Iman Thabet, Bernd Ludwig, Frank-Peter Schweinberger, Kerstin Bücher,Günther Görz

Using EuroWordNet within the Speech Operated System EMBASSI

Abstract

In natural language processing, incremental semantic composition is one of the most prominent issues. In the past, numerous approaches have been developed for assigning meaning to noun and verb phrases and their complements and modifiers. However, their inferential power is often too weak to be applied to practical applications, or the expressiveness of the representation language is so complex, that it leads to intractable inference procedures. As an answer to these problems, we have developed an approach that relies on Description Logics (DL) for handling semantic construction. First, we will discuss this approach and show how a semantic knowledge base can be setup dependant on EuroWordNet[1] (EWN) as a linguistic ontology. Subsequently we will outline our experience with and demands on EWN.

1 What is EMBASSI and its Objective?

Embassi („Elektronische Multimediale Bedien- und Service-Assistenz“) has been a German joint project sponsored by the German Fed. Ministery of Research[2].

Our contribution to this project consists mainly of three components:

- The dialogue manager,
- formal ontologies for several multilingual application domains, and
- the language generation component to communicate system utterances to the user.

The long-term goal of our research is to design and implement a generic dialogue system for rational (spoken) dialogues that helps a user to achieve certain goals in terms of operations of a technical application system – e.g. an information system, a system for controlling devices, or any other kind of problem solving systems. One of its design criteria is the ability to recognize users' intentions in order to establish corresponding subgoals and control their processing. Furthermore, it should enable mixed-initiative, flexible and cooperative conversations, provide a high level of robustness as well as scalability at the linguistic and application dimensions, and easy portability to new domains. In addition, it should be possible to integrate multilingual linguistic interaction with multimodal forms of input and output such as graphical user interfaces, and – by means of appropriate devices – the recognition of deictic actions.

2 DL Models of Applications

Applications are characterized by a DL terminology which models the concepts used for making propositions about application situations. Basically, Embassi's knowledge base is composed of two parts: the EWN ontology, which encodes the linguistic meaning of words determined on an empirical basis, and the Standard Upper Ontology (SUMO) (NP01), which is used as a generic base model for concepts of the application domain (see Ludwig 2002).

3 Semantic Construction

This section discusses the issue of semantic construction during analyzing natural language input. We are using an incremental approach to the composition of semantic representations. The backbone of our approach is λ-DRT (see Fillmore 1969). The parser builds Discourse

Representation Structures (DRSes, see KAMP & REYLE 1993) incrementally and maps them onto ABoxes[3] (see BÜCHER ET AL. 2002).

The main question here is how the mapping of domain independent - in terms of EWN - to application specific language usage - in terms of a domain model - is done. In the discourse domain, referents usually refer to instances in the application domain. Such pairs of a discourse referent and a corresponding instance are represented by means of a special role called **has-lex**. For instance, in the definition

$$\texttt{AvEvent} \sqsubseteq \forall\texttt{has-lex.Program1}$$

it is claimed that an `AvEvent`[4] is related to a discourse referent of `Program1`. Consequently, all words that are assigned `Program1` as a meaning in EWN, designate an instance of `AvEvent` in the application domain. The DRS:

$$\left[\begin{array}{l} \quad y \\ \hline \texttt{AvEvent}(\ \ ) \\ \texttt{Program1}(y) \\ \texttt{has-lex}(\ \ , y) \end{array}\right]$$

can be mapped onto an ABox asserting:

$$\texttt{Program1}(y) \wedge \texttt{has-lex}(\ \ , y) \wedge \texttt{AvEvent}(\ \ )$$

The set R of possible application specific readings of an instance of `Program1` is the set of all concepts (in the application domain) subsumed by the concept

$$\forall\texttt{has-lex.Program1}$$

Except in trivial cases, a direct mapping from the user utterance to a system command cannot be accomplished. In general, we have to take complex speech acts into account, where the interpretation of the utterance's propositional content is determined by its (local) linguistic-pragmatic context in the first place. This, in turn, is to a large extent influenced by (global) discourse-pragmatic features which provide constraints based on the dialogue history and the actual place of the utterance in the dialogue as being, for instance, the expected answer to a question. In addition, the application provides further constraints to limit the possible meanings of words and phrases to their particular use within the given thematic framework. Therefore, we have to distinguish between several interleaved levels of analysis of user utterances:

- Linguistic analysis on the utterance local level, which in turn consists of several levels of syntactic and semantic construction;
- Semantic evaluation, i.e. evaluation of semantic operators (e.g.disjunction and conditional expressions), reference resolution, and additional transformations of the logical form, augmented by specific computations;
- Application specific constraints on the evaluated semantic representation (see LUDWIG 2002);
- Discourse-pragmatic analysis, i.e. determining the underlying speech act and accordingly the user's intention - a proper function of the dialogue manager.

4 Two Parsing Phases

If we want human-computer-dialogues to be natural, we must enable humans to talk to the computer as they do to humans. However, spontaneous speech is often incomplete or incorrect, full of interruptions and self-corrections leading to an ungrammatical input to the parser. Moreover, speech recognizers themselves may produce ungrammatical output even with correct input. Apart from this, parsing German input is difficult because of its fairly free word order and discontinuous constituents. Therefore the grammar cannot rely only on a linear sequence as its main concept.

We tried to overcome these problems by designing a two-phase parsing process (as presented in Bücher 2002). The first phase works with a grammar that employs phrase structure rules to build small phrases called chunks (similar to Abney 1991). Assigning semantic representation to the chunks also takes place in this phase. In the second phase, the interpretation of the whole utterance is derived by relating these chunks and their interpretation to each other.

4.1 Phase 1: Determining the Semantics of Chunks with the help of EWN

In order to ease the adaptability of the dialogue system to different domains and to reflect general and domain independent usage of language as distinct from that of a specific application, the semantics of the chunks is expressed in terms of concept expressions taken from EWN.

In this context, we would like to point out that Embassi is a multilingual system, so we depend not only on GermaNet[5] but also on EWN in our research. Also because the size of EWN is bigger than that of GermaNet we used to search for definitions in EWN if they were missing in GermaNet.

A chunk may consist of either only one element which is normally the head of the chunk C_h, or of a head element and one or more constituents that can be possible fillers of a free position in the head's structure.

After the categorization of the constituents the parser tries to build the chunks by combining the constituents pairwise:

$$C \rightarrow C_1\, C_2$$

The filler is usually a specifier (the determiner in case of a noun phrase *NP* or a modifier (e.g. a prepositional phrase *PP* modifying a verb).

If the chunk consists of only one constituent $C \rightarrow C_1$, which is the head of the chunk and therefore a terminal lexical category, we get the semantics of C from the lexicon, where the semantic information is stored as a λ-DRS (Kuschert 1996) (also referred to as the *extension* of the constituent). If C_1 is an expanded category[6] it contains the head of the chunk, and the semantics of C is derived from C_1. So, if there is only one symbol on the right side of the grammar rule, then the *extension* of the left side is determined as follows:

$$\mathrm{ext}(C) := \mathrm{ext}(\mathrm{head}(C))$$

In the case of a chunk consisting of a head C_h and another constituent C_f ($h \neq f \in \{1,2\}$), C_f is related to the discourse referent of C_h by a role R either taken from the inventory of EWN (see Vossen 1999) or defined by us[7]. Syntactically, the combination of two categories to a chunk is determined by a grammar rule which relates the two constituents via the role R. We then get the *extension* of the chunk by λ-composition of the DRSes of both constituents. In this case, the semantic head of the chunk is the one of its DRS:

$$\mathrm{head}(C) := \mathrm{head}(\mathrm{ext}(C))$$

When combining two elements, the parser checks the compatibility of the morphological features (e.g. agreement in case of the combination of a determiner with a *NP* and merges their DRSes resulting in a new DRS for the chunk. In this way, each chunk is assigned an interpretation already at this early stage. This has the advantage that if no further parsing is possible we thereby have means to interpret the whole utterance chunk by chunk.

To illustrate this, consider the utterance "*Kommt Tatort im ZDF?*"[8] taken from our Embassi application domain: To combine the preposition *im* and the *NP* chunk *ZDF* which was built using the (*NP* →*EN*)-rule we apply the following *PP*-rule[9]:

PP: P NP:
head = P:
role = has-value:
P morphfeat position = prepos,
P morphfeat kasrek = NP morphfeat case,
PP vpsynfeat clausetype =
NP vpsynfeat clausetype,
PP = P:

The *PP*-rule contains syntactic as well as semantic information about the chunk-combination. The DRS for the *PP*-chunk is constructed by the use of λ-composition of the DRSes of *ZDF* and *im* obtained from the lexicon:

$$\left[\frac{i}{\texttt{im-SP}(i)}\right] + \left[\frac{l}{\begin{array}{l}\texttt{TVStation1}(l)\\ \texttt{has-name}(l, ZDF)\\ \texttt{Name}(ZDF)\end{array}}\right] + \left[\frac{\emptyset}{\texttt{has- alue}(i,l)}\right] = \left[\frac{i\,l}{\begin{array}{l}\texttt{TVStation1}(l)\\ \texttt{has-name}(l, ZDF)\\ \texttt{Name}(ZDF)\\ \texttt{im-SP}(i)\\ \texttt{has- alue}(i,l)\end{array}}\right]$$

After having applied all phrase structure rules we get three chunks: The *NP Tatort*, the *PP im ZDF*, and the verb phrase *VP kommt*. Each chunk gets after this first phase a semantic interpretation on its own. The interpretation of the whole utterance is derived by relating these chunks and their interpretation to each other in phase two.

4.2 Phase 2: Applying Case Frames to Chunks

Phase two is different from phase one in that it combines chunks that needn't be adjacent to each other. Consequently, the order of the constituents is not relevant but may be an indicator for preferred readings when disambiguation is required. In this phase we use a kind of dependency grammar that determines for each chunk of phase one a list of possible syntactic functions it may have:

C_1 has C_2 → `<synfunc>`

<constraint equitation>

e.g.:

VP has *PP* → `adverbial`

NP has *PP* → `attribut`

VP has *NP* → `subject`

NP agr case = nom,

NP agr num = *VP* agr num.

The options are constrained by the morphological features of the chunk, e.g. a *NP*-chunk functions as a subject only if it has nominative case.

For the semantic head of each chunk there is a case frame[10] in which information about the valencies[11] are stored. The valencies of each chunk are filled by combining it with other chunks, e.g. building a *VP* from a verb and a *NP* that functions as its direct object, or expanding a *VP* by an adverb.

The suitability of the combination of two chunks is determined by the semantic constraints of the application domain. For example, consider the case frame for the verb *kommen* in the sense of "running a program":

infinitive: `kommen`

synt. function	thematic role	EWN concept
subject	agent:	Program1
adverbial	location:	TVStation1

From the case frame we derive hypotheses about possible fillers of a complement position of a chunk using the syntactic functions. Whether a hypothesis is satisfiable or not is determined by the concepts of the chunks. If they fit, the DRS can be computed: For a semantic head C_h, its complement C_k and a theta role R = thema (C_h, `synfunc`) that C_k can fill, we get the extension of the modified chunk $\tilde{C}_h$ as follows (h := head(C_h), k := head(C_k)):

$$\mathrm{ext}(\tilde{\ }_h) = \mathrm{ext}(\ _h) + \mathrm{ext}(\ _k) + \left[\frac{\mathrm{h\ k}}{\mathrm{thema}(\ _1, \texttt{synfunc})(d_1, d_2)}\right]$$

In our example, the *VP kommt* can be combined with the adverbial *PP im ZDF* since in the case frame of *kommen* there is a valency for an adverbial with the concept `location`. So we get

$$\left[\frac{i\ \ k}{\begin{array}{l}\texttt{Run}(k)\ \texttt{TVStation1}(\ )\\ \texttt{Name}(ZDF)\\ \texttt{im-SP}(i)\\ \texttt{has-location}(k,\ )\\ \texttt{has-name}(\ ,ZDF)\\ \texttt{has-value}(i,\ )\end{array}}\right]$$

After λ-composition of the DRS above with the DRS for *Tatort* we get a full DRS for our example utterance.

5 Building a Case Frame Database

In order to encode the semantics of a natural language expression in our DL-domain, we always had to search in EWN for this expression, and if it was found, we had to manually follow up the taxonomic chain until we arrived at a superconcept that was already defined in our domain, and then begin from that point to encode the subtree we expanded in the last step. This task is time consuming and can be a source of errors, like encoding some concepts with their trees more than once, or forgetting subconcepts within a chain, not to mention the typing mistakes, missing parentheses, etc. which makes the domain model inconsistent and the processing difficult or rather impossible.

Moreover we use our approach to semantics construction in different applications. Consequently we gathered a huge amount of semantic definitions (i.e. taxonomic chains) and case frames (i.e. thematic roles) defined by these applications. Some of these data are specific to a given application, whereas others are used by several applications. This made the need for a tool that enables efficient storage and easy and fast access, as well as preparing the data required by the parser be of prime importance.

For this purpose, we have developed a lexicon tool that helps editing semantic data, checks their coherence according to the algorithm presented in sect. 4, and visualizes them as well (see fig.1).

The tool depends on the following resources as a basis for its data:

- EWN Ontology
- SUMO Ontology
- Semantic lexica

In this respect, it is worth highlighting the differences between our frame data base and FrameNet (see Baker et al. 1998). FrameNet is an online lexical resource[12] for English based on the principles of frame semantics and supported by corpus evidence. It can serve as a dictionary, for it includes definitions and grammatical functions of the entries. And hence entries are linked to the semantic frames in which they participate, FrameNet can serve as a thesaurus as well.

However, the information provided by FrameNet is not formal enough to be directly applicable to our system; in other words it is not possible to use FrameNet for parsing utterances directed to the system or constructing the semantic representation for them. So from the practical point of view, what we need is a formal specification for the information represented in FrameNet and which, on the one hand, can directly be encoded in DL notation, and on the other hand, can be used as an efficient inference mechanism. Another difference is that FrameNet is basically constructed for the English language and hence

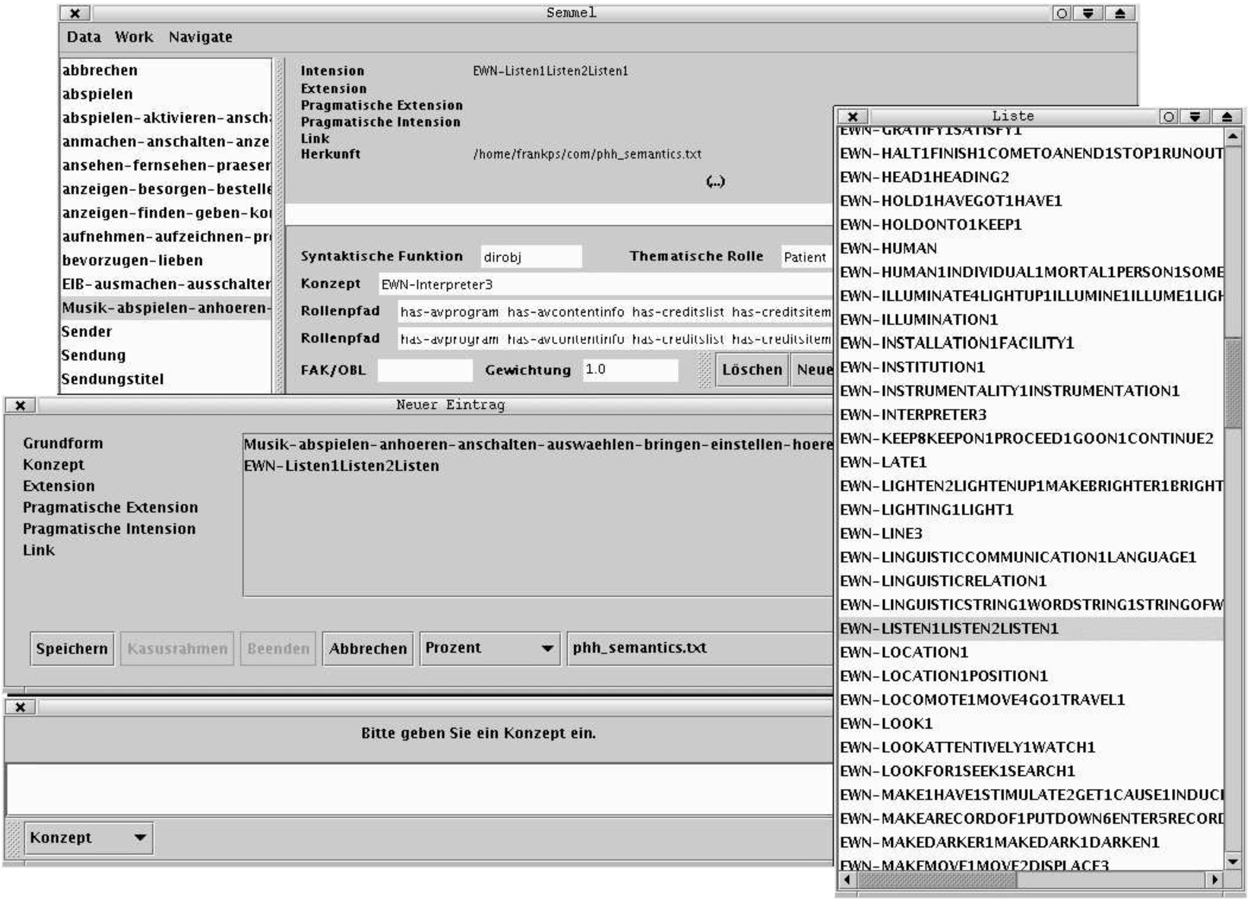

Figure 1: Lexicon-tool

can be used only in English based systems. Since our application is multilingual, our DL is based on the ILI-representation of EWN, which makes our tool language independent.

5.1 The Functionality of the Lexicon Tool

In our semantic lexica each entry has the following structure:

BASEFORM
LEXICAL INTENSION
LEXICAL EXTENSION
PRAGMATIC INTENSION
PRAGMATIC EXTENSION
CASE FRAMES
LINK
COMMENTS

The LEXICAL INTENSION refers to the lexical concept (as presented in EWN) that describes the lexical meaning of the entry, whereas the LEXICAL EXTENSION presents a DRS for the entry, and which has the following schema:

$$\left[\frac{x}{\textbf{lexical-intension}(x)}\right]$$

PRAGMATIC INTENSION and PRAGMATIC EXTENSION[13] provide optional information that can be used by the dialog system.

CASE FRAMES are the valencies assigned to the entry and which need to be filled by other instances that can satisfy the syntactic (e.g. subject), semantic (e.g. agent), and pragmatic ‚also called the thematic role' (e.g. user) constraints.

LINK refers to the name of the corresponding case frame.

The lexicon-tool can be considered as an interface between our application system and the semantic resources mentioned above, because, on the one hand, it stores the expressions used by the different applications and presents them as entries, to which the corresponding case frames are assigned and which are needed by the parser. On the other hand, it stores for each entry the underlying semantic concept as it is represented in EWN together with its taxonomic chain.

It can also be seen as a reproduction of the domain model due to three factors:

1. It maps the pragmatic intension of an entry onto the lexical one. This mapping is essential for determining the fillers (in the syntactic sense) or roles (in the pragmatic sense) specified by the case frame.
2. It maps the roles specified by the application domain onto the concepts obtained from the semantic lexicon. These roles must not violate the conceptual structure.
3. It verifies the consistency between the lexical and pragmatic intensions of the roles.

The interface provides an easy access to the stored information with the help of navigation tools like pop-up menus, text fields, lists, etc. It also enables the user to add new entries to the data base and define their word classes, syntactic functions, thematic roles, and semantic concepts (after obtaining it from EWN). While doing this the lexicon-tool offers lists with options that help the user determinig the most appropriate category by which the selected gap (text field) can be filled, and in the case of ill-formed or inappropriate input it returns detailed error messages with improvement suggestions.

One of the most valuable features of our lexicon-tool is the possibility of controlling and checking the coherence of entries both in terms of complete conceptual hierarchies with regard to our linguistic domain, and appropriate thematic roles with regard to the application domain. So if the user wants to check consistency or dependency relations between some concepts he can do that by typing the required sequence of concepts into the corresponding text field and getting the response after the check. Similarly, on adding new entries to the data base, if the given concept doesn't exist or collides with other concepts it won't be added, and subsequently the tool produces a corresponding error message and proposes possible solutions.

A further feature is the visualization of dependency relations in terms of links and cross references existing in the knowledge base. The possibility of checking whether a concept defined in a semantic lexicon specific to a given application also exists in the domain model or not remains to be done in future development.

6 The Influence of EWN on the Performance of EMBASSI

Our approach distinguishes between discourse and application domains, which in turn leads to a separation between linguistic and application specific knowledge. For this purpose, our knowledge base contains the complete SUMO ontology encoded in DL, the EWN upper ontology, and the concept definitions specific to Embassi applications. However, many SUMO and EWN concepts could be removed from the knowledge base as they were not used by the application specific part. So in an automatic precompilation step, we deleted 862 concepts, which were only defined but not a part of other definitions. This step improved the performance of Embassi a great deal. Nonetheless, the ratio of deleted concepts would be worse in more complex domains. As we didn't include the whole set of EWN concepts, in a more complex application the EWN portion even of the reduced knowledge base would be larger.

7 Experience with and Demands on EWN

In this section, some difficulties that we encountered while using EWN as the linguistic ontology in our knowledge base will be addressed. In the light of these difficulties, we will outline our strategy in dealing with them and consequently our demands on EWN. As EWN is the upper ontology in our system, most of the examples mentioned below are mainly taken from EWN, but the presented problems are valid for both EWN and GermaNet.

- **Missing parts of speech:** EWN is mainly limited to nouns, verbs and adjectives. However, meanings are not just expressed by these elements. Definitions for adverbs, temporal and spatial expressions, function words (e.g. auxiliary verbs, modal verbs, prepositions, etc.), not to mention multi-word elements (e.g. phrasal and prepositional verbs), idioms, collocations, and widely used abbreviations (e.g. ‚CO' for company) are generally not accounted for in EWN. Therefore we had to expand the linguistic domain model to include concepts for temporal and spatial expressions – to mention only the most prominent ones. It is evident that these elements are very important within the domain of Embassi in particular and similar systems in general, because, on the one hand, they function as fillers of roles in the application specific domain, which, in turn, helps determinig the sort of action to be triggerd off as a response to an utterance. On the other hand, in a language like German, prepositions, for instance, determine the case of the following noun. This fact can be used to enhance the mechanisms employed for disambiguation and sense differentiation. In conclusion, these elements are not optional but essential in any natural language system and can play a central role both on the semantic and application level, they shouldn't therefore be ignored in any tool for semantic representation.
- **Missing senses:** Another problem was the case in which the word being searched for already exists in EWN but not in all its senses. A definition of the word ‚part', for instance, in the sense of "member of a group" doesn't exist. Also, the word "subscribe" is only defined in the domain of financial transactions, so when we were searching for the same word in the sense of "being a member of or join" (a mailing list for instance) the corresponding definition couldn't be found. In such cases, we had to get the required sense by using synonymous words, despite the fact that the required word is already defined in EWN but not in all or at least not in the most dominant senses of it.
- **Conceptual gaps:** The definitions of some verbs (e.g. contain, glow, test, treat, sweat, apply, charge, ...) and most adjectives are so short, that they don't lead to the superset of all concepts that already exists in EWN. Concequently, gaps in the conceptual hierarchy may arise. In order to fill in the gaps in the hierarchy, we added general concepts like DO, CHANGE, CAUSE, STATE, QUALITY, MODAL-PROPERTY, MENTAL-PROPERTY and others to our knowledge base. On the one hand, these concepts function as subconcepts of already defined superconcepts in EWN, on the other hand, we can derive the required or rather the missing concepts from them.
- **Long taxonomic chains:** In contrast to verbs and adjectives, some nouns are assigned very long taxonomic chains (see, for example, the definitions of "mall", "tour", "cloth", "stuff"), which makes their encoding in DL and hence the consistency control rather difficult, not to mention the storage place and processing time they may take. We by-passed this problem by taking the definition of the underlying syno-

nym (marked by "=" in EWN), which usually has a shorter taxonomic chain. A side effect of this strategy is that some of the semantic properties of the word get lost, which leads to inacurracy in the semantic representation. Also the synonym definitions always imply a kind of generalization, which may be a source of ambiguity.

- **Antonymy:** Antonyms that can be regularly built by using some negation prefixes like (un-, in-, anti-, dis-,..), in general, are poorly represented in EWN. For example, the word "subscribe" exists but not "unsubscribe", the same holds for "scented" and other words. One would argue that antonyms shouldn't be accounted for in a lexicon, and their semantic representation can be obtained by negating the corresponding affirmative form. However, the negation of a form doesn't always reflect the meaning of the corresponding antonymous form (cp. unsubscribe vs. not subscribe). Apart from the processing perspective, within the foreign language teaching domain, a learner should be able to look up an antonymous form, or at least get information about how to build an antonymous form. In conclusion, it would be helpful, if EWN would pick up the most frequent antonyms either as separate entries or by assigning to every word the corresponding antonymous form or prefix.
- **Derivations:** Like antonyms, many standard[14] derivations are not existent in EWN. To illustrate this, take, for example, the word "moisturizer"; it is not defined, although the verb "moisturize" already exists. So the possibility to account for derivations either statically or dynamically in EWN is essential for building a uniform and balanced taxonomic hierarchy.
- **Insufficient syntactic coverage:** By "syntactic coverage" we mean syntactic features like valencies of a verb; gender, number of nouns, and so forth. Such features are not represented in EWN. In a system for natural language processing (e.g. machine translation system, parser, language generation system,..) these features are very essential not only on the syntactic but also on the semantic level.
- **Compounds:** Like derivations, there are only few entries for compound words in EWN, and there is no way to generate them dynamically. Examples: *Bruttopreis*, *Nettopreis*, *Schutzfolie*, a.o. In our application, we dealt with this problem either by combining the concepts of the individual constituents making up the compound expression, provided the constituents are already defined in EWN, or by searching for synonymous expressions, each consisting of a single word in order to take its definition as a substitute for the compound being actually searched for. The disadvantage of this method is that it makes the semantic construction more difficult and the semantic representation very complex and in some cases even inaccurate as well. This problem becomes more obvious in languages like English, where the constituents of a compound expression are separated by spaces. So it is sometimes difficult to recognize compounds as such. Therefore generating all possible conceptual combinations dynamically would be of a great advantage.
- **Orthographic variants:** As there are no uniform orthographic rules, it would be a big plus for EWN if it would account for possible orthographic variants of an expression like in (*email / e-mail*, *anti-perspirant / antiperspirant*, *Web-Seite / Webseite*), which will accelerate the search and retrieval.

8 Conclusion

The main goal of our research is the design and implementation of a generic dialogue system for spoken language that enables users to achieve specific tasks. This requires an efficient mechanism for incremental semantic construction, in which lexical data can be reused within different DL domains and by several applications. In our system, we have been using EWN as a lexical resource for modelling linguistic data. Our experience with EWN within Embassi showed that the encoding of lexical data in DL and processing them in real-time was so far possible.

However, the practical experience always yields new demands on lexical resources (see sec. 7) and open questions for discussion and further improvement as well. For example:

- Considering FrameNet, which data can be extracted and practically applied to NLP-systems?
- How can they be encoded so that they can be generally used by different systems?
- How can the linguistic data be standardized so that they can be adaptable to several languages?
- Should frequent expressions (like greetings, polite expressions, etc.) be lexicalized to enhance the performance of the system?
- Which linguistic data must be accounted for in a lexicon?

etc.

Most of the above mentioned issues are not really new but as there are no general concepts for handling them they are still relevant both linguistically and practically.

Acknowledgements

The research presented in this paper has been carried out and tested in the framework of the EMBASSI project (Grant No.: 01IL9904F8) providing multi-modal assistance for controlling audio and video equipment.

Notes

1 EuroWordNet Project (2001). Building a Multilingual Database with Wordnets for Several European Languages. University of Amsterdam, Department of Computational Linguistics, *http://www.illc.uva.nl/EuroWordNet/* [accessed April 2004].

2 It aims to provide easy access for everybody to complex technical systems (A/V home theatre, car devices, and public terminals), encouraging multimodal as well as multilingual user input.

3 A general characteristic of DL-Systems is that the knowledge base is made up of two components: the intensional one, called TBox, and the extensional one, called ABox. TBox is a general schema characterizing the classes of individuals to be represented, their general properties and mutual relationships, while ABox is a partial instantiation of this schema, containing assertions relating either individuals to classes, or individuals to each other. So given a concept language L, an ABox-statement in L has one of the forms (Donini et al. 1996): C(a) Concept Membership Assertion R(a, b) Role Membership Assertion where C is an L-concept, R is an L-role, and a, b are individuals.

4 AvEvent refers to the concept for TV programs in the EMBASSI application.

5 GermaNet Project (2003). GermaNet Homepage. University of Tübingen, Linguistics Department, Computational Linguistics Division, *http: //www.sfs.uni-tuebingen.de/lsd/* [accessed April 2004].

6 An example would be a determiner phrase DP that is built from a NP which in turn is built from the lexical category N.

7 Sometimes we needed a thematic role that was not existent in EWN, consequently we had to define some thematic roles that are required by the application domain in order to facilitate the semantic construction.

8 Means: *Is Tatort coming on ZDF?* , where *Tatort* is the name of a TV program, and *ZDF* the name of a TV channel.

9 The fact that this utterance is a Yes/No-question is irrelevant to phase 1, but word order information is stored and made available when the pragmatics of the utterance is computed.

10 The term case frame here is used in the same way described by Fillmore 1969 and refers to thematic roles of an expression.

11 The term valency here is used in a broad sense: it doesn't only imply the obligatory elements needed in order to make a phrase syntactically complete; but it also refers to the possible semantic and pragmatic modifications an element may take and their syntactic representations, e.g. attributes for nouns or adverbials for verbs.

12 FrameNet Project (2004). "FrameNet II Homepage." International Computer Science Institute, Berkeley, CA, *http: //www.icsi.berkeley.edu/framenet/* [accessed April 2004].

13 The pragmatic extension also presents a DRS for the corresponding pragmatic intension.

14 "standard" here implies derivations that are built by using productive rules like (e.g. verb + -er →noun adjective + -ly →adverb. etc.).

References

Abney, S. (1991). "Parsing By Chunks." In: Berwick, R. et al. (eds.). Principle-based Parsing. Dordrecht: Kluwer Academic Publishers.

Baker, C.F. et al. (1998). "The Berkeley FrameNet Project." In: Proceedings of the COLING-ACL, Montreal.

Bücher, K. et al. (2002a). "Anything to Clarify? Report Your Parsing Ambiguities!" In: Proceedings of the 15th European Conference on Artificial Intelligence (ECAI-2002), Lyon, July 2002, 465-469.

Bücher, K. et al. (2002b). "Corega Tabs: Incremental Semantic Composition." In: Proceedings of the Workshop on Applications of Description Logics (ADL 2002), Aachen, Germany, September 2002 [= CEUR Workshop Proceedings Vol. 63], *http://CEUR-WS.org/Vol-63/*.

Donini, F. M. et al. (1996). "Reasoning in Description Logics." In: Brewka, G. (ed.) (1996). Principles of Knowledge Representation. Stanford / CA: Center for the Study of Language and Information [= CSLI Publications - Studies in Logic Language and Information], 193-238.

Fillmore, Ch. (1969). Universals in Linguistic Theory. New York: Holt, Rinehart, & Winston.

Fischer, I. et al. (1996). "Incremental Semantics Construction and Anaphora Resolution Using Lambda-DRT." In: Botley, S.; Glass, J. (eds.) (1996). Proceedings of Discourse Anaphora and Anaphor Resolution Colloquium (DAARC-96), Lancaster, July 1996, 235-244.

Kamp, H.; Reyle, U. (1993). From Discourse to Logic. Dordrecht: Kluwer Academic Publishers.

Kuschert, S. (1996). Higher Order Dynamics: Relating Operational and Denotational Semantics for λ-DRT. University of Saarbrücken, CLAUS-Report 84.

Ludwig, B. (2002). "Corega Tabs: Mapping Semantics onto Pragmatics." In: Proceedings of the Workshop on Applications of Description Logics (ADL 2002), Aachen, Germany, September 2002 [= CEUR Workshop Proceedings Vol. 63], *http://CEUR-WS.org/Vol-63/* [accessed April 2004].

Niles, I.; Pease, A. (2001). "Toward a Standard Upper Ontology." In: Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001), 2001.

Vossen, P. (ed.) (1999). EuroWordNet General Document, Version 3, 1999. (EuroWordNet LE2-4003, LE4-8328, Part A, Final Document). *http://www.illc.uva.nl/EuroWordNet/docs/GeneralDocPS.zip, http://www.illc.uva.nl/EuroWordNet/docs/GeneralDocDOC.zip* [accessed April 2004].