Machine Translation Supported by Terminological Information

Jörg Schütz
1AI
Martin-Luther-Straße 14
D-6600 Saarbrücken
joerg@iai.urn-sb.de

Bärbel Ripplinger
CRP - Centre Universitaire
162a, Avenue de la Faïencerie
L-1511 Luxembourg
babs@crpcu.lu

Abstract

It is well-known that natural language (NL) is highly complex and ambiguous, and designing a system in the sense of 'large scale engineering' rather than in the sense of so-called 'runnable specifications', i.e. computational solutions to pre-selected NL problem areas, which could cope with most complexities of NL, seems not to be feasible in the foreseeable future. Nevertheless, there is a widespread recognition that systems designed for specific purposes are far more likely to be viable. However, in this context the discipline of computational terminology has received little attention in computational linguistics; an unfortunate situation given that natural language processing (NLP) systems seem to be most successful when applied to specialised domains.

In this paper we present an approach that integrates an instance of computational terminology into a constraint-based NLP/MT environment. Parts of this research have been carried out in the context of the ET-10/66 project 'Terminology and Extra-linguistic Knowledge' financed by the Commission of the European Communities (CEC). Like in this project we have chosen the subject field telecommunications as the domain of reference, and the text corpus on which the work is based is the *Handbook on Satellite Communication* of the International Radio Consultative Committee (CCIR); this corpus is an expository type of text.

The problems to be solved through our approach, and which are characteristic for sublanguage texts, relate to multiword term identification, domain-specific attachment of prepositional phrases and the disambiguation of lexical ambiguities. The terminology knowledge used in our project for constructing a terminology knowledge base was partly extracted from the information encoded in the EIRETERM term bank, that is designed primarily for human users, and is based on a linguistically motivated statistical analysis of the reference corpus.

1 Introduction

In this paper we put forward an approach that integrates domain-specific terminological knowledge into a natural language (NL) translation process primarily on a lexical basis. The system has been developed for Gerrnan-to-English partly within the ET-10/66 project funded by the CEC; its computational linguistic basis is the ALEP[1] framework, which constitutes a constraint-based computational linguistic development environment based on the toolbox architecture paradigm (cf., for example, [Schütz et al., 1991]).

Based on ALEP intrinsic requirements we have subdivided the NL analysis into a parsing step based on pure syntactic information and a refinement step that adds semantic and terminological information to the parsing result, thus enhancing and filtering the syntactic structures in terms of information content and of well formedness according to the considered domain. For the refinement

[1] ALEP is an acronym for *Advanced Language Engineering Platform* promoted by the CEC for their Linguistic Research and Engineering (LRE) programme. It is derived from the ET-6 series of studies ([Pulman (ed.), 1991], [Schütz (ed.), 1991] and [Devillers (ed.), 1991]).

process we have extended the semantic lexicon with additional terminological information in order to make use of domain-specific information.
The problems characteristic for the analysis and translation of sublanguage texts relate to identifying multiword terms (MWT), e.g. 'time division multiplexing', attachment problems of domain-specific prepositional phrases (PP) and domain-specific lexical ambiguity e.g. term and non-term reading. In most NLP systems (cf., for example, the EUROTRA approach) MWTs were/are handled by specialised lexical rules to avoid a structural description, but to ensure an easy information access for subcategorisation purposes. In EUROTRA, for instance, each term (including MWT) got an unique term identification which only had relevance for triggering the translation process.
In our approach we use terminological information in analysis for the identification and selection of domain-specific readings; this, then, will reduce the actual translation amount, since transfer can operate on language independent conceptual information. At a first stage we have added domain-specific conceptual information only to lexical descriptions, which are associated with different surface realisations for different languages through the appropriate grammars. We have not taken into account extensions to the analysis grammar (designed for general language) which would reflect the specificities of the domain's sublanguage, although some changes to the grammar were made in order to ensure an appropriate use of the terminological information, i.e. the percolation of this information and the testing of its validity in terms of combinations that are allowed in the domain ('connectability' of information).
Our lexicon based approach so far has proven to be sufficient for demonstrating the advantage of using domain-specific information for selecting the readings that are valid within the domain (disambiguation), at least as one approach for introducing domain-specific information and knowledge into a NLP/MT framework.
The actual translation process will work on these specific conceptual descriptions plus other necessary translational relevant information, like register information (according to the considered domain). The synthesis process is responsible for generating the appropriate syntactic structures from the conceptual information, taking into account determiners, prepositions and the generation and/or selection of language dependent MWTs.

The methodological idea underlying our approach is to associate specific concepts of the domain (which are represented in the domain's ontology; cf. below) with automata, so-called conceptual graphs (cf. [Sowa, 1991]), that express potential linguistic realisations in the domain. At present these automata are realised in terms of the type system facility of ALEP and the Unix m4 macro facility. Additionally, these automata would also serve as the recognition machinery for terminological expressions (cf. [Schütz, 1992]). This method can be easily adapted to a multilingual environment; for example, a concept EARTH_STATION will have an English realisation as *earth station,* a French as *station terrienne*[2] , and a German as *Erdefunkstation.*

2 Information and Knowledge Resources

2.1 The Domain

The knowledge resources we are using in this project are based on terminological information about the telecommunications domain which was approved and classified by an expert. A set of terms were extracted from the reference corpus and classified according to the three-level classification scheme developed in previous EUROTRA work (cf. [Alberto et al. 90]). Within this classification schema, CLASS1 defines the global domain, CLASS2 potential subdomains, for example, the telecommunications domain can be divided into the subdomains *Transmission, Antennae, Space Systems, etc.*. The CLASS3 defines the conceptual types, which are PROCESS denoting concepts, like *transmission* and *modulation,* EQUIPMENT describing instruments, like *modem,* but also devices, like *earth station,* PRODUCT containing *signal* and *data,* METHOD describing methods, like *access,* and PROPERTY characterising concepts belonging to one of the other classes, for example, *frequency* is a property

[2] This is the actual term used in the French version of the ITU corpus.

of signals, *antenna size* one of antennae. These concepts provide the fundamental basis for the conceptual organisation of the domain.

The most important class is that of PROCESS which contains the core concepts of the domain. Each process performs a specific function (often realised by verb phrases) by using a certain instrument which belongs to the class of EQUIPMENT. The whole process can be facilitated by a certain METHOD and has usually an input and output object which are conceptualised as PRODUCT. Besides these characteristics, that are more or less common for all processes, each process has in addition several discriminative properties (attributes).

The conceptual organisation of the domain can be derived from so-called *term definition forms* which serve as input to a term bank (EIRETERM) for the use of human users (translators) as well as a basis for the knowledge representation device in our project. Each term definition form contains: classification information, conceptual information and corpus attested linguistic information. The first information bundle gives a rough (general) classification of a term according to the classification schema described above; this kind of information is mainly of interest for pure terminology. The more important and useful knowledge for our purpose is represented in the second information resource. The features there contain information about the term itself, i.e. properties, for instance, *made_of* and *unit_of_measurement,* but also the relations to other concepts in the domain which will contribute to the construction of a conceptual subsumption lattice (the ontology of the domain). The features therein depict temporal (e.g. *precedes, follows),* causal (e.g. *for_purpose_of, affector)* and taxonomical relations (e.g. *has_properties, composed_of).* The third information bundle provides knowledge, such as lexical information, collocations and related terms, which is useful for the actual integration of extra-linguistic knowledge into the analysis and translation process; for instance, to keep the style of the text or to solve domain specific references (e.g. *also_known_as).*

2.2 Terminological Knowledge Bases

To translate a text human translators use specialised term lexicons or term banks in addition to a general language dictionary. Especially for texts belonging to a specific subject field (and which therefore exhibit a concentration of terms) the use of an additional lexicon containing information about these terms is absolutely essential. Although, they provide linguistic information about terms (e.g. synonyms, homonyms, equivalents in other languages) the conceptual information is often limited to verbal definitions or contexts. This information is not sufficient to support a computational treatment of NL translation: the conceptual knowledge has to be made computationally operatable, since this kind of knowledge expresses the relations between different concepts of a certain subject field.

Therefore, new directions in terminology research turn their efforts to the creation of terminological knowledge bases (TKB) instead of term banks (cf. [Meyer et al. 92]); this is in parallel to the design of lexical knowledge bases (LKB) in computational lexicology which, contrary to the conventional lexical databases permit *generalisation* and *inferencing* capabilities, and the possibility of *dynamically extending the lexicon.* In contrast, a TKB must make explicit the knowledge (of a domain expert) about concepts denoted by *specialised* lexical items (terms).

TKBs have advantages over LKBs and over the classical terminological lexica: the information is encoded explicitly, they allow for an explicit representation of conceptual relations and they facilitate consistency. Conventional term banks are handicapped by their *term-to-concept* orientation: knowing a term, the term bank provides its meaning, synonyms etc. but nothing else. Using a knowledge base goes the other way around, it is *concept-to-term* oriented, i.e. also the relations to other concepts are provided. This can help to get a better translation, because translation would be independent of any structural requirements. The correct structure will be generated on the target side, and depends only on the syntactic realisation of the corresponding term in the target language. In addition, TKBs provide mechanisms to assist the acquisition and systematisation of information.

Building such a terminological knowledge base is the objective of a research and development project undertaken at the University of Ottawa. COGNITERM ([Meyer et al. 92]) is designed as a hybrid

between a conventional term bank providing all structural linguistic information and a knowledge base where each concept is represented similar to frames and arranged in inheritance hierarchies. The system focused on the domain of optical storage technologies is a bilingual (English/French) TKB which is constructed by using the knowledge engineering tool CODE ([Meyer/Skuce 90]). By contrast, in our project the same representation formalism is used for the linguistic information and for the terminological/conceptual information. This obviously is an advantage for the integration process, since different knowledge sorts, i.e. linguistic and domain knowledge, are modelled through the same notational device, and no specialised interface between different knowledge resources has to be implemented.

2.3 Description of the Project's Knowledge Representation Device

The more generalised approach used in our project goes beyond merely conceptual representation, i.e. it takes into account the relationship between the conceptual organisation of the domain and the linguistic realisation in terms of domain-specific (text) situations.

These requirements establish three interrelated problem areas. The first problem is how knowledge of the world, be it general knowledge or specialised knowledge concerning a particular domain, like the field of telecommunications in the project's case, is to be represented. The second problem is how such organisations of knowledge are to be related to linguistic system levels of organisation such as grammar and lexis. The third problem is how these knowledge organisations are being made to operate in a natural language processing system. For the knowledge organisation (problem 1 and problem 2) the concept of ontologies for NLP has been suggested to be of potential value. Very generally, an ontology offers a 'conceptual' framework for the representation of data, information and knowledge. This framework then is sufficiently general, but also sufficiently detailed, to provide a rich supportive backbone for the construction of models of the world.

Several types of ontologies (cf. [Bateman 1992]) have being assessed and evaluated for the special purpose of the ET-10/66 project. In doing this, we had to bear in mind that the ontological knowledge we intend to use in the project has to be related to grammatical and lexical knowledge formulated in the ALEP formalism. Since this formalism makes use of constructions known from knowledge representation languages, such as KL-ONE and its descendants, and typed feature logics (e.g. [Carpenter, 1992]), this has proven not to be a major task. As a basic model for the representation of ontologies we have assumed a *subsumption lattice over concepts* (here: used in terms of types, sometimes called sorts) with a mechanism corresponding to structured inheritance of attribute information associated with the concepts, and probably with additional axioms or particular inferences, licensed by specific combinations of types (cf. figure 1).

To be more or less application independent and because the relation to linguistic information the knowledge base should contain all relations (the expert has designated), since each of the connected concepts can take over a special linguistic function which can be in turn realised through particular PPs or NPs. PPs are often introduced by special prepositions corresponding to this function, so instruments by *by,* locations by *at* and purposes by *for.* Besides these temporal, causal and spatial relations between the concepts of the individual classes, there are also interrelations between concepts belonging to the same class: *transmission,* for instance, is often preceded by *modulation* or *amplification.*

The ALEP formalism ([Simpkins 92]) provides for the design of the project's ontology a **type system facility.** The underlying principles force the development of a system which has to be strongly typed, i.e. each type is denoted by a unique name and all its appropriate attributes as well as the types of the possible values have to be specified. Currently, ALEP provides, beside user specifiable types, four built-in types: &atom, &list, &term and &boolean. Over the latter type all boolean operations (conjunction, disjunction and negation) can be formulated. The term type offers the possibility to have as values Prolog style term expressions, e.g. *exists(X).*

The *subtype of/supertype of* relations allow the declaration of generic types (supertypes) which can be further specialised by having subtypes.These relations which can be directly extracted from the

```
transmission:process
atts input                ⇒ signal,
     output               ⇒ signal,
     manner               ⇒ &boolean{pre-assigned/demand-assigned/occasional/continuous},
     means                ⇒ &boolean{satellite,cable},
     preprocess           ⇒ &boolean{modulation/amplification/down_conversion},
     postprocess          ⇒ reception,
     tmanner              ⇒ &boolean{sequential/simultanous/multi_channel/single_channel},
     frequency            ⇒ &atom _,
     modulation_method    ⇒ &atom _,
     power_density        ⇒ &atom _,
     polarisation         ⇒ &atom _,
     bandwidth            ⇒ &atom _
subs digital_transmission/ analogue_transmission.

digital_transmission:transmission
vals input          ⇒ digital_signal
atts bit_rate       ⇒ &atom _,
     used_by        ⇒ digital_system,
     synchronous    ⇒ &atom yes.

analogue_transmission:transmission
vals input      ⇒ analogue_signal
     frequency  ⇒ carrier_frequency,
atts used_by    ⇒ analogue_systems.
```

Figure 1: *Part of the Ontology*

ISA-feature of the term definition forms introduce a taxonomical hierarchy (isa-hierarchy) of the domain's conceptual structure. The feature structures for the type declarations are based on those found in the corresponding term definition forms.

The ontology, which due to its complexity is still under construction, is mainly process-based, this means around the domain's main concepts transmission, reception, modulation and multiplexing, the corresponding concepts to describe them are added. Generally, each of these processes has an *input*- and *output* feature whose values are always a kind of signal, the instrument used to perform the action *(used_by),* list of possibly preceding *(preprocess)* and/or following processes *(postprocess)* and the manner how the process can be done *(manner)*. Every type declaration for a process contains, besides these more general features, some which describe properties that are inherent to a special process, like *means, polarisation* for *transmission* or *channel_capacity* for *multiplexing* etc. In figure 1 examples for the process type declaration are given. The other types, those that are used for products, equipments and methods, are described in the same way.

For the different kinds of relations in the domain (described in [Pearson (ed.), 1992]) the current version of the ALEP formalism provides no particular data structure as, for instance KL-ONE like systems with *roles.* The only possibility to represent the relations explicitly is to express them by means of corresponding feature values. Thus they can't be described in more detail or constrained as it is possible in classical knowledge representation formalism. However, this can be partially overcome by introducing constraints in the respective grammar rule, e.g. for a correct PP-attachment (cf. below).

3 The Architecture of Analysis and Translation

3.1 General Outline

The general architecture (cf. figure 2) of our analysis module is based on the staged processing suggested in the ET6.1 study (cf. [Pulman (ed.), 1991]). There, analysis is composed of two steps: (1) a 'shallow' syntactic analysis for efficient parsing, and (2) a semantic refinement of the parsing result. In our approach we enhance the second step with a terminological component, thus achieving a semantic filtering and a domain-specific filtering of the parsing results. For parsing we have used a grammar and a lexicon for general language (including the domain's terms); for the refinement process (filtering) we have used a lexicon with general semantics and domain-specific information (where necessary); in this step the grammar remains the same. A further research issue is the use of an additional refinement grammar which might take into account sublanguage specific constructions as a further filtering mechanism (sublanguage constraints).

For the transfer module which has been designed for mapping German analysis output (so-called linguistic structure) to English synthesis input, we have adopted the option foreseen in ET6.1 (and. thus, in ALEP) that translation may be called on a specific type contained in the top-most feature structure of the input linguistic structure, i.e. the semantic and terminological (sub-) feature structures. Compared to the German analysis module, the German-to-English transfer module as well as the English synthesis module have a very limited coverage. This is mainly due to the fact that the focus of the project was/is on the conceptual organisation of the domain, and, on the other hand, that the semantic representation is currently too restricted, i.e. it lacks functional semantic information which is necessary for the generation of, for example, determiners, auxiliaries or complementisers.

3.2 Type system and information distribution

The top-most type 'sign' of the analysis type system has an additional entry point (attribute) 'term' for the description of terminological information, thus 'sign' is defined as:

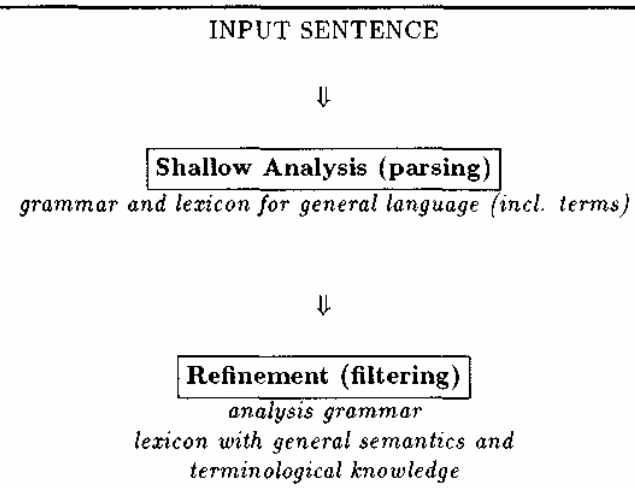

Figure 2: *The General Architecture of Analysis Module*

```
sign
atts  pho  ⇒ pho_fs,  % phonological (here: graphemic) information
      syn  ⇒ syn_fs,  % syntactic information
      sem  ⇒ sem_fs,  % semantics
      gap  ⇒ gap_fs,  % gap threading information
      term ⇒ term_fs, % terminological information
```

The attribute *term* takes a value of type *term_fs,* a feature structure describing relevant terminological information derived from the ontology of the domain, the term definition forms and further domain-specific linguistic information. The parsing grammar and lexicon just open the term slot without any real specific information besides the actual information distribution; the necessary terminological information is described in the refinement lexicon (cf. fig. 2) and used within the semantical and domain-specific filtering. Figure 3 shows the German grammar rule for prepositional phrases and the propagation of terminological information (expressed by the variable **TERM**).

Terminological information is a kind of additional semantic information and can therefore be integrated into the *sem_fs* feature structure. For the purpose of easier experimenting with different sorts of terminological information, i.e. other conceptualisations, we have separated the feature structures. A second advantage of this separation is the possibility of exchanging different domain dependent descriptions, thus enhancing the reusability of the module.
Examples of the terminological information used in the analysis module are shown in figure 4, figure 5 and in figure 6 (cf. section 2).

According to the domain analysis (cf. section 2) we have designed an abstract, reusable information and classification lattice (based on the type facility of ALEP) which is constrained by the language of the domain. These constraints are expressed in the refinement lexicon of the analysis module. For the analysis we have not used the full power of the domain's ontology; we have extracted those parts which serve as an interface between (general) linguistic realisations and the domain-specific use of language. This is mainly due to the fact that the ontology of the domain (cf. [Ripplinger (ed.), 1993]) is a well structured information resource for the telecommunications field from which task specific knowledge can be extracted, for example, for term recognition, analysis, translation and generation, in order to minimise the actual information search space for a specific task.

```
p_max =
sign:{
        syn=>pred:{
                bar=>max,
                full=>yes,
                head=>HEAD,
                subcat=>[]},
        gap=>GAP,
        sem=>SEM,
        term=>TERM}
->[
sign:{
        syn=>pred:{
                bar=>zero,
                head=>@HEAD p:{
                        p_compl=>P_COMPL},
                subcat=>[X]},
        sem=>SEM},
@X sign:{
        syn=>pred:{
                full=>yes,
                head=>P_COMPL},
        gap=>GAP,
        term=>TERM}
] head 1.
```

Figure 3: Grammar Rule for Prepositional Phrases

The selection and the extraction of the task related information from a domain's ontology can be triggered by the results of a (linguistically motivated) statistical analysis of text corpora about the domain (cf. [Brustkern, 1993]). For our purpose we have used information that is encoded in the ontology of the domain and sublanguage relevant information which we extracted from a limited statistical analysis of parts of the German ITU corpus and from the domain specific information we got from the statistical analysis of the English corpus.

3.3 Analysis Results

The integration of the entire semantic and terminological information into the parsing process, i.e. no separation into parsing and refinement, results in an extreme run-time increase[3] because all information is used for building up the analysis structures (bigger information search space and worse back tracking behaviour). The separation of information into syntactic parsing information and semantic (terminological) refinement information, as used in our approach, has proven to be the fastest approach for analysis within the ALEP framework.

We exemplify our analysis approach with the following German sentences:

1. Das Gerät adaptiert den Eingabestrom digitaler Datenbits.
   (The equipment adapts the input stream [of] digital data bits.)
2. Das Gerät adaptiert den Eingabestrom digitaler Datenbits für die Übertragung.
   (The equipment adapts the input stream [of] digital data bits for [ ] transmission.)
3. Das Gerät adaptiert den Eingabestrom digitaler Datenbits für die Übertragung durch den Modulator.
   (The equipment adapts the input stream [of] digital data bits for [ ] transmission through the modulator.)
4. Das Gerät adaptiert den Eingabestrom digitaler Datenbits für die Übertragung über einen Satelliten durch den Modulator.

[3] The average runtime increase can be computed by the following formula: $(t_1 \times R_{sem}) \times 2$, where t_1 is the parse time without semantic distinction and R_{sem} is the number of semantic variants; that means for each additional semantic distinction the runtime doubles.

```
satellit ~
sign:{
        pho=>pho_fs:{
                string=>[satellit|Rstr],
                rest=>Rstr},
        syn=>pred:{
                bar=>zero,
                r_periph=>undefined,
                head=>n:{
                        cat=>noun,
                        congr=>(mas&sg&p3&nom)},
                subj=>[],
                subcat=>[]},
        sem=>sem_fs:{
                gov=>n_sem:{
                        pred=>satellit,
                        v_mod=>no,
                        n_mod=>_,
                        n_class=>_,
                        n_props=>_},
                args=>_,
                mods=>mods_fs:{
                        mod_list=>[]}},
        gap=>gap_fs:{
                in=>GAP,
                out=>GAP},
        term=>term_fs:{
                  t_descr=>t_sem_fs:{
                           t_class=>_,
                           mwt=>no,
                           t_props=>_},
                  t_roles=>_,
                  t_mods=>t_mods_fs:{
                          t_mod_list=>[]}}}.
```

Figure 4: *Lexicon Entry for the Parser*

```
satellit
sign:{
        pho=>_,
        syn=>pred:{
                bar=>zero,
                r_periph=>undefined,
                head=>n:{
                        cat=>noun,
                        congr=>_},
                subj=>[],
                subcat=>[]},
        sem=>sem_fs:{
                gov=>n_sem:{
                        pred=>satellit,
                        v_mod=>no,
                        n_mod=>_,
                        n_class=>common,
                        n_props=>n_props_fs:{
                                n_abstraction=>concrete,
                                n_animacy=>artificial,
                                n_struct_prop=>struct_prop_fs:{boundedness=>count,
                                        homogeneity=>inhomogeneous,
                                        complexity=>individual,
                                        granularity=>nil},
                                n_temp_prop=>nil_temp,
                                n_spat_prop=>spat_prop_fs:{
                                        shape=>sh_realisation,
                                        norm_or=>nil,
                                        intr_or=>nil}}},
                args=>zeroval,
                mods=>mods_fs:{
                        mod_list=>[]}},
        gap=>gap_fs:{
                in=>GAP,
                out=>GAP},
        term=>term_fs:{
                  t_descr=>t_sem_fs:{
                           t_class=>telecommunications,
                           t_props=>t_props_fs:{t_prop=>satellite:{
                                                        used_in=>transmission,
                                                        input=>signal,
                                                        location=>orbit,
                                                        orbit_pos=>_,
                                                        transponder_pos=>_,
                                                        operation_pos=>_},
                                                t_v_mod=>no,
                                                t_n_mod=>_}},
                  t_roles=>z_val,
                  t_mods=>t_mods_fs:{
                           t_mod_list=>[]}}}.
```

Figure 5: *Lexicon Entry for Refinement*

```
adaptieren ~
sign:{
        pho=>_,
        syn=>pred:{
                bar=>zero,
                r_periph=>undefined,
                head=>v:{
                        cat=>verb},
                subj=>[
                        sign:{
                                syn=>pred:{
                                        bar=>max,
                                        head=>n:{
                                                congr=>nom},
                                        subcat=>{]},
                                sem=>ARG1}},
                subcat=>[
                        sign:{
                                syn=>pred:{
                                        bar=>max,
                                        head=>n:{
                                                congr=>acc},
                                        subcat=>[]},
                                sem=>ARG2}]},
        sem=>sem_fs:{
                gov=>v_sem:{
                        pred=>adaptieren,
                        predtype=>action},
                args=>bival:{
                        arg1=>arg_fs:{
                                role=>agent,
                                arg_sem=>ARG1},
                        arg2=>arg_fs:{
                                role=>affected,
                                arg_sem=>ARG2}},
                mods=>mods_fs:{
                        mod_list=>[]}},
        gap=>gap_fs:{
                in=>GAP,
                out=>GAP}}.

        term=>term_fs:{
                  t_descr=>t_sem_fs:{
                            t_class=>telecommunications,
                            t_props=>t_props_fs:{
                                      t_prop=>modulation:{
                                                input=>signal,
                                                output=>carrier,
                                                used_for=>transmission,
                                                used_by=>_}}},
                   t_roles=>t_tetra_val:{
                            t_role1=>t_role_fs:{
                                      t_role=>equip},
                            t_role2=>t_role_fs:{
                                      t_role=>product},
                            t_role3=>t_role_fs:{
                                      t_role=>process},
                            t_role4=>t_role_fs:{
                                      t_role=>method}},
                    t_mods=>t_mods_fs:{
                              t_mod_list=>[]}}}.
```

Figure 6: *Lexical Verb Entry for Refinement*

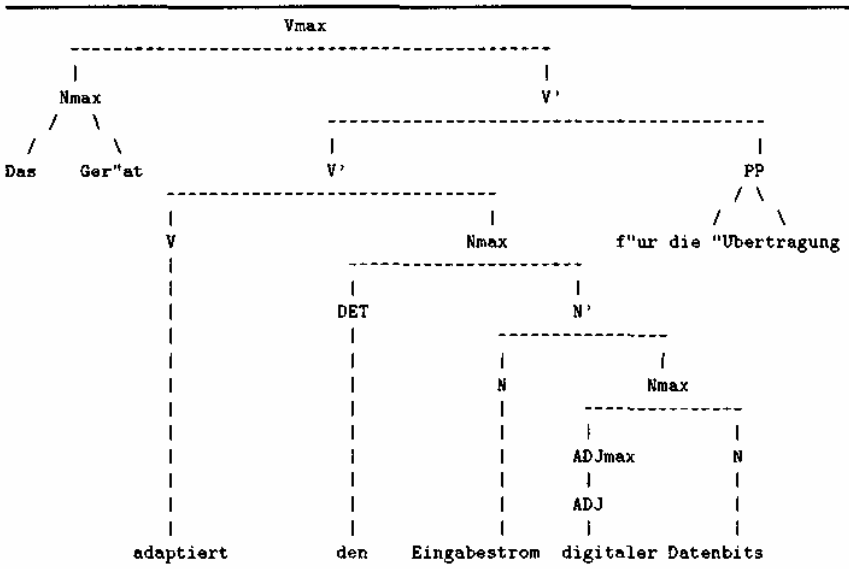

Figure 7: *Syntactic Structure of Sentence (2)*

(The equipment adapts the input stream [of] digital data bits for [ ] transmission via [ ] satellite through the modulator.)

The different semantic readings for the sample sentences are due to the fact that the semantic information allows for different semantics based descriptions of the PP-attachment variants which are syntactically valid. These attachments can only be constrained by additional domain-specific information (or further semantical restrictions). For example, according to our analysis module the prepositional phrase 'für die Übertragung' (for [ ] transmission) in sentence (2) can be attached to the verb 'adaptieren' (adapt), to the noun phrase 'den Eingabestrom' (the input stream) and to the noun phrase 'digitaler Datenbits' (digital data bits), thus resulting in three readings as opposed to possible five readings, which is caused by the fact that both NPs are not allowed to function as a modifier of the verb. This is also reflected in the number of objects for sentence (1) (one object). In the terminological description we have restricted the PP to belong to the verb's frame in the telecommunications domain.
Similarly we have restricted the PP 'durch den Modulator' (through the modulator) in sentence (3) as the means (method) of adapting a signal in the domain. Whereas in sentence (4) we have allowed the PP 'über einen Satelliten' (via [ ] satellite) to belong to the verb's conceptual frame as the purpose of adapting a signal and to modify the PP 'für die Übertragung' (for [ ] transmission), thus, the analysis produces two readings of the sentence.

Figure 7 shows an abstract syntactic structure of sentence (2) as used in the German analysis module. According to the binary branching strategy used in X-bar syntax based descriptions, the reader easily can imagine the other possible syntactic structures.

3.4 The Transfer Module

Within the transfer module there is one rule for initialising the translation process as shown in figure 8. Once translation is called on the semantic and the terminological (sub-) feature structures specified as the value of the linguistic structure's top-most 'sem'- and 'term'- attribute, translation is called recursively on type 'sem_fs' and on type 'term_fs', and all subordinate types respectively. When translation is called on type 'sem_fs' the predicate string specified by the 'pred'-attribute

```
trans_init *
(de<=>en)
sign:{
     syn=>pred:{
             bar=>max,
             punct=>yes,
             head=>v,
             subj=>[],
             subcat=>[]},
     sem=>SEM1,
     term=>TERM1} < _ ,
sign:{
     syn=>major:{
             bar=>s,
             punct=>yes,
             head=>v,
             subj=>[],
             subcat=>[]},
     sem=>SEM2,
     term=>TERM2}
[ de:SEM1 =* en:SEM2,
  de:TERM1 == en:TERM2].
```

Figure 8: *Start Rule of Translation Process*

```
trans_pred_senden =
(de<=>en)
sem_fs:{
     gov=>v_sem:{
             pred*>senden,
             predtype=>PT1},
     args=>@ARGS1 trival,
     mods=>MODS1},
sem_fs:{
     gov=>v_sem:{
             pred=>send,
             predtype*>PT2},
     args=>@ARGS2 trival,
     mods=>MODS2}
[ de:ARGS1 == en:ARGS2,
  de:MODS1 =* en:MODS2,
  de:PT1 = en:PT2].
```

Figure 9: *Translation Rule for 'senden' (semantic dimension)*

within the governor feature structure is translated from one language into the other. This is illustrated by the rule in figure 9 which translates the German predicate string *senden* into the corresponding English predicate string *send.*

For the translation of the appropriate terminological information rules for the different conceptually dependent arities are used. In the case of 'senden' the rule in figure 10 will be applied.

This approach allows for a straightforward account of instances of complex transfer where changes have to be performed according to the argument structure of the predicate that has to be translated (this applies to 'sem_fs' only).

A domain-specific role structure of a concept, identified by the terminological attribute 't_roles', is translated by a rule dedicated to the relevant subtype of type 't_role_fs'. For instance, the role structure assigned to the predicate *senden* is translated by a rule operating on the subtype 't_trival' and calling recursively for translation on type 't_role_fs' which is the type assigned to the roles of a concept (cf. figure 11).

Type 't_role_fs' will, then, be translated by a rule which, in turn, calls for translation on type 'term_fs'

```
trans_term_trival =
(de<=>en)
term_fs:{
     t_roles=>@TARG1 t_trival,
     t_mods=>TMODS1}
term_fs:{
     t_roles=>@TARGS2 t_trival,
     t_mods=>TMODS2}
[ de:TARGS1 == en:TARGS2,
  de:TMODS1 == en:TMODS2].
```

Figure 10: *Translation Rule for Terminological Information*

```
trans_term_args_trival =
(de<=>en)
t_trival:{
     t_role1=>TARG1d,
     t_role2=>TARG2d,
     t_role3=>TARG3d},
t_trival:{
     t_role1=>TARG1e,
     t_role2=>TARG2e,
     t_role3=>TARG3e}
[ de:TARG1d == en:TARG1e,
  de:TARG2d == en:TARG2e,
  de:TARG3d == en:TARG3e ].
```

Figure 11: *Translation Rule for Conceptual Roles*

again, since the value of 't_role_sem' is a terminological feature structure. Accordingly the semantic argument structure is translated.
The translation of the modifier-list of a concept (in 'sem_fs' and 'term_fs'), finally, is performed by distinct rules with each of them accounting for a specific number of elements specified in the modifier list (including the empty modifier list).

3.5 The English Synthesis Module

The English synthesis module has been designed on the basis of the German analysis module; it does not yet account for the insertion of auxiliaries, complementisers, or determiners. The basic reason for this limitation is that the functional semantic information which is conveyed by these elements is currently not reflected in the semantic representation serving as synthesis input. At present, the English synthesis module covers a very limited range of phrase structural complexity concerning NP, AP or ADVP structure.

Ideally, the basic sign feature structure and, more specifically, the semantic and terminological feature structures should be the same for all languages. With this assumption, it was only the syntactic feature structure which has been revised in designing the type and feature specification for the English grammar.

Since no refinement is applied in synthesis, the English synthesis grammar operates on a single lexicon which contains fully specified lexical entries including terminological information too (i.e. specific syntactic realisation information). The integration of a refinement stage in synthesis could, in principle, support an application for speech processing (here: speech synthesis).

4 Conclusions and Perspectives

In this paper we have briefly outlined work ongoing in a project that aims at integrating terminological knowledge into the NL analysis and translation process. With our approach it is also possible to solve some of the problems also arising in more general approaches in computational linguistics, namely the support of NLP/MT by different information and knowledge resources. In this sense our approach constitutes a specific instance of control.

The work is based on factors and assumptions that were to some extent discussed elsewhere, especially in the field of knowledge-based machine translation; our contribution is to apply some of these findings to computational terminology, and to implement a demonstrator system within a constraint-based NLP/MT framework for a specific application domain. The main difference to existing knowledge-based systems in computational linguistics is that we have used the same formal devices for expressing and representing different sorts of knowledge, and, thus, we have avoided the explicit design of an interface between these knowledge sorts.

A future research/application field is the attempt to amend and extent the existing demonstrator and to couple it with a term recognition and classification module (cf. [Schütz, 1992]).

References

[Alberto et al. 90] Alberto P.F., B. Andersen, B. Barbone, D. Kenny, A. Michiels, J. Pearson, H.S. Sorensen, J. Vollmer 1990. Terminology in Eurotra. Internal EUROTRA document. CEC, DG-XIII, Luxemburg.

[Bateman, 1992] J. Bateman, 1992. The Theoretical Status of Ontologies in Natural Language Processing. In: Proceedings of the International Workshop on Text Representation and Domain Modelling, TU Berlin.

[Brustkern, 1993] J. Brustkern. 1993. Statistical Analysis of the ITU Corpus. In: Ripplinger (ed.), ET-10/66 Report 3, Luxembourg.

[Devillers (ed.), 1991] C. Devillers (ed.), 1991. ET-6/3 Final Report. CEC, DG-XIII, Luxembourg.

[Meyer et al. 92] Meyer, I., L. Bowker, K. Eck, 1992. Cogniterm: An Experiment in Building a Terminological Knowledge Base. In: Proceedings of the Fifth Euralex International Congress, 1992.

[Meyer/Skuce 90] Meyer, I., D. Skuce, 1990. Concept Analysis and Terminology: A Knowledge-Based Approach to Documentation. In: Proceedings of COLING'90.

[Pearson (ed.), 1992] J. Pearson (ed.), 1992. ET10/66 'Terminology and Extra-linguistic Knowledge', Report 1. CEC, DG-XIII, Luxembourg.

[Pulman (ed.), 1991] S. G. Pulman (ed.), 1991. ET-6/1 Final Report. CEC, DG-XIII, Luxembourg.

[Ripplinger (ed.), 1993] B. Ripplinger (ed.), 1993. ET10/66 'Terminology and Extra-linguistic Knowledge', Report 3. CEC, DG-XIII, Luxembourg.

[Schütz (ed.), 1991] J. Schütz (ed.), 1991. ET-6/2 Final Report. CEC, DG-XIII, Luxembourg.

[Schütz et al., 1991] J. Schütz, G. Thurmair, R. Cencioni, 1991. An Architecture Sketch of Eurotra-II. In: Proceedings of MT-Summit III, Washington, D.C.

[Schütz, 1992] J. Schütz, 1992. Advanced Multi-lingual Term Recognition and Classification Toolbox. Internal Research Paper, IAI, Saarbrücken.

[Simpkins 92] N. K. Simpkins. 1992. ALEP-0 Version 2.2 - Prototype Virtual Machine, User Guide. CEC DG-XIII, Luxembourg.

[Sowa, 1991] J. F. Sowa, 1991. Towards the Expressive Power of Natural Language. In: J. F. Sowa (ed.). Principles of Semantic Networks: Explorations in the Representation of Knowledge. Morgan Kaufmann Publishers, San Mateo, CA.

[Zajac 92] R. Zajac. 1992. Inheritance and Constrained-based Grammar Formalisms. In: Computational Linguistics, 18(2), p. 159-182.