The "User Vocabulary Definition and Meaning Mapping Module": a lexical knowledge base for Communication Aids

Constantine Stephanidis and Margherita Antona

ABSTRACT

This report describes the UVDMM ("User Vocabulary Definition and Meaning Mapping Module") a tool for lexicon selection and lexical translation in Assisted and Augmentative Communication (AAC), designed and implemented in the framework of the TIDE - ACCESS TP1001. The UVDMM Module consists of a multilingual multifunctional lexical knowledge base containing lexical linguistic knowledge related to orthographic and non-orthographic user languages. The Module’s functionalities make linguistic and translational knowledge available to the different phases of use of a communication aid (language configuration and communication), providing an innovative solution to lexicon related problems such as the selection of a suitable lexical set for each target user of a communication aid and the translation between different user languages. The design of the Module is based on state-of-the-art techniques in the fields of Computational Lexicology and Machine Translation, it adopts a hierarchical approach to lexical information representation and a lexicalist domain knowledge based approach to translation. It has been implemented in Prolog using the Attribute Logic Engine (ALE) formalism and contains a fully encoded 200 words lexicon for English, Finnish, Bliss and Pictograms.
CONTENTS

1. Introduction 3

2. An approach to lexical knowledge representation and to translation 6
   2.1. A Hierarchical approach to the representation of lexical knowledge 6
   2.2. A lexicalist domain knowledge based approach to translation 9

3. UVDMM Module design 12
   3.1. Module’s Architecture 12
   3.2. The representation of linguistic and domain knowledge 14
   3.3. The domain model 16
   3.4. Module functionalities 18

5. Summary 23

References 25

Glossary 27
1. Introduction

The adequacy of interpersonal communication aids to the requirements of users depends, among other factors, on the possibility of providing each user with an appropriate language and vocabulary set according to her/his communicative and expressive needs, while at the same time ensuring communication with as many communication partners as possible. A variety of linguistic means is available in order to meet the user communication needs and careful consideration of both user expressive and communicative needs and language characteristics are necessary in creating/selecting user vocabulary. A fundamental choice concerns the orthographic vs. symbolic nature of the language. Furthermore, different symbolic languages exist out of which the most appropriate to individual users has to be chosen according to several criteria, e.g. intelligibility, ease of acquisition, flexibility, etc. (see Fuller, Lloyd and Schlosser 1992). New symbolic languages can also be created from scratch by editing symbols or pictures. Finally, the appropriateness of a language to the requirements of a user also depends on the adequacy of its vocabulary, i.e. on the availability and accessibility of the vocabulary items which the user needs for complete and efficient communication in her/his environment. Therefore the lexicon of each user language has to be carefully selected taking into account the user’s basic needs and her/his communicative and linguistic capabilities (see e.g. Fried-Oken and More L. 1992, Stuart et al 1993, Marvin et al. 1994).

As a consequence of such a variety of user needs and possible options defining individual languages, an almost infinite range of potential languages can be created/adapted. The combination of a linguistic form (either orthographic or symbolic) with a lexicon set will be referred to in this report as user language. A user language can thus be constituted by a subset of a universal language such as English or Bliss, or the whole Pictograms symbol set, or a small collection of pictures representing objects familiar to the user, etc. Providing a user of a communication aid with a specific user language is a complex task given the many choices to be performed and factors to be taken into account. Furthermore, communication between the user of a specific language and people who are not trained to use the same language may become very complicated, unless some form of translation is provided.

In current communication aid systems, both problems of user language definition and translation between user languages are only partially faced. Current systems usually provide the user facilitator, i.e. the person facilitating the system’s utilisation by the end user, with symbol editors for creating and editing symbols of user languages, which may be associated to natural language words. In such a way, the facilitator defines the user language and associates symbols to words of a natural language. During the communication phase, the user selects symbols, and therefore automatically selects words and can exchange messages with natural language speakers who are not familiar with symbolic languages. Some fundamental problems are however left unsolved:
The facilitator is not provided with any support by the system in the process of deciding which concepts a user language has to cover, and must proceed by collecting lists of words manually or semi-manually.

Lexical translation is enabled only from user languages to one natural language, i.e. the user does not have the possibility of communicating with another communication aid user, familiar with a different user language, or with the speaker of a natural language different from the one pre-selected by the facilitator.

Some of these problems are partially addressed by various tools appearing in the market, which provide lists of words commonly occurring in user languages, out of which the facilitator can select the vocabulary suitable for each user, or tables of correspondence between some symbolic languages and a set of natural languages. These solutions, however, do not address the entire range of existing problems, and furthermore do not propose a global strategy for dealing with vocabulary and lexical translation in interpersonal communication aids. The UVDMM Module aims at providing a new global approach to the treatment of vocabulary and lexical translation in communication aids, based on the adoption of state-of-art techniques of Computational Lexicography and Lexicology and Machine Translation, and in particular:

(i) the identification and explicit representation of multilingual lexical knowledge

(ii) the representation of lexical translational relations between user languages.

In the UVDMM Module, lexical knowledge is captured by means of an appropriate representation language and made available to the user’s facilitator in the task of defining the user vocabulary. Furthermore, translational relations between lexical items of all user languages are encoded. The Module therefore guarantees the following improvements with respect to current communication aids:

- the user facilitator can use already existing multilingual vocabularies for selecting concepts to be included in user languages and define user languages

- multilingual lexical translation is ensured without the need of specifying correspondences between symbols and words or between symbols of different user languages.

The knowledge base has been designed in order to satisfy the following requirements:

- Multifunctionality: the lexical knowledge base must work as a comprehensive source of lexical information for all different communication aid functionalities which can require such a knowledge, as for example user language definition, layout creation, translation of messages, etc.

- Multilinguality: the module must represent linguistic information related to the types of languages suitable for being used as user languages in communication aids (as for
example Bliss, or an appropriate subset of English, or a symbolic language created for the needs of a specific user and made up of pictures).

- Extendibility: the module must be designed in such a way to allow easy extension as for the lexicon size for each language and the number of languages included. Future extensions like message parsing, full translation and generation, rate-enhancement, etc, should also be foreseen.

As a unified source of lexical knowledge, the UVDMM Module can be interfaced to both the user facilitator (so that lexical material is available for consulting and selection), and to system functions (so that lexical translation can be performed and, in future extensions, full syntactic analysis and interpretation of user messages becomes possible). As a consequence, interpersonal communication aid quality is improved by enabling adaptability to user requirements concerning user language lexical resources. Furthermore, the knowledge base could be enriched with further information to be exploited by techniques like word prediction, word expansion, generation of messages, etc, which are often adopted in communication aids in order to minimise the user's effort in using the system.

This report briefly introduces some relevant concepts concerning the representation of lexical knowledge and lexical translation, and outlines the approach adopted in this respect in the design of the UVDMM Module. It also provides descriptions of the internal module structure, the type of linguistic information represented in the lexical knowledge base and the module's functionalities.
2. **A proposed approach to lexical knowledge representation and to machine translation in the context of interpersonal communication aids**

The approach adopted for the design of the UVDMM Module combines state-of-the-art methodologies and techniques in the fields of Natural Language Processing and of Machine Translation, with the aim of providing a general framework for dealing with linguistic information in interpersonal communication aids. This chapter of the report contains a short overview of the theoretical concepts underlying the proposed approach. Section 2.1 briefly introduces some aspects of recent approaches to the representation and structuring of lexical knowledge, elaborated in the field of Computational Lexicography and Lexicology, and discusses their relevance for the UVDMM design. Section 2.2 briefly introduces some Machine Translation techniques and explains the reasons for their adoption in the approach proposed for the UVDMM Module.

2.1. **A Hierarchical approach to the representation of lexical knowledge**

A suitable approach to lexical information representation in the context of the UVDMM design should ensure the following:

- a consistent and economic encoding of information for all user languages
- capturing and exploiting relations of various kinds between lexical items of the same language or different languages (e.g. class membership, inclusion of classes, etc)
- retrieval of lexical items according to several criteria
- possibility of describing in a homogeneous and coherent way both language-dependent (monolingual) and language-independent (semantic) information
- possibility of encoding, at a lexical level, information of different types (e.g. orthographic, semantic, etc) in a homogeneous way and of allowing the progressive expansion of the lexical knowledge base in terms of both the lexicon covered and the type of information included (e.g. addition of syntactic lexical information)

Recent approaches to the description and investigation of the lexicon of natural languages emphasise the hierarchical aspects of lexicon structure. From this point of view, the lexicon can be described as a set of interrelated hierarchies of classes and sub-classes of lexical objects, defined by the morphological, syntactic, and semantic
characteristics of words. For example, in an appropriate hierarchy fragment, the word bread could be classified both syntactically and semantically. From a syntactic point of view, bread can be classified as belonging to the class of mass-nouns, which belongs to the class of common nouns, which belongs to the general class of nouns. Semantically, the concept of “bread” as a referent of the noun bread can be defined as belonging to the class “food”, which belongs to the class “physical object”, which in turn belongs to the general class “object”. The hierarchical structure of the lexicon can be enriched by assigning each class a set of particular characteristics which distinguishes it from other classes and is inherited by sub-classes.

Such a view of the lexicon provides the means for capturing and “expliciting” relations between lexical items at several levels: for example, from the morphological and syntactic point of view, hierarchies define classes of elements showing homogenous behaviour (e.g., the class of all nouns whose plural form is obtained by adding an ‘s’, or the class of all verbs which require a subject and an object complement). Semantic relations between lexical items also constitute hierarchies: for example, an individual corresponding to the definition of dog also corresponds to the definition of mammal, and an individual corresponding to the definition of mammal also corresponds to the definition of animal, etc (hyponymy and hypernymy relations). Semantic relations established in hierarchies are usually derived from dictionary definitions of the type ‘a dog is a mammal, carnivore, etc’.

In recent approaches to the design of lexical knowledge bases, high-level declarative representation languages based on “types” and “feature-structures” are adopted (typed feature-structure representation languages) for the representation of hierarchical lexica. Types can be informally defined as labels for classes in a hierarchy and are explicitly declared to give rise to a legal hierarchy (for a formal definition, see e.g. Carpenter 1992).

Feature-structures (descriptions) are attribute-value pairs representing type characteristics and are inherited through the hierarchy. Typed feature-structure representation languages are based on Prolog and fully exploit Prolog term unification (instantiation of Prolog variables). Operations on both types and feature structures are performed by unification: for example, two feature structures not containing incompatible information (different values for the same attribute) can be unified to produce a new feature structure which contains the information of both. Rules can be defined which perform operations on types and descriptions. Formalisms of this type are declarative, i.e. they represent knowledge in the form of facts, which can be combined to produce new knowledge by rule inference. They are therefore particularly suited for a structured approach to the representation of the lexicon, since they allow capturing morphological, syntactic and semantic regularities and therefore provide an effective and economic way of encoding lexical information (see e.g. Calzolari et al., 1992).

Recently, typed feature structure representation languages have been used in the context of the representation of multilingual information and translational equivalence (see e.g. Copestake et al. 1992, Sanfilippo et al. 1992).
The ALE system (Carpenter & Penn 1994), which has been used in the implementation of the UVDMM Module, provides a complete package including type hierarchy, feature structure description and constraints, definite clause logic using typed feature structures as terms, and a chart parser. The ALE formalism allows representing lexical, grammatical and semantic information to be exploited for structured word retrieval and lexical translation, but also, in possible future extensions of the module, for further processing of lexical information.
2.2. A lexicalist domain knowledge based approach to translation

Lexical translation constitutes one of the major problems in current approaches to Machine Translation, since translational relations at a lexical level are not straightforward and different techniques are appropriate according to the translation task. Copestake 1995 provides a brief introduction to state-of-the-art approaches and techniques and discusses some advantages and disadvantages of each.

Mapping between lexical entries of different languages can take place directly by means of specific rules (transfer) or indirectly through an abstract language-independent representation of meaning (interlingua). Interlingua can be based on the semantic representation of lexical meaning or on some model of the knowledge concerning the topic of the text to be translated (domain knowledge). While the transfer approach is usually claimed to avoid the complexity of fine-grained meaning representation and ensure qualitatively good translation output, the interlingua approach is preferred in multilingual systems, where a proliferation of transfer modules (and rules) would occur (one per each language pair translation direction), and is considered feasible for translation in restricted domains, where the representation of semantic and domain knowledge can be reasonably limited.

Furthermore, lexical translation cannot occur in isolation. Context has to be taken into account in order to solve translational problems such as lexical ambiguity (a word has more than one sense, i.e. corresponds to more than one concept, e.g. the word “bank” in English, synonymy (more than one words correspond to a concept), lexical gaps (no words corresponding to a concept, or to a word of another language) and translational ambiguity (one word in the source language can translate into more than one word in the target language). Current translation systems perform translation on some abstract representation of text at a sentence or discourse level. Recently, however, lexicalist linguistic theory, which assume the lexicon as a fundamental repository of both syntactic and semantic knowledge, have given rise to lexicalist (transfer) approaches to translation stating translational relations between feature structure representation of lexical items of different languages.

A suitable approach to lexical translation in the context of the UVDMM Module is strictly related to the characteristics of the specific application and of the types of languages involved (orthographic and symbolic), and should satisfy the following requirements:

- ensuring the coverage of any type of user language independently from its form (symbolic, orthographic, mixed, etc);

- ensuring translation between an undefined number of languages (in principle, there may be as many user languages as users);
- ensuring the possibility of translation between user languages still to be defined (both in terms of lexicon selection and symbol creation) without necessarily involving a revision of translation rules;

- ensuring the possibility of extending and modifying user-languages without affecting the translation process.

Therefore, an approach to translation has to be adopted which guarantees maximum flexibility as for the type and number of the languages to be translated and ensures system extendibility. Clearly, a transfer approach is not appropriate in such a case since it requires writing as many transfer modules as are the translational directions for language pair (e.g. a system translating between 4 languages contains 12 transfer modules) and extensive up-dating is necessary when some change occurs in the system lexical coverage. The adoption of language independent meaning representation for lexical meaning mapping is better suited to the requirements posed by the specific type of applications and by the nature of involved languages. For example, in case of a user language defined from scratch by a facilitator, it would be impossible to perform a direct mapping between such a language and all other user languages in the system, while it is much simpler to perform a single mapping between such a language and an abstract semantic representation directly in the phase of language definition. The system is meant to provide translation for orthographic languages as well, and the question may arise of whether a transfer approach would be more efficient in the case of orthographic languages. On one hand, however, orthographic languages have to be considered in this context as specific and well-defined subsets of natural languages functioning as user-language. On the other hand, system multilinguality requires a homogeneous approach to translation adequate for all user languages independently from their orthographic or non-orthographic nature.

Given the restricted domain covered by communication aid user language, a knowledge-based interlingua constitutes a suitable choice for translation in the UVDMM Module, since it requires a limited domain model containing some representation of domain entities to be progressively enriched with common-sense knowledge useful for translation (details of the UVDMM Module domain model and the types of information it contains are described later on).

As for the linguistic level at which translation is to be performed, the nature of the UVDMM Module and types of languages it includes require that translation is as lexicalist as possible, i.e. as independent as possible from non-lexical information (sentence structure, etc). One reason leading to the choice of a lexicalist approach is that lexical information is reasonably expected to play a central role in the translation between communication aids user languages. In fact, for some symbolic languages, whose morphological and syntactic characteristics are much simpler than those of natural languages, lexical information may even be the only type of information available for the translation process, and lexical translation adequate for the production of a correct output in the target language. On the other hand, typical phenomena of lexical semantics such as ambiguity and synonymy are expected to be less common in user languages than in non-restricted natural language, due to both the lexical
limitation of symbolic languages and the limited discourse domains covered. A second reason is that translation in communication aids, unlike in most other machine translation applications, does not necessarily mean high-quality translation. Despite good translation quality (at least correctness of translation) is per se desirable, in AAC correctness of translation may be considered as a secondary goal with respect to ensuring some form of communication by means of an understandable, even if not completely correct, translation output. Lexical word-by-word translation can be considered as a good starting point for establishing a minimal level of communication between users of different languages. Therefore, a theoretical approach to lexical translation is required which ensures the immediate possibility of word-by-word translation, and on the other hand constitutes the basis for future developments aimed at better translation quality. A lexicalist approach is suitable for the given purposes since it performs translation by unification of source and target language feature structures, and allows therefore the possibility to create a word-by-word translation backbone to be further elaborated in a progressive way by both enriching the source language representation with syntactic and semantic information and encoding target language linguistic knowledge to produce a correct output.

Summing up, the particular characteristics of the UVDM Module and of user languages in communication aids lead to the adoption of an approach to translation, which is on the one hand language-pair independent and based on domain knowledge and on the other hand basically lexicalist. These requirements can be fulfilled by exploiting the unification mechanism of typed feature structure formalisms. Such formalisms can in fact be used for representing both lexical entities and their ontological domain, assigning a domain referent to each entry as a feature value. As a consequence, translational equivalence can be stated as identity on the value of the appropriate feature between lexical entries of different languages, giving rise to an innovative lexicalist domain knowledge based approach to translation. The direct representation of a domain model by means of a typed feature-structure based formalism represents in fact a particularly innovative aspect of the module design, allowing the combination of lexicalist translation to the representation of a domain model to be used as an interlingua.
3. UVDMM Module Design

3.1. Module's Architecture

The adoption of a hierarchical approach to the representation of lexical knowledge and of a typed feature structure formalism, allows capturing different types of knowledge in a common format and manipulating information by means of very simple mechanisms such as unification and information inheritance. This enables the adoption of a very simple internal architecture of the module, which is constituted by the following components (see Figure 1.):

(i) Interface to other communication aid components

The UVDMM Module is not provided with an independent user interface since it is designed in order to interact with other Communication Aid modules (such as the module for language definition, including a symbol editor, and the communication module). The user (either the facilitator or the final user) does not directly access the UVDMM Module, but rather has access to other independent modules which extract or encode linguistic information from the UVDMM Module. Also, the UVDMM Module does not contain graphic information concerning symbolic languages, so symbols in both the language definition and the communication phases have to be retrieved from an external symbol database and related to the corresponding lexical information (encoded in the UVDMM) by means of a unique symbol identifier. At run time, the UVDMM Module receives requests for information retrieval or modification. Each request is transformed into an appropriate Prolog query, is executed and the results are returned to the requesting module. Retrieval queries return the Prolog query results while modification rules return a Boolean value (normally ‘true’) meaning that the requested routine has been executed successfully by Prolog.

(ii) ALE compiler

The ALE compiler version 2.0 (Carpenter&Penn 1994) is used in the UVDMM Module.
(iii) ALE source files

ALE source files include type declaration, logic rules and lexica. Section 3.2. describes the way knowledge is represented in ALE source files.

(iv) ALE compiled files

ALE compiled files contain type declaration, logic rules and lexica in compiled form for consulting by retrieval rules.
(v) Prolog rules

The module contains Prolog rules for retrieval, translation and modification of the encoded knowledge. Rules exploit built-in mechanisms of the representation language for type feature structure unification. Prolog rules correspond to the functionality's of the Module and are described in Section 3.4.

The current Module’s implementation contains a domain knowledge base of about 250 concepts (covering discourse domains common in AAC) and the related lexica for English, Finnish, Bliss and Pictograms.

3.2. The representation of linguistic and domain knowledge

The UVDMM knowledge base contains the following types of information:

- languages names
- user names
- syntactic categories
- the semantic ontology

Knowledge categories are hierarchically organised and feature structure descriptions are used where appropriate. The content of the ontological domain model is described in the following Section. Besides type and feature structure declaration, the knowledge base contains also logic rules representing characteristics of languages such as their symbolic vs. orthographic nature, subset relation to another language (for example, a user language can be constituted by the subset of Bliss used by the specific user), universal vs. user-defined character and the user to which each language is assigned.

The rule pattern of lexical entries is also defined as a type introducing the features which encode the language, the syntactic category and the ontological referent of each entry. Lexical entries for symbolic languages correspond to a symbol identifier number which refer to the symbol non textual representation in an external symbol database. Lexical entries for orthographic languages are represented by means of the word’s graphical form. Descriptions of lexical entries are defined by a macro in order to simplify lexicon writing.

Each monolingual dictionary entry contains a reference to the language, a reference to a syntactic category and a reference to a type in the semantic representation, thus inheriting all the information contained in such a type. All monolingual lexical entries referring to the same ontological domain entity are therefore translationally equivalent, and lexical translational equivalence is established by identity with respect to the types encoding the semantic information.
Figure 2. shows how user languages relate to the semantic representation by means of feature unification. All translationally equivalent entries in monolingual lexica correspond to the same entity in the ontology, as appearing from the values of the trans_sem feature. Figure 3. shows the top part of the type hierarchy and its interrelationships to logical rules and lexicon entries.

Figure 2. Meaning Mapping in the UVDMM Module
Figure 3. The top ALE hierarchy
3.3. The domain model

In Section 2.3 the reasons for choosing a domain knowledge based approach to translation in the UVDMM Module were explained. This Section describes the adopted domain model and its representation. The domain model constitutes the biggest and most important part of the whole Module's type hierarchy. It builds on various ontological knowledge classifications proposed in literature (Alshawi and Carter 1992, Dahlgren 1988, Onyshkevych and Niremburg 1992, Matsukawa and Yokota 1992, Pustejovsky and Boguraev 1994, Quantz et al. 1994). It can be defined as an ontological type hierarchy classifying domain entities into types and describing their characteristics by means of feature structures. It contains concepts related to the discourse domains to be covered by the lexicon (including common activities, people, body and body parts, home equipment, food, places, weather, time). Concepts in the ontology type hierarchy are meant to correspond to real-world entities rather than to monolingual word sense distinctions, i.e. represent the referential meaning of basic lexical units. Concepts correspond as strictly as possible to the level of word meaning (lexical meaning is not decomposed in smaller units).

The basic sub-types of the entity type are situations, objects, qualities and relations. Within each basic type, entities are grouped into sub-types according to the common-sense hierarchical structure of the domain (e.g. places include buildings, rooms). Situations are assigned argument frames according to the participants they may take, and frames are constrained to types likely to occur as arguments of specific situations (e.g. the situation eat has as constraint on the object argument the type food). Objects are assigned features representing semantic relation like part_of, group_of, etc, and also features related to common-sense knowledge about object location, typical association with other objects, etc. Qualities are classified according to the type they modify (qualities modifying people, modifying situations, etc).

The classification does not claim universal semantic validity, but rather is designed in order to capture possible semantic and common-sense relation between domain elements. Semantic domains can be derived for every entity in the hierarchy. For example, constraints on situation frames allow the retrieval of "semantic domains" for situations and situation classes by collecting all argument types and their subtypes (at the moment, however, type restrictions on arguments are not meant to be used as selectional restrictions in semantic interpretation, since no parsing takes place). The semantic domain of objects may include all sub-types, qualities which can refer to an object, the object typical location and the other objects found in such place, etc. So, for example, the semantic domain related to food includes all food types, food properties, situations related to food such as eating and cooking, etc. Many concepts in the hierarchy appear in more than one semantic domain.

The ontological model can be extended at any time by adding new concepts to the existing ones in the type hierarchy and establishing semantic relations with respect to them. In future extensions of the system, the ontological model can be exploited for
providing valuable information for sentence parsing and for building a full semantic interpretation of sentences.

### 3.4. Module functionalities

The UVDMM Module functionalities can be distinguished into the following classes:

(a) Type hierarchy information retrieval

Information retrieval is performed by means of Prolog queries on the compiled type hierarchy and logic rules. Type hierarchy information retrieval functions include:

(i) retrieval of all languages declared in the module
(ii) retrieval of all orthographic languages declared in the module
(iii) retrieval of all symbolic languages declared in the module
(iv) retrieval of all user defined languages (i.e. all subset languages and symbolic languages created by the facilitator) declared in the module
(v) retrieval of the language assigned to a given user
(vi) retrieval of subset languages of a given language
(vii) retrieval of all syntactic categories defined in the module
(viii) retrieval of all user names defined in the module
(ix) retrieval of all users of a language

Functions (i) to (iv) and function (vii) and (viii) take no input arguments. Function (v) requires a user name as an input argument, function (vi) requires a universal language name and function (ix) a language name. All functions output lists of elements in the type hierarchy (with no internal structure).

(b) Retrieval of lexical entries, lexical feature structure descriptions and portions of the ontological hierarchy

Different types of lexical knowledge can be used for accessing lexical material existing in the Module. For example, syntactic knowledge concerning parts of speech allows retrieving all nouns (or verbs, or adjectives, etc) of a user language, while language independent knowledge semantic relations allows retrieving all lexical units related to food (or to furniture, etc). Combining the two types of knowledge allows retrieving all nouns of a given language.
semantically related to a given concepts. Lexical entries retrieval is performed in the UVDMM Module by means of a unique Prolog query, which takes as a first argument a feature structure description and outputs a list of lexical entries whose feature structure representation unifies with the given description. For example:

- `lan:english` retrieves all entries of the English lexicon
- `lan:english, syn:noun` retrieves all English nouns
- `lan:english, syn:noun;verb` retrieves all English nouns and verbs
- `lan:english, trans_sem:food1` retrieves all English entries whose, ontological referent is a subtype of `food1` in the type hierarchy, etc.

The lexical entries retrieval rule extracts information from compiled lexica and exploits ALE built-in mechanisms for feature structure unification. Each instance of the query retrieves all lexical entries whose description unifies with the feature structure description provided as input argument. In such a way, a unique query captures lexical entries unifying with different descriptions. For example, in case the desired retrieval set is all English nouns related to the concept `food`, the query will look for all lexical entries unifying with the description:

```
language:english, syn:noun, trans_sem:X
```

where `X` is a variable whose possible instantiations are requested to be sub-types of the type `c food`. Feature structure description can be complex (i.e. include coordination/alternation of features), but must be consistent with the type declaration. Some predefined retrieval queries are provided, including the retrieval of all possible arguments of a situation, all sub-types or sisters of an entity in the hierarchy, all entities linked by a semantic relation to some object, all qualities modifying an entity.

Functionalities also include the retrieval of the value for a specified feature (or features) appearing in the description of an entry. For example, the returned value for the features `lan` and `syn` for the entry `bread` will be `english` and `noun` respectively. Feature values retrieval is performed by means of unification as well. Such a function requires as input arguments an feature structure description with a variable feature value and a lexical entry. E.g.:

```
language:LAN, bread
```

The result will correspond in this case to the instantiated value of the variable `LAN`. 
Another type of retrieval concerns ontological sub-hierarchies. The ontological hierarchy can be explored in several ways:

(i) sub-types of a given entity can be retrieved in a flat list
(ii) sub-types of a given entity can be retrieved in a structured list (inclusive parenthesis format), e.g.:

\[ \text{c_food}([\text{c_bread}, \text{c_fruit}([\text{c_apple}, \text{c_orange}, ...]), ...]) \]

(iii) elements of the ontological hierarchy interrelated by means of semantic relations can be retrieved, thus extracting semantic domains.

Ontology sub-hierarchies retrieval of types (i) and (ii) requires as an input argument a type in the ontological hierarchy, while retrieval of type (iii) requires as input arguments a type in the ontological hierarchy and a set of semantic relations.

(c) Translation

Translation can be performed on individual lexical entries or messages (i.e. unprocessed lists of entries). Lexical translation equivalence relations are declarative and reversible. They are established by a Prolog query, which must be called with four arguments (source entry, source language, target entry, target language) and performs inference on the basis of lexical descriptions (in particular of the features lan whose value correspond to a language and trans_sem whose value correspond to an element in the ontological hierarchy). Each argument of the query can serve as both an input argument, or an output variable, depending on the format of the query. For example:

- \( \text{translates} (\text{bread}, \text{english}, \text{TargetEntry}, \text{italian}) \) where TargetEntry is a variable, retrieves the Italian entry equivalent to bread;

- \( \text{translates} (\text{bread}, \text{english}, \text{TargetEntry}, \text{TargetLang}) \) where TargetEntry and TargetLang are variables, retrieves the translational equivalents of bread defined in the Module, along with their respective languages;

- \( \text{translates} (\text{SourceEntry}, \text{SourceLan}, \text{TargetEntry}, \text{TargetLang}) \) where all the arguments are variables, retrieves all pairs of translationally equivalent entries in the Module along with their respective languages, etc.

Messages (lists of entries) are translated as well by means of a similar query. Lexical entries’ translation returns strings containing output lexical entries when source entry, source language and target language are given as input arguments. Message translation returns lists of lexical items corresponding to the translation of the input list.
(d) Addition/removal of facilitator defined languages and user names

User languages can be created or removed from the system. Two types of user languages can be distinguished: subset languages (i.e. languages which are constituted by a subset of the lexicon of a universal language such as English or Bliss), and user-defined symbolic languages, which are created from scratch by the facilitator designing new symbols. All languages supported by the module at a given stage must be declared: when the facilitator wants to define or remove a language, she/he is presented with a list of subset or user defined language and can provide a new name for a new language or indicate the language to be removed.

User names and languages can be added/removed from the type hierarchy by means of specific rules. Adding user names requires two input arguments, the name to be added and the language to be assigned to the new user, while user name removal only requires the name to be removed. Addition/removal of facilitator-defined symbolic languages requires as sole input argument the language name. Finally, addition/removal of subset languages requires the subset and the universal language name.

The Prolog rules responsible for carrying out the above additions/removals automatically up date ALE source files, but do not perform ALE compiling and do not load the modified hierarchy; they return Boolean values.

(e) Definition/deletion of entries in lexica

One of the basic steps for the definition of a user language is the selection of the vocabulary the user will have at her/his disposal. Lexical entries can be added and removed from all user languages. In the case of subset languages, lexical entries to be added are selected directly from the vocabulary of the related universal language. In case of new symbolic languages, the facilitator must provide as an input: the identifier of the selected symbol, the syntactic category of the new lexical entry, and the lexical entry meaning, which is assigned by selecting the corresponding translational equivalent in the facilitator mother tongue or in another language of her/his choice.

Entry definition in user-defined symbolic languages is performed by selecting an equivalent entry in a source language, therefore it requires as input arguments the source language, the target language, the source entry, the target entry (symbol identifier) and the syntactic category to be assigned to the new entry in the target language. The semantics of the source entry are automatically copied to the new entry.

Entry definition in user-defined symbolic languages can also be performed by directly consulting the knowledge base. In this case, four arguments are needed: the target language, the entry to be defined (symbol identifier), the syntactic
category and the ontological concept to be assigned. For entry definition in subset languages three arguments are required, namely the subset language, the universal language and the entry to be added to the subset. Entry deletion requires two arguments, the language and the entry to be deleted. Routines for definition/deletion of lexical entries return Boolean values.
5. Summary, discussion and conclusions

The purpose of the UVDMM Module in the framework of interpersonal communication aid design is to support the process of user language definition by the facilitator through making lexical material and information accessible, and to facilitate lexical translation between different user languages. The module is multifunctional and multilingual lexical knowledge base, suitable for representing lexical knowledge of both orthographic and non-orthographic languages, as well as lexical translational equivalence relations between items of different languages.

The design of the module is based on methodologies and techniques elaborated in the fields of Computational Lexicography and Lexicology and of Machine Translation. A hierarchical approach to the representation of lexical knowledge is proposed, exploiting the characteristics of declarative formalism for lexical knowledge representation and of typed feature structure languages in particular. Lexical translation equivalence in the module is established by means of a domain model represented by means of the mentioned formalism. The domain model reflects the ontological hierarchy of the discourse domains typical of communication through interpersonal communication aids, covering the concepts underlying the related lexicon. Retrieval of lexical items and sub-hierarchies in the knowledge base is performed by means of appropriate rules which exploit the built-in mechanisms of the formalism. Lexical translational relations are inferred by means of a unique rule.

The adopted approach to the treatment of vocabulary related problems in the framework of communication aid design facilitates the creation of a complete and coherent source of lexical knowledge into the system modular architecture, to be exploited whenever information related to user language(s) is required, e.g. in the user language definition phase or in the communication phase. The introduction of multilingual and hierarchically structured lexical knowledge in the design of interpersonal communication aids constitutes an innovative step with respect to current practice, which enables the adaptability of the system to user requirements related to vocabulary and communication between users of different languages.

The functionalities offered by the UVDMM Module can be appropriately exploited in Interpersonal Communication Aids in order to assist language configuration. The facilitator is provided with adequate means to extract and use knowledge on already existing lexical resources according to syntactic and semantic criteria. As a consequence, language configuration in Communication Aids becomes faster and more consistent with respect to current practice, ensuring at the same time that user requirements in terms lexical coverage are met and suitable expression means are provide to each user in her/his communication environment.

The domain model included in the Module also allows establishing lexical translational equivalence relations between all languages without the need of explicitly specifying them for all possible language pairs, and therefore makes practically possible lexical translation between all language pairs in the Module.
As a consequence, the exploitation of the UVDMM Module in Interpersonal Communication Aids offers the concrete possibility of improving communication aid adaptability to user communicative and linguistic requirements, along with the adaptability of the communicator’s functionalities ensured by the modular design approach, thus overcoming many of the current problems related to languages and vocabularies in Interpersonal Communication Aids.

The Module design, based on typed feature structure encoding of linguistic knowledge, also ensures the possibility of expanding the Module’s content and functionalities to include morphological and syntactic information, message parsing and full translation, rate enhancement techniques such as linguistic based word prediction and telegraphic style message expansion.
References


Glossary

**Discourse domain**: discourse topic and related ontological model

**Entity**: element of an ontological model representing a real-world object or relation.

**Feature structure**: Recursive attribute-value pair used in high-level formalisms for the representation of linguistic information.

**Homonymy**: type of lexical ambiguity occurring when two words are superficially identical but belong to different part of speech, e.g. "walk" as a verb and as a noun in English.

**Interlingua**: abstract language independent representation of meaning used as intermediate representation language in Machine Translation.

**Interlingua based translation**: approach to Machine Translation adopting an interlingua as an intermediate representation for mapping meaning between different languages.

**Lexical ambiguity**: phenomenon occurring when a word has more than one sense, i.e. corresponds to more than one concept, e.g. the word ‘bank’ in English.

**Lexical gap**: phenomenon occurring when a language lacks a specific word for expressing a concept.

**Lexical hierarchy**: hierarchy of lexical elements organised according to syntactic or semantic classes.

**Lexical information retrieval**: retrieval of information concerning the lexicon.

**Lexical knowledge**: knowledge concerning the lexicon.

**Lexical knowledge base**: knowledge base representing lexical knowledge.

**Lexical representation language**: formalism for the representation of lexical information.

**Lexical Translation**: translation between lexical elements of different languages.

**Ontological Domain Model**: model of meaning representing real-world objects and relations.

**Part of speech**: syntactic category to which a word belongs (noun, verb, adjective, adverb, etc)
**Polysemy**: type of lexical ambiguity occurring when a word can be interpreted in different ways according to the context, e.g. "adopt a child" and "adopt an approach".

**Synonymy**: phenomenon occurring when two different lexical elements have the same meaning.

**Transfer based translation**: approach to translation based on the direct mapping between lexical elements of different languages.

**Translation ambiguity**: phenomenon occurring when a lexical item of a language has more than one translational equivalent in another language.

**Translation rule**: rule establishing translation equivalence relations between items of different languages.

**Translational equivalence**: relation occurring between two expressions belonging to different languages when one constitutes the translation of the other.

**Type**: class of homogeneous elements.

**Type declaration**: declaration of types in typed formalisms.

**Type hierarchy**: hierarchy of types in typed formalisms.

**Typed Feature Structure Representation Language**: high-level formalism based on the hierarchical typed organisation of feature structures.

**Unification**: operation which combines two compatible feature structures producing a third one.