklinstr. 28/29, W-1000 Berlin 10, Germany
email: jjq@cs.tu-berlin.de

Abstract

Ambiguity is a notorious problem for Natural Language Processing. According to results obtained by Schmitz and Quantz I see disambiguation as a process in which contextual defaults are used to derive the most preferred interpretation of an expression.

I show how contextual information comprising grammatical as well as conceptual knowledge can be modeled in a homogeneous manner using Terminological Logics (TL). I slightly modify the default extension to TL presented by Quantz and Royer to allow a relevance ordering between multisets of defaults. The preferred interpretation is the one containing the fewest exceptions with respect to such an ordering. Interpretation is thus achieved by exception minimization.

I combine this idea with deductive and abductive approaches to interpretation and show how they can be formalized in terms of TL entailment. Furthermore, I obtain a variable depth of analysis by controlling the granularity of interpretation via a set of relevant features.

1 Introduction

Ambiguity is a notorious problem for Natural Language Processing (NLP). Schmitz and Quantz have shown that the same general strategy can be used for the resolution of referential, lexical, and structural ambiguities [Schmitz and Quantz, 93]. The basic idea is that contextual information is used to resolve the local ambiguities (see also [Kameyama *et al*, 91] and [Eberle *et al*, 92]). In most cases, this information cannot be formalized as hard constraints but only as defaults, i.e. as rules which allow for exceptions. Each possible interpretation of an ambiguous expression satisfies some of the defaults while violating others. The idea of *interpretation as exception minimization* is to choose the interpretation which contains the fewest exceptions according to a relevance ordering on the defaults.

A fundamental assumption of this approach towards disambiguation is that contextual information of quite heterogeneous nature, such as syntactic, semantic, conceptual, or domain specific information is used in the disambiguation process. Consequently, the strictly sequential architecture of interpretation used in most existing NL systems, e.g. [Allgayer *et al*,

89], [Stock, 91], [Schmitz *et al*, 91], in which syntactic, semantic, and conceptual information is represented on different levels and in different formalisms, seems to be inadequate.

I see *Head-Driven Phrase Structure Grammar* (HPSG) as an attempt to provide a homogeneous formalism for integrating syntactic and semantic information [Pollard and Sag, 87]. In this paper I try to combine the ideas of HPSG and *Terminological Logics* (TL), which are widely used in NL systems for the representation of conceptual and domain specific knowledge (e.g., the systems cited above and [Bateman, 92]). The aim of this paper is twofold:

- I will show how the syntactic principles of HPSG can be represented in TL;
- I will show how interpretation can be achieved by minimizing the exceptions to contextual defaults and how this can be formalized in terms of TL entailment.

This paper is a formal elaboration of the technical aspects sketched in [Schmitz and Quantz, 93]. I will therefore reduce the linguistic motivation to a minimum and refer the interested reader to the examples presented in that paper. I will also contend myself with a very superficial treatment of HPSG and rather aim at presenting the general formal structure of interpretation as exception minimization in a TL framework. Of course, the general idea of exception minimization could be combined with other formalisms as well.

One final remark: the task of interpretation or disambiguation can be seen as a special case of model-based diagnosis, where the phonological realization is the observed symptom, the grammar and the conceptual knowledge are the underlying model, and the chosen interpretation is the diagnosis. The approach of interpretation as exception minimization is therefore of relevance not only for NLP but for diagnosis tasks in general.

2 Terminological Logics and Typed Feature Structures

The similarity between Terminological Logics (TL) and Typed Feature Structures (TFS), which provide the formal paradigm of Unification Grammars as HPSG, has been mentioned in a number of papers (e.g., [Nebel and Smolka, 87], [Backofen *et al*, 90], and [Carpenter, 92]). The basic constructs of TFS are *feature structures*, i.e. sets of feature value pairs, where features are partial functions from feature structures into values. The corresponding entities in TL are *concepts*. Like

Satisfaction of formulae is then defined as follows:

$$\mathcal{M} \models t_2 \sqsubseteq t_1 \quad \text{iff} \quad [t_2]^{\mathcal{M}} \subseteq [t_1]^{\mathcal{M}} \quad (9)$$

$$\mathcal{M} \models o :: c \quad \text{iff} \quad [o]^{\mathcal{M}} \in [c]^{\mathcal{M}} \quad (10)$$

A structure $\mathcal{M}$ is a model of a formula γ iff $\mathcal{M} \models \gamma$; it is a model of a set of formulae Γ iff it is a model of every formula in Γ . A formula γ is entailed by a set of formulae Γ (written $\Gamma \models \gamma$) iff every structure which is a model of Γ is also a model of γ .

Note that the following abbreviations are frequently used in this paper:

$$c_1 \rightarrow c_2 \stackrel{\text{def}}{=} c_1 \sqsubseteq c_2 \quad (11)$$

$$t_1 := t_2 \stackrel{\text{def}}{=} t_1 \sqsubseteq t_2, t_2 \sqsubseteq t_1 \quad (12)$$

$$\text{the}(r, c) \stackrel{\text{def}}{=} \forall r : c \sqcap 1 r \quad (13)$$

$$r : (o_1 \sqcap \dots \sqcap o_n) \stackrel{\text{def}}{=} r : o_1 \sqcap \dots \sqcap r : o_n \quad (14)$$

$$r : (o_1 \sqcup \dots \sqcup o_n) \stackrel{\text{def}}{=} r : o_1 \sqcup \dots \sqcup r : o_n \quad (15)$$

In addition to this core TL, I assume in this paper the integration of *test/compute facilities*, of *epistemic operators*, and of *defaults*. I will need these constructs for the modeling of the grammar, for the definition of explicit interpretations, and for the specification of preferences guiding the disambiguation. I will briefly sketch the basic ideas of these extensions but refer the interested reader to the literature for more detailed information. I will not address the issue of decidability or tractability for the resulting TL. Even without these extensions, the core TL needed for representing linguistic information is probably undecidable (see [Donini *et al*, 91] for complexity results of TL).

The test/compute construct allows the attachment of external procedures into a TL system. The formal integration of the compute construct into the TL system BACK is described in [Kortüm, 93]. The basic idea is that **compute**($r, \text{pred}(arg_1, \dots, arg_n)$) can be used to determine the role-filler(s) for a role r by calling the external predicate *pred* with the arguments arg_i . These arguments will typically be the values of other roles.

The integration of epistemic operators into TL has been proposed in [Donini *et al*, 92]. The **K** operator can be used to distinguish cases in which an object is known to be a filler for role r , from cases in which just the type of the role-filler is known. Thus $1 r$ describes all objects which are known to have exactly one filler for r , but $1 K r$ describes only those objects for which this filler is explicitly known.

Finally, I need a default extension of TL as the one developed in [Quantz and Royer, 92]. The idea is to model defaults as rules $c_1 \rightsquigarrow c_2$ that allow for exceptions. Each object which is a c_1 is assumed to be a c_2 unless this leads to inconsistencies. A priority ordering on defaults stemming from the specificity of the premises wrt the subsumption hierarchy is used to resolve conflicts between defaults.

Given such a TL I will now show how interpretation can be defined formally in terms of TL entailment, how syntactic principles from HPSG can be modeled as TL formulae, and how defaults can be used to guide disambiguation by realizing interpretation as exception minimization.

3 Interpretation and Disambiguation

Consider the following example taken from [Schmitz and Quantz, 93]:

- (3) Die Hersteller produzieren Schaltkreise.
- (4) (The manufacturers produce integrated circuits.)
- (5) $e :: \text{phon: "Die Hersteller produzieren Schaltkreise"} \sqcap$
 $\text{phrase} \sqcap 4 \text{ atom} \sqcap \text{atom: } (w_1 \sqcap w_2 \sqcap w_3 \sqcap w_4)$
 $w_1 :: \text{word} \sqcap \text{pos:1} \sqcap \text{phon: "Die"}$
 $w_2 :: \text{word} \sqcap \text{pos:2} \sqcap \text{phon: "Hersteller"}$
 $w_3 :: \text{word} \sqcap \text{pos:3} \sqcap \text{phon: "produzieren"}$
 $w_4 :: \text{word} \sqcap \text{pos:4} \sqcap \text{phon: "Schaltkreise"}$

I see (5) as representing the information explicitly given by sentence (3). Since it contains mainly phonological information I use Π_e to denote this set of TL formulae.

The task of interpretation consists in computing the information implicitly given in (5). Let us assume that another set of TL formulae Γ contains grammatical information (examples will be given in the next section). The *deductive* approach to interpretation (see [Pereira and Warren, 80]) can then be formalized as follows:¹

Definition 1 Let e be any NL expression. If $\Gamma \cup \Pi_e$ is inconsistent, e has no interpretation. Otherwise, the **deductive interpretation** of e is defined as the most specific concept i_d such that $\Gamma \cup \Pi_e \models e :: i_d$.

Due to the (semi-)free word order of German, both Schaltkreise and Die Hersteller can be the subject of sentence (3) (thus the English sentence (4) is the translation of the preferred interpretation in which Die Hersteller is subject). Given an adequate formalization of German grammar as Γ we obtain the following deductive interpretation

$$\Gamma \cup \Pi_e \models e :: \text{subject:}(p_1 \sqcup p_3)$$

where p_1 is the phrase Die Hersteller and p_3 is the phrase Schaltkreise.

I do not think that this matches our intuition, however: I'd rather say that e has two interpretations and that $\text{subject:}(p_1 \sqcup p_3)$ is too unspecific to count as an interpretation. It thus seems that further restrictions on interpretations are required. The general idea is that the values of certain features have to be determined and that disjunctive information about them is not enough.

This can be formalized by using epistemic operators as proposed in [Donini *et al*, 92]. The requirement that a value for a feature f must be known is expressible as the concept $1 \text{ K}f$. We say that this concept is *explicit* wrt f and give the following general definition of explicitness.

Definition 2 A concept c is **explicit** wrt a role r and a set of TL-formulae Γ iff $\Gamma \models c \sqsubseteq K \geq n \text{ r}$ only if $\Gamma \models c \sqsubseteq \geq n \text{ K}r$. A concept c is **explicit** wrt a set of roles ρ iff it is explicit wrt each role in ρ .

Note that explicitness is a relative notion which is only meaningful wrt a given set of roles. But this is exactly what is needed for a variable depth of analysis. We can stipulate that an interpretation has to be explicit wrt a set of *relevant roles*.

¹In all the definitions in this paper I assume that Γ is a set of TL formulae modeling the grammar, that Π_e is a set of TL formulae modeling the phonological realization of an NL expression e , and that Δ is a set of TL defaults.

Thus, the depth of analysis can be controlled via the set of relevant roles.²

Given the notion of explicitness we can formalize an abductive approach to interpretation (see [Hobbs *et al*, 93]). In general, abduction is the inference of a formula A from the formulae $A \rightarrow B$ and B . To give an intuition for this inference, which clearly is not deductively valid, consider B an observed fact and A an explanation for B given that $A \rightarrow B$.

Definition 3 Let e be any NL expression and ρ a set of roles relevant for interpretation. If $\Gamma \cup \Pi_e$ is inconsistent, e has no interpretation. Otherwise, the **abductive interpretations** of e are the least specific concepts i_a , explicit wrt ρ , such that $\Gamma \cup \{e :: i_a\}$ is consistent and $\Gamma \cup \{e :: i_a\} \models \Pi_e$.

Note that the requirements of generality and explicitness control the depth of analysis: on the one hand, an abductive interpretation must be specific enough to be explicit wrt a set of relevant roles; on the other hand, it is only as specific as necessary to explain the phonological information, i.e. as general as possible.

Note further that in cases of ambiguity, we obtain several abductive interpretations and the task of disambiguation consists in determining the preferred one. I will address this issue in Section 5.

4 Linguistic Information and Ambiguity

Having sketched the general framework of interpretation, I will now turn my attention towards the formalization of HPSG principles as TL formulae. In HPSG two types of signs, namely *words* and *phrases* are distinguished. In the *lexicon*, feature structures are assigned to the words of a language. Lexical rules can be used to model the general principles underlying the lexical inventory of a language. First, the wordforms of a lexeme have a lot of information in common and only some information is specific to the single wordforms. Second, there are general rules governing the derivation of wordforms from the stem of a lexeme. These general rules allow for exceptions, however, which are known as irregular forms (see [Krieger and Nerbonne, 91]).

Lexical information can be used to illustrate the difference between the deductive and the abductive approach to interpretation. In the previous section I took the phonological information associated with an expression as the starting point for interpretation. Thus given the information $e :: \text{phon: "is"}$ we would like to infer $e :: \vee \sqcap \text{per:3} \sqcap \text{plu:-} \sqcap \text{lexeme:be}$. To get this inference in the the deductive framework, we would need rules like

$$\begin{aligned} \text{phon: "is"} &\rightarrow \text{lexeme:be} \sqcap \text{per:3} \sqcap \text{plu:-} \\ \text{lexeme:be} &\rightarrow \vee \sqcap \text{stem: "be"} \end{aligned}$$

This representation has two disadvantages, however: for one thing it does not capture the general derivation rules; furthermore, for ambiguous wordforms we have to model disjunctive information in the righthand side of the rules.

It seems more natural to choose a representation like the following:

²I would like to thank Christa Hauenschild for pointing out to me the importance of a variable depth of analysis for Machine Translation. The basic idea is that ambiguous expressions do not have to be disambiguated if the same ambiguity is expressible in the target language.

lexeme:give → $v \sqcap \text{stem:} \text{"give"} \sqcap \dots$
 $v \sqcap \text{per:3} \sqcap \text{plu:-} \rightsquigarrow \text{compute}(\text{phon}$
 $\text{concat}(\text{lexeme.stem}, \text{"s"}))$

lexeme:be $\sqcap$
 $\text{per:3} \sqcap \text{plu:-} \rightarrow \text{phon:} \text{"is"}$

Such a modeling yields the desired inferences within the abductive approach. Thus $e :: \text{phon:} \text{"is"}$ can be explained by assuming that $e :: \text{lexeme:be} \sqcap \text{per:3} \sqcap \text{plu:-}$.

Note that in HPSG this modeling is used with a deductive approach together with a closure of grammatical information. Thus, for example, the lexemes mentioned in the lexicon are taken to be all lexemes of the language:

$\text{word} \rightarrow \text{lexeme:}l_1 \sqcup \dots \sqcup \text{lexeme:}l_n$

I think that this approach is not cognitively adequate, since it does not allow the interpretation of wordforms for which no lexeme exists in the lexicon. But clearly, human speakers are able to interpret expressions containing such forms and usually form a hypothesis what the stem of the ‘missing’ lexeme would be like.

For the purpose of this paper it suffices to see a general equivalence between the deductive and the abductive approach: the abductive approach can be simulated deductively by adding the closure

$c \rightarrow c_1 \sqcup \dots \sqcup c_n$

for a list of rules

$c_1 \rightarrow c$

$\dots$

$c_n \rightarrow c$

In the remainder of this section I will present some of the syntactic principles of HPSG and show how they can be modeled in TL.

Whereas words form the atomic elements of a natural language — at least with respect to the task of interpretation — phrases are complex entities arising from the combination of signs. In addition to the lexicon, a grammar has thus to specify the principles underlying this combination. In HPSG there are a number of such principles, in this paper I will concentrate on the most important ones only. In general, these principles constrain the set of valid feature structure, i.e., they filter out some feature structures as ill-formed.

The internal structure of a phrase is represented in HPSG by the daughters (dtrs) feature, which further divides into the features head-dtr and comp-dtrs. Representing this in TL I will model dtrs as a role with a sub-feature head-dtr and a sub-role comp-dtrs.

The head-dtr is assigned special treatment for two reasons. First, some feature values of the head-dtr determine the corresponding feature values of the overall phrase. This is the content of the *Head Feature Principle* (HFP), which can be formalized as follows:

$\text{phrase} \rightarrow \text{plu} = \text{head-dtr.plu} \sqcap$
 $\text{cas} = \text{head-dtr.cas} \sqcap \dots$

The features constrained in the HFP are usually called *head features*.

Second, the head-dtr acts like a functor in that it takes the comp-dtrs as arguments to produce the overall phrase. This perspective, which owes a lot to the ideas underlying Categorical Grammar, is reflected in the subcat feature in which

number and types of the arguments of a functor are specified. I will model subcat as a TL-role with sub-features arg1, ..., arg4. Thus, the following lexicon entry expresses that produzieren expects two arguments, one being in nominative case, the other in accusative:

$\text{lexeme:produzieren} \rightarrow 2 \text{ subcat} \sqcap$
 $\text{the}(\text{arg1}, \text{np} \sqcap \text{cas:nom}) \sqcap$
 $\text{the}(\text{arg2}, \text{np} \sqcap \text{cas:acc})$

The functional application is controlled by the *Subcategorization Principle* (SP), which basically requires that the subcat value of a phrase contains those arguments of head-dtr, which are not members of comp-dtrs:

$\text{phrase} \rightarrow \text{compute}(\text{subcat},$
 $\text{subtract}(\text{head-dtr.subcat}, \text{comp-dtrs}))$

Whereas HFP and SP express general constraints on the combination of signs, *Immediate Dominance* (ID) *Schemata* are used to single out those combinations that actually are phrases of a language. The ID *Principle* states that each phrase must be an instance of one of the ID Schemata. We thus get:

$\text{phrase} \rightarrow \text{id1} \sqcup \text{id2}$
 $\text{id1} := \text{phrase} \sqcap 0 \text{ subcat} \sqcap$
 $\text{the}(\text{head-dtr}, \text{lex:-}) \sqcap$
 1 comp-dtrs
 $\text{id2} := \text{phrase} \sqcap 1 \text{ subcat} \sqcap$
 $\text{the}(\text{head-dtr}, \text{lex:+})$

The purpose of this section was to show that the syntactic information modeled in HPSG can be in principle represented in TL. I will give no examples for the modeling of semantic, conceptual, or encyclopedic information in TL, since this has been already exemplified in a number of existing systems, e.g. [Allgayer *et al.*, 89], [Stock, 91], [Schmitz *et al.*, 91], [Bateman, 92]. In the following I will therefore assume that Γ contains information of quite heterogeneous nature modeled homogeneously as TL formulae. It remains to be shown how defaults can be integrated into the formal framework of interpretation presented so far.

5 Disambiguation as Exception Minimization

The abductive approach to interpretation can be refined to handle ambiguities. Note that multiple abductions result from the existence of several explanations, i.e. formulae $A_i \rightarrow B$. The idea of weighted abduction, as proposed in [Hobbs *et al.*, 93], is to assign costs and factors to the A_i . Then the preferred abduction is the cheapest one, i.e. the one containing the least expensive explanations.

Instead of pursuing the approach of weighted abduction, however, I will use defaults to model preferences between interpretations. The main reason for doing so is that the approach of weighted abduction cannot deal with the default information we need in order to model the contextual constraints.³

In [Schmitz and Quantz, 93] it is explained in detail how defaults can be seen as a means to model preference rule systems as proposed in [Jackendoff, 83]. Thus the default c_1

³Hobbs pointed out to me, however, that their “et cetera” predicates can be used to model uncertain information.

$\leadsto c_2$ says that a situation in which a c_1 is also a c_2 is more likely (and hence a preferred interpretation) than a situation in which it is not a c_2 .

Quantz and Royer have developed a default extension for TL in which a priority ordering between defaults can be specified [Quantz and Royer, 92]. The basic idea of their preference semantics is to prefer models with fewer exceptions. If two defaults are in conflict and stand in a priority relation, then the exception to the less prior default is tolerated.

Consider again the examples from [Schmitz and Quantz, 93]:

(6) Die Hersteller produzieren Schaltkreise.

(7) Schaltkreise produzieren die Hersteller.

We might introduce two defaults to deal with the ambiguity of these examples:

$\delta_1: \text{top:}+ \leadsto \text{cas:nom}$
 $\delta_2: \text{lexeme:produce} \leadsto \text{the(agent,company)}$

The first default says that a topicalized expression is usually nominative, i.e. the unmarked word order in German is the one where the subject is topicalized. The second default says that the filler of the agent role for the lexeme produce is usually a company, i.e. it specifies a selection restriction. In example (6) the interpretation in which Die Hersteller is subject satisfies both defaults, whereas in example (7) it violates the first one. This might be taken as evidence for assigning δ_2 more relevance than δ_1 .

As shown in [Schmitz *et al.*, 91] and [Schmitz and Quantz, 93] most interpretations will generally violate more than one default. In order to choose between such interpretations, we thus need a priority ordering not only between single defaults but rather between sets of defaults. Following [Quantz and Royer, 92] we can define the exceptions to defaults in a model $\mathcal{M}$ as:

$$E_{\mathcal{M}} \stackrel{\text{def}}{=} \{ \langle \delta, d \rangle : d \in \llbracket \delta_p \rrbracket^{\mathcal{I}} \wedge d \notin \llbracket \delta_c \rrbracket^{\mathcal{I}} \}$$

i.e. as pairs of defaults and domain elements which are members of their premise (δ_p) but not of their conclusion (δ_c). Let us now assign a natural number p_δ to each default representing the price of violating it: the more relevant a default, the higher its price. Given these prices we can redefine exception preference [Quantz and Royer, 92, Def. 5] as follows:

Definition 4 Let Γ be any set of TL-formulae, M_1 and M_2 any models of Γ , and Δ any set of defaults. M_2 is **exception-preferred** to M_1 wrt Γ and Δ (written $M_1 \sqsubseteq_E M_2$) iff $\Sigma(E_1) > \Sigma(E_2)$, where $\Sigma(E_i)$ is the result of summing up the prices of the defaults in E_i . (Note that the price of a default has to be added for each occurrence in E_i , i.e., if $\langle \delta, d \rangle \in E_i$ and $\langle \delta, e \rangle \in E_i$ the price of δ will be added twice.)

Having thus modified exception preference, we obtain a Δ -entailment ($\Gamma \models_{\Delta} o :: c$, see [Quantz and Royer, 92, Def. 8]), which takes into account a relevance ordering between multisets of defaults. Based on this Δ -entailment we can define preferred deductive interpretations:

Definition 5 Let e be any NL expression. If $\Gamma \cup \Pi_e$ is inconsistent, e has no interpretation. Otherwise, the **preferred deductive interpretation** of e is defined as the most specific concept i_p such that $\Gamma \cup \Pi_e \models_{\Delta} e :: i_p$.

Thus given the two defaults above together with $p_{\delta_1} < p_{\delta_2}$ and a formalization of German grammar we would obtain

$$\begin{aligned} \Gamma \cup \Pi_{e_6} &\models_{\Delta} e_6 :: \text{subject:p}_1 \\ \Gamma \cup \Pi_{e_7} &\models_{\Delta} e_7 :: \text{subject:p}_3 \end{aligned}$$

where e_6 is example (6), e_7 example (7), p_1 the phrase Die Hersteller in example (6), and p_3 the phrase die hersteller in example (7).

Of course, preferred deductive interpretations suffer from the same shortcomings as deductive interpretations in general: though defaults will resolve most ambiguities, some expressions might be ambiguous even when defaults are taken into account. Thus we might obtain a preferred deductive interpretation which is disjunctive and not explicit wrt a relevant role. But even worse, the depth of analysis is not variable in the deductive framework and we will therefore perform disambiguations which might not be necessary. In order to overcome these shortcomings I will now integrate defaults into the abductive approach to interpretation.

In order to take contextual defaults into account we have to redefine abductive interpretations:

Definition 6 Let e be any NL expression and ρ a set of roles relevant for interpretation. If $\Gamma \cup \Pi_e$ is inconsistent, e has no interpretation. Otherwise, the **abductive interpretations** of e are the least specific concepts i_a , explicit wrt ρ , such that $\Gamma \cup \{e :: i_a\}$ is consistent and $\Gamma \cup \{e :: i_a\} \models_{\Delta} \Pi_e$.

For ambiguous expressions we nevertheless obtain several abductive interpretations. In order to choose between them we need to compare them with respect to the defaults they violate. I thus define

$$E(\Gamma, \Delta) \stackrel{\text{def}}{=} \{ \langle \delta, o \rangle : \Gamma \models_{\Delta} o :: \delta_p \wedge \Gamma \not\models_{\Delta} o :: \delta_c \}$$

Note that these are not exceptions relative to a model, but rather exceptions induced by a set of TL formulae Γ . Hence, the second elements of the exception pairs are objects and not the entities of the domain D of a model.

Definition 7 Let i_1 and i_2 be abductive interpretations of an NL expression e . i_1 is **preferred** to i_2 (written $i_2 \sqsubseteq i_1$) iff $\Sigma(E_1) > \Sigma(E_2)$ (see Definition 4), where $E_i \stackrel{\text{def}}{=} E(\Gamma \cup \{e :: i_i\}, \Delta)$.

An abductive interpretation i_1 of e is **preferred** iff there is no abductive interpretation i_2 of e such that $i_1 \sqsubseteq i_2$.

Note that expressions which are ambiguous even when taking into account default information have more than one preferred abductive interpretation. Furthermore, since abductive interpretations have to be explicit only for a specified set of relevant roles, we obtain a variable depth of analysis.

6 Conclusion

I have shown that Terminological Logics (TL) can be used to model grammatical as well as conceptual knowledge in a homogeneous manner. Furthermore, I have presented a deductive and an abductive formalization of interpretation in terms of TL entailment. In case of ambiguous expressions, contextual constraints modeled as defaults can be used to determine the preferred interpretation. The basic idea is to choose the interpretation yielding a set of exceptions to defaults which is minimal wrt a relevance ordering on the defaults.

To formalize this, I slightly modified the default extension of TL as proposed in [Quantz and Royer, 92]. Instead of a precedence relation between defaults, a precedence relation between multisets of defaults is needed. Moreover, epistemic operators can be used to define the notion of interpretations being explicit wrt a given set of roles. This set of relevant roles can be used to control the granularity of interpretation and thus provides the basis for a variable depth of analysis. The resulting approach to interpretation is very similar to the approach of weighted abduction described in [Hobbs *et al.*, 93]. A detailed evaluation of both approaches is needed to compare their respective advantages and disadvantages.

I see the logical framework presented here as a formal foundation for developing NL systems. To obtain efficient performance, however, heuristics have to be integrated to choose the preferred interpretation. Thus instead of evaluating all possible interpretations by comparing the exceptions they induce, heuristics are needed to test as few interpretation as possible. This should probably be done outside the TL system. Related to this issue is the general topic of integrating abductive reasoning into TL. There is currently some research aiming at this integration for configuration tasks [Buchheit *et al.*, 93].

Another issue concerns the representation of ambiguous information. Variable depth of analysis means that not all ambiguities have to be resolved but only the ones relevant for the specific purpose of interpretation (e.g., to select the appropriate transfer rules in Machine Translation). Thus an efficient representation of ambiguities is needed to avoid exponential computation of disjunctive normal forms. This topic is currently discussed in the literature on feature formalisms [Kasper, 87], [Trost and Backofen, 92].

Acknowledgements

I would like to thank Bob Carpenter, Christa Hauenschield, Jerry Hobbs, Bob Kasper, Werner Nutt, Susanne Preuß, Birte Schmitz, and Carla Umbach for providing valuable information and/or thoroughly discussing the ideas underlying this paper.

References

- [Allgayer *et al.*, 89] J. Allgayer, R. Jansen-Winkel, C. Reddig, N. Reithinger, "Bidirectional Use of Knowledge in The Multi-Modal NL Access System XTRA", in *IJCAI'89*, 1492–1497
- [Backofen *et al.*, 90] R. Backofen, H. Trost, H. Uszkoreit, *Linking Typed Feature Formalisms and Terminological Knowledge Representation Languages in Natural Language Front-Ends*, DFKI Research-Report 91–28, 1991
- [Bateman, 92] J. Bateman, "The Use and Design of Linguistically Motivated Ontologies for Controlling and Deployment of Linguistic Resources", in [Preuss and Schmitz, 92], 50–99
- [Bläsius *et al.*, 87] K.H. Bläsius, U. Hedtstück, C.-R. Rollinger (eds), *Sorts and Types in Artificial Intelligence*, Berlin: Springer, 1987
- [Buchheit *et al.*, 93] M. Buchheit, R. Klein, W. Nutt, "Constructive Problem Solving—A Model Construction Approach Towards Configuration", *Preliminary Draft*
- [Carpenter, 92] B. Carpenter, *The Logic of Typed Feature Structures*, Cambridge: Cambridge University Press, 1992
- [Donini *et al.*, 91] F.M. Donini, M. Lenzerini, D. Nardi, W. Nutt, "The Complexity of Concept Languages", *KR'91*, 151–162
- [Donini *et al.*, 92] F.M. Donini, M. Lenzerini, D. Nardi, A. Schaerf, W. Nutt, "Adding Epistemic Operators to Concept Languages", in [Nebel *et al.*, 92], 342–353
- [Eberle *et al.*, 92] K. Eberle, W. Kasper, C. Rohrer, "Contextual Constraints for MT", *Proceedings of the TMI'92*, 213–224
- [Hobbs *et al.*, 93] J.R. Hobbs, M. Stickel, D. Appelt, P. Martin, "Interpretation as Abduction", to appear in *Artificial Intelligence Journal*
- [Hoppe *et al.*, 93] T. Hoppe, C. Kindermann, J.J. Quantz, A. Schmiedel, M. Fischer, *BACK v5 Tutorial & Manual*, KIT Report 100, Technische Universität Berlin, 1993
- [Jackendoff, 83] R. Jackendoff, *Semantics and Cognition*, Cambridge: MIT Press, 1983
- [Kameyama *et al.*, 91] M. Kameyama, R. Ochitani, S. Peters, "Resolving Translation Mismatches With Information Flow", *ACL'91*, 193–200
- [Kasper, 87] R.T. Kasper, "A Unification Method for Disjunctive Feature Descriptions", *ACL'87*, 235–242
- [Kortüm, 93] G. Kortüm, *How To Compute 1+1? A Proposal for the Integration of External Functions and Computed Roles into BACK*, KIT Report 103, 1993
- [Krieger and Nerbonne, 91] H.-U. Krieger, J. Nerbonne, *Feature-Based Inheritance Networks for Computational Lexicons*, DFKI Research Report 91–31, 1991
- [Nebel and Smolka, 87] B. Nebel, G. Smolka, "Representation and Reasoning with Attributive Descriptions", in [Bläsius *et al.*, 87], 112–139
- [Nebel *et al.*, 92] B. Nebel, C. Rich, W. Swartout, *Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference (KR'92)*, San Mateo: Morgan Kaufmann, 1992
- [Pereira and Warren, 80] F.C.N. Pereira, D.H.D. Warren, "Definite Clause Grammars for Language Analysis—A Survey of the Formalism and a Comparison with Augmented Transition Networks", *Artificial Intelligence* 13, 231–278, 1980
- [Pollard and Sag, 87] C. Pollard, I.A. Sag, *An Information Based Syntax and Semantics*, Vol. I Fundamentals, Stanford: CSLI Lecture Notes 13, 1987
- [Preuss and Schmitz, 92] S. Preuß, B. Schmitz (eds.), *Workshop on Text Representation and Domain Modelling*, KIT-Report 97, Technische Universität Berlin, 1992
- [Quantz and Royer, 92] J.J. Quantz, V. Royer, "A Preference Semantics for Defaults in Terminological Logics", in [Nebel *et al.*, 92], 294–305
- [Schmitz and Quantz, 93] B. Schmitz, J.J. Quantz, *Defaults in Machine Translation*, KIT Report 106, Technische Universität Berlin, 1993
- [Schmitz *et al.*, 91] B. Schmitz, S. Preuß, C. Hauenschield, *Textrepräsentation und Hintergrundwissen in KIT-FAST*, KIT Report 93, Technische Universität Berlin, 1991
- [Stock, 91] O. Stock, "Natural Language and Exploration of an Information Space: the ALFresco Interactive System", in *IJCAI'91*, 972–978
- [Trost and Backofen, 92] H. Trost, R. Backofen (eds), "Coping with Linguistic Ambiguity in Typed Feature Formalisms", *Proceedings of a Workshop held at ECAI'92*