Lexical Options in Multilingual Generation from a Knowledge Base

Manfred Stede

[1] Dept. of Computer Science, University of Toronto, Canada
[2] Research Center for Applied Knowledge Processing (FAW)
Helmholtzstr. 16, D-89081 Ulm, Germany

Abstract. From the viewpoint of multilingual generation, the common underlying knowledge base should be kept clear of language-specific concepts. This goal presupposes that lexical items of various languages cannot map one-to-one onto concepts all the time. We propose a more flexible way of attaching lexical items to *configurations* of concepts and roles, and a *lexical option finder* that determines the set of content words that *cover* pieces of the message to be expressed, thereby performing the first half of the "chunking" task (dividing the message into separately verbalizable parts). This pool of lexical options will also include synonyms and near-synonyms: items with identical *denotation*, that is semantic representation in the KB, but different *connotational* characteristics. From this set, the subsequent steps of the generation process can select the most preferred subset for expressing the message.

1 Introduction

When language is to be generated from an underlying knowledge base, lexical items need to be linked somehow to the representational units in the KB. In the following, we assume a KL-ONE style representation (LOOM [MacGregor, Bates 1987], in this case) and draw in particular on two distinctions made in this family of languages. First, on the division between *concepts* and *relations* (or *roles*) holding among concepts; second, on the division between *terminological* knowledge, which are concept and relation definitions, and *assertional* knowledge, which are instances of concepts and relations, representing entities of the world.

Previous language generators have typically employed a one-to-one mapping from KB units to lexical items, with their producers occasionally acknowledging this as a simplification (e.g., [Novak 1991, p. 666]). The direct association between concepts and words (or phrases) evades the problem of genuine *lexical choice*, which has prompted several researchers to point out the lack of work on this subject (e.g., [Marcus 1987], [Nirenburg, Nirenburg 1988], [McDonald 1991]).[3] Choice means selecting from similar words and thus requires these to be adequately represented as synonyms or near-synonyms. While in principle one

[3] For a general overview of research on lexicalization in language generation, see [Stede 1995].

could do so by creating an individual concept for each of them in the KB, this "solution" would merely shift the responsibility for lexical choice to the application program that produces the representation of the content to be expressed—after all, it has to instantiate the 'right' set of concepts and relations that are given to the generator. But many lexical choices are clearly linguistic matters, and no application ought to bother with them (e.g., the decision to use either *die* or *kick the bucket*), although it has to supply the general *parameters* like pragmatic factors, a user model, etc., that will direct the generator in decision-making.

Moreover, we are interested in *multilingual* generation, and a one-to-one correspondence between concepts and lexical items would require the presence of language-specific concepts in the KB, which is not desirable from the knowledge engineer's perspective. Even when the target languages are closely related, we have to deal with incongruities like this one from a bilingual automobile manual:

Disconnect the spark plug wire and ...
Das Zündkabel *abziehen* und ...

The closest translation of *disconnect* is *trennen*, and that of *abziehen* is *pull off*. Hence, the German sentence describes the nature of the physical movement and some property of the connection between the two parts, whereas the English version focuses on the effect that the action will have on the connection of the parts. Either way of using literal translations would be awkward in the given context; but again, we want to be able to generate these sentences from the same representation, not using a concept DISCONNECT for English and a separate concept ABZIEHEN for German. Later on we will examine this example more closely.

In this paper, we attempt to make progress on lexical choice in the following way. First, we argue in favor of separating *denotation* from *connotation*.[4] We treat the former as the conceptual part of word meaning, including selectional restictions, that is represented in the knowledge base and in the KB–lexicon interface, and the latter as features pertaining to emphasis, style, and the like, that do not change the truth-conditions of the utterance, and yet contribute to what Cruse [1986] calls *evoked* meaning, or what systemic-functional grammar labels as *interpersonal* meaning. While the denotation amounts to the *necessary conditions* for using a word, the connotational features establish a *preference ranking*: a stylistic goal of, say, FORMALITY cannot always be achieved fully (if the lexicon does not provide any formal words for the particular proposition at hand); instead it is a matter of *maximizing* the fulfillment of the various stylistic goals.

We represent the denotation of similar words by mapping them onto the same semantic predicate, but with different thematic roles, selectional restrictions, or distinguishing semantic traits. Therefore, we associate lexical items not to concepts only, but to entire configurations of a concept and various roles and

[4] This distinction has been made in semantic theory since medieval times, but in a variety of ways. For a comprehensive historical overview, see [Garza-Cuarón 1991].

fillers. Furthermore, to achieve multilinguality, we apply the notion of near-synonymy across languages: pairs of equivalent or almost-equivalent words in different languages are seen as synonyms or near-synonyms, respectively.

Having represented word denotations, we define a *lexical option finder* (similar in spirit to the proposal by Miezitis [1988]) that traverses the proposition to be expressed and determines all lexical items that can denote some parts of the proposition. These items may vary in semantic specificity and in connotation; later stages of the generation process will have to select from this pool a subset of items that is most appropriate to express the given message, which is measured by the preference function. In addition, the selection will have to respect *collocational* constraints: lexical choices for different parts of a proposition can be dependent on one another, as certain words commonly occur together (e.g., *heavy smoker*), whereas others cannot (e.g., ??*heavy reader*). However, aspects of the selection procedure are not discussed further here.

2 Knowledge representation and lexical items

Consider the group of lexical items *to die, to perish, to pass away, to kick the bucket*, which all denote the same event, DIE, but which have different connotations. From the perspective of knowledge representation, we want to avoid having four DIE-concepts in the KB merely to gain the ability of generating the four different items (not to mention the addition of similar items in other languages); the distinctions made in the KB should be geared towards the underlying *reasoning* tasks and not towards possible verbalizations.[5] Hence, we assume a single concept DIE, and in turn have to represent the differences between the lexical items in another way. One difference is that the items do not equally apply to the same class of entities: anything that lives, including plants, can *die*, but *pass away*, according to Cruse [1986], applies only to human beings, whereas both humans and animals can be said to *kick the bucket* or *perish*. These different restrictions are to be represented in the link between KB and lexicon. On the other hand, the items clearly differ in terms of their formality, which we represent using features associated to lexical items, outside of the KB.

2.1 The KB–lexicon link

For the reasons given in section 1, we want to move away from a one-to-one mapping between concepts and lexical items, and instead enable the generator to actively *choose* its words. When generating from a representation based on a taxonomic KB, this step has the following ramifications:

[5] In other words, we do not want the grain-size of the conceptual representation to be determined by the grain-size of the lexicons of the languages we want to generate — Novak [1993] also discusses the desirability of separating the conceptual ontology from linguistic (including lexical) knowledge. This contrasts with approaches like that of Emele *et al.* [1992], who deliberately introduce a new concept wherever there is a word in one of the target languages to be generated.

- Choice between synonyms and near-synonyms
  The same concept may be mapped to more than one word, differing only in terms of stylistic features (see [Stede 1993]) or other pragmatic attributes, as in the PAULINE system [Hovy 1988].
- Choice between more or less specific words
  By considering the most specific as well as more general lexical items, we enable the system to favor a general over a specific word for connotational reasons or due to genre-specific preferences or whatever (Reiter [1990] has done so to prefer basic-level-category terms). This contrasts with the approach of looking for the most-specific item only, as proposed, for example, by Levelt [1989] in his discussion of the 'hypernym problem', and as it has become a standard heuristic in NLG. Hypernyms are an important source to draw from in lexical choice. For example, if we are talking about a poodle and want to mark the utterance as pejorative, we may use the word *mutt* (not necessarily meant in the literal sense) or the German *Köter*, which presupposes accessing the lexical items associated not only with the POODLE concept, but also with DOG.
- Unnamed concepts
  As soon as a one-to-one mapping is not required, there may be concepts that do not have a lexeme attached to them. For verbalization, a phrase has to be constructed out of a more general word and a restrictive modifier; this situation was dealt with by Sondheimer, Cumming, Albano [1990].
- Words covering entire concept configurations
  Words may correspond not to a single concept, but to attached roles and their fillers as well. Proposals to this end have been made by Horacek [1990] and Nogier, Zock [1992]. The following subsection explains a new approach to the problem, based on the distinction between *concepts* and *instances* in KL-ONE languages.

2.2 Mapping lexical items to concept configurations

Recall the example given at the beginning of this section. To establish the link between the concept DIE, the role EXPERIENCER, and the appropriate filler (ANIMATE-BEING, ANIMAL, HUMAN) on the one hand, and the lexical item on the other, we create an *instance* of the concept, whose properties exactly reflect the conditions necessary for using the lexical item. These instances serve as interface between conceptual knowledge and lexicon, and they ought to be kept separate from other, "standard" instances in the KB: they have roles pointing to the actual lexical entries for the languages used (here, English and German), where the connotational features as well as the syntactic properties are stored. Figure 1 shows instances in the LOOM language corresponding to the four items discussed above. We assume that in the knowledge base HUMAN is a subclass of ANIMAL, which in turn is a subclass of ANIMATE-BEING. Since *perish* and *kick the bucket* have identical restrictions, the interface instance is the same; differences in formality are to be noted in the lexical entries.

```
(tell (:about
  die_i DIE
  (experiencer animate_being_d)
  (e-lexeme die_l)
  (g-lexeme sterben_l)))

(tell (:about
  pass_away_i DIE
  (experiencer human_d)
  (e-lexeme pass_away_l)
  (g-lexeme entschlafen_l)))
```

```
(tell (:about
  perish-and-ktb_i DIE
  (experiencer animal_d)
  (:filled-by e-lexeme
     perish_l kick_bucket_l)
  (g-lexeme abkratzen)))
```

Fig. 1. Instances linking (simple) concept configurations to lexical items

When verbalizing a represented proposition, we look up those lexical items that can be used to express some part(s) of that proposition, depending on the desired language. Given the definitions in figure 1, if the event is the death of a dog, for example, the English options are *die, perish, kick the bucket*, whereas in German there is only the choice between *sterben* and *abkratzen*, where the latter is marked as informal.

If a lexical item always corresponds to exactly one concept, this lookup needs simply to be done for every instance participating in the proposition. However, since we intend to map lexical items to entire *configurations* of concepts and roles, a complication arises: we need a new criterion that tells us when we have assembled a set of lexical items that together express the complete proposition. As a prerequisite, we have to add to the lexical items the information as to what concepts and relations they actually express, or what part of a proposition they can *cover*. An additional, multi-valued role named COVERING performs this function by pointing to the set of covered concepts and relations, which is always a subset of those appearing in the instance definition, including inherited roles.

With the help of the COVERING relation, we can represent cases of *incorporated* meaning: words expressing additional aspects of the proposition, which could otherwise be conveyed by separate words. Incorporation occurs most prominently with verbs: To *drive* implies not only movement, but also the nature of the associated instrument, which can otherwise be expressed separately as *go by car*.[6] Thus, while *go* covers merely the concept MOVE, *drive* is only applicable if the instrument is a car (selectional restriction), and it covers in addition to MOVE the INSTRUMENT role and its filler CAR. A generator that has only these two verbs at its disposal for verbalizing MOVE events can always use *go* and express the instrument, if given in the proposition, with a *by* -PP. In cases where the instrument is given and known to be a car, there is the additional

[6] Of course, another difference between the two is that *drive* implies agency, whereas *go by car* can also be said of the passengers.

option of covering everything with *drive*.

```
(tell (:about                                  (tell (:about
  go_i  MOVE                                     drive_i  MOVE
  (covering move_d)                              (instrument car_d)
  (e-lexeme go_l)))                              (:filled-by covering
                                                    move_d instrument_d car_d)
                                                 (e-lexeme drive_l)))
```

Fig. 2. Representing incorporation with the COVERING role

In general, we cannot expect the relations in the KB to be identical with thematic roles needed in lexical semantics. That is, when a KB predicate is mapped to a verb, the associated KB relations have to be mapped to thematic roles as well. This can be a trivial step, for instance when only an agent and a patient are involved, but the mapping is nonetheless required. Consider the event of a person named Tom putting water into a tank; this can be described, inter alia, as *Tom fills the tank with water* or as *Tom pours water into the tank*.[7] Clearly, they both ought to be generated from the same representation, so that we need to map the tank and the water to different case roles depending on the verb we intend to use. Or, consider another example from the automobile manual domain:

Twist the cap until it stops.
Drehen Sie den Deckel bis zum Anschlag.

Again we have the same content yet different verbalizations—and in fact it is not possible to translate the constructions literally. Therefore, at the deepest level of representation the condition that terminates the twisting action has to be represented in a way neutral between the two linguistic expressions. It seems that using a semantic thematic role that could equally pertain to *until it stops* and *bis zum Anschlag* would amount to a confusion of representation levels.

2.3 Towards multilingual generation: contrastive lexicology

As a contribution to a (partial) contrastive dictionary, Schwarze and François [1985] analyzed 40 French and 25 German verbs belonging to the semantic field REPAIR/HEAL and provided translations based on fine-grained distinctions of meaning. The semantic field is characterized by the common fact that some entity has previously undergone a change to a less desirable state and is now being restored to the original state, either fully or to a certain degree only.

[7] In German, you can use the same verb *füllen* in two different configurations (locative alternation): *Tom füllt den Tank mit Wasser* or *Tom füllt Wasser in den Tank*.

The authors found that the meanings of the verbs differ along the following dimensions:

- the class of entities to which the verb can be applied,
- the class of entities that cause the improvement,
- the type of defect/disease,
- the method employed in causing the improvement,
- the degree of meticulousness in performing the repairing/healing action.

Starting from Schwarze and François's results, we taxonomized the respective class restrictions and defined instances representing the verb meanings, as outlined in the last section. The dimensions suggested by the authors needed further refinement, though. The restriction on the patient-role, for instance, is usually a more-or-less general class like 'animate being' or 'shoe'; but for *repriser* it is 'object made of fabric', which does not fit into the same taxonomy—objects made of fabric can belong to many different classes. For this and a few similar cases, we need to introduce an additional role representing the substance that the patient is made of.[8] (As with all the other roles, if no such restriction is relevant for a verb, it is simply left undefined, which means that the word can be used to describe any event, whether that particular role is present or not.) Thereby we capture the subsumption relationships holding between the verbs, and our lexical option finder can for a given repair-event determine the range of applicable verbs, from the most specific to the most general. For example, if the event is DARNing a sock, we find the German verbs *stopfen*, *ausbessern*, *flicken*, and the French *repriser*, *raccommoder*, *rafistoler*—which do not have pairwise *identical* definitions, though!.

Other issues in contrastive lexicology that multilingual generation needs to deal with include the presence of lexical *gaps* in one language, or the mapping of a word in one language to several words in another one. For example, both German and French make a distinction between *wissen* and *kennen*, or *savoir* and *connaitre*, whereas English uses *to know* for both senses (*know that something is true* vs. *know a person, a place*, etc.).

Also, languages can distribute the elements of a proposition differently across lexical items. Occasionally, this results in the well-known cases of "head switching", as in *Sally likes to swim* vs. *Sally schwimmt gern*. Verbs are especially prone to covering different material; Talmy [1985] has collected a wealth of examples showing, for instance, the tendency of English motion verbs to express the MANNER of motion, while Romance languages prefer to incorporate the PATH into the verb. Thus, in English we find *The bottle floated into the cave*, which in Spanish is rendered *La botella entró a la cueva flotando* (lit. *The bottle entered the cave floatingly*). Similarly, English *He swam across the river* corresponds to

[8] Not unexpectedly, this taxonomization also illuminated the principal limitations of representing meaning differences *solely* on the basis of a conceptual hierarchy: Certain aspects of denotational meaning escape the framework of mere subsumption. For discussion and an outline of a possible solution, see [DiMarco, Hirst, Stede 1993].

French *Il a traversé la rivière a la nage*, where the MANNER of motion again is external to the verb.

But verbs are not the only words that invite cross-linguistic comparison. In much the same way as they can incorporate different aspects of a situation, do we occasionally find compound nouns emphasizing different aspects of an object. What in British English is a *lightning conductor* the Americans prefer to call a *lightning rod*—one word focuses on function, the other on shape. The German *Blitzableiter* is close to the British version, but the morpheme *ab* adds the aspect that the lightning is being lead *away*. In Polish, *piorunochron* means *lightning protector*, thus emphasizing not its physical function but rather its utility.

Further, also non-lexical, cases of *divergences* between languages and their impact on interlingual machine translation (where the problems are essentially the same as in multilingual generation) are discussed by Dorr [1993].

2.4 Linking the lexicon to the KB: a concrete example

Often, translating a sentence into a closely related language yields only minor lexical differences, for instance slightly different word meanings as outlined above. Occasionally, however, the same event is described with words that no dictionary lists as translations, because the words used emphasize different aspects. Recall the example from the automobile manual, mentioned in section 1: *disconnect* vs. *abziehen*. In other sections of the same manual, one notices a tendency for using the general *remove* in English, where the German version gives more specific verbs like *herausziehen*, *herausnehmen*, *abnehmen*. In the following, we suggest an explanation for the contrast, and describe a representation of the action in our system, which will allow for the production of the relevant lexical options for the very short phrase *disconnect the spark plug wire* and its German counterpart.

The KB, designed for the automobile domain, models a connection of engine parts as a pair of objects that can be in one of a few discrete states ('disconnected' or 'connected', and for some subtypes also 'loosely connected'), with operations that switch between those states, such that some reasoning about actions with connections in terms of their start and goal states is possible. 'Connection' specializes into 'plug connection', 'screw connection', 'cap connection', and others, which have a few particular properties. For more details, see [Rösner, Stede 1992]; here we omit all the properties that do not pertain to the explanation of the example at hand. The lower part of figure 3 gives the relevant subset of this representation, showing ISA-links and relations (dotted lines) between concepts. Connections are in turn a specialization of general associations between objects—for instance, a book lying on a table—, which is modeled in the upper part of the figure.

We attach lexical items to these concepts and roles as follows. The verb *to remove* attaches to DISSOCIATE and covers this concept; it optionally covers O2, the part of the association that the other one (O1) is being removed from, as well as the HAS-SOURCE relation, because both *Remove the wire* and *Remove the wire from the spark plug* are all right. We can furthermore attach *to take off* to DISSOCIATE, which always covers the HAS-SOURCE relation (expressed

by *off*) and optionally also its filler. It comes with the additional selectional restriction that O2 be a SURFACE-LOC, which is not shown in the figure, but supposed to be the class of objects that are not containers: we can *remove* a book from a bag, but not, in this sense, $*$*take* the book *off* the bag. To DISCONNECT, we attach *to disconnect*, covering this concept and optionally the C2-part of the CONNECTION and the HAS-SOURCE relation. *To pull off* attaches to DISSOCIATE+PULL, thereby meaning a removal that is at the same time a pulling action; its covering behavior is the same as that of *take off*. In the figure, the area covered by lexical items is surrounded by dashed lines. To avoid confusion, only the maximum covering areas are given for the words, i.e., those including the optionally covered parts.

When the proposition to be lexicalized is an instance of UNCAP with the PATIENT being an instance of CAP, all the aforementioned lexical items are candidates for expressing the action (presuming that CAP is a subtype of SURFACE-LOC), and are supplied via inheritance. As for the German vocabulary, we have *entfernen* with the same definition as *to remove* and *abziehen* with the same as *to pull off*. *Trennen* attaches to DISCONNECT, but, unlike *to disconnect* does not optionally cover the C2; it has to be overtly expressed: $*$*Das Zündkabel trennen.* Again, all these items apply to the expression of the action; hence the choice has to be made on the grounds of connotational features and with regard to the overall length of the sentence. In an instruction manual, there is a strong preference for being concise, which does not favor the use of *trennen*, since it would require the additional verbalization of the CAPPED; this is unnecessary, because it is obvious to the reader of the manual. Both *abziehen* and *entfernen* remain as candidates, and the former wins because it is more specific. On the English side, both *take off* and *pull off* are marked as more informal than *remove* and *disconnect*, which is not favorable in the manual genre. Again, *disconnect* wins on the grounds of specificity.

In short, maximum specificity alone is not enough to govern lexical choice; collocations, connotational features and genre-specific preferences (e.g., sentence length, conciseness) need to be accounted for as well. The task of the lexical option finder is to first supply all the "denotationally correct" lexical items, before such a choice can be made. The details of the preferential choice process are, as indicated earlier, not of our concern in this paper, though.

3 A lexical option finder

In this section we discuss how the representation scheme that links KB objects to lexical items can be used in the generation process. If generation is based on a rich dictionary, offering an array of synonymous or nearly synonymous lexical items for expressing a certain concept, lexical decisions will interact not only with one another, but also with other, syntactic decisions to be made by the system. Therefore we take the first step in the generation process to be the determination of *lexical options*: the set of all words or phrases that can cover some part of the proposition to be expressed. This set will include

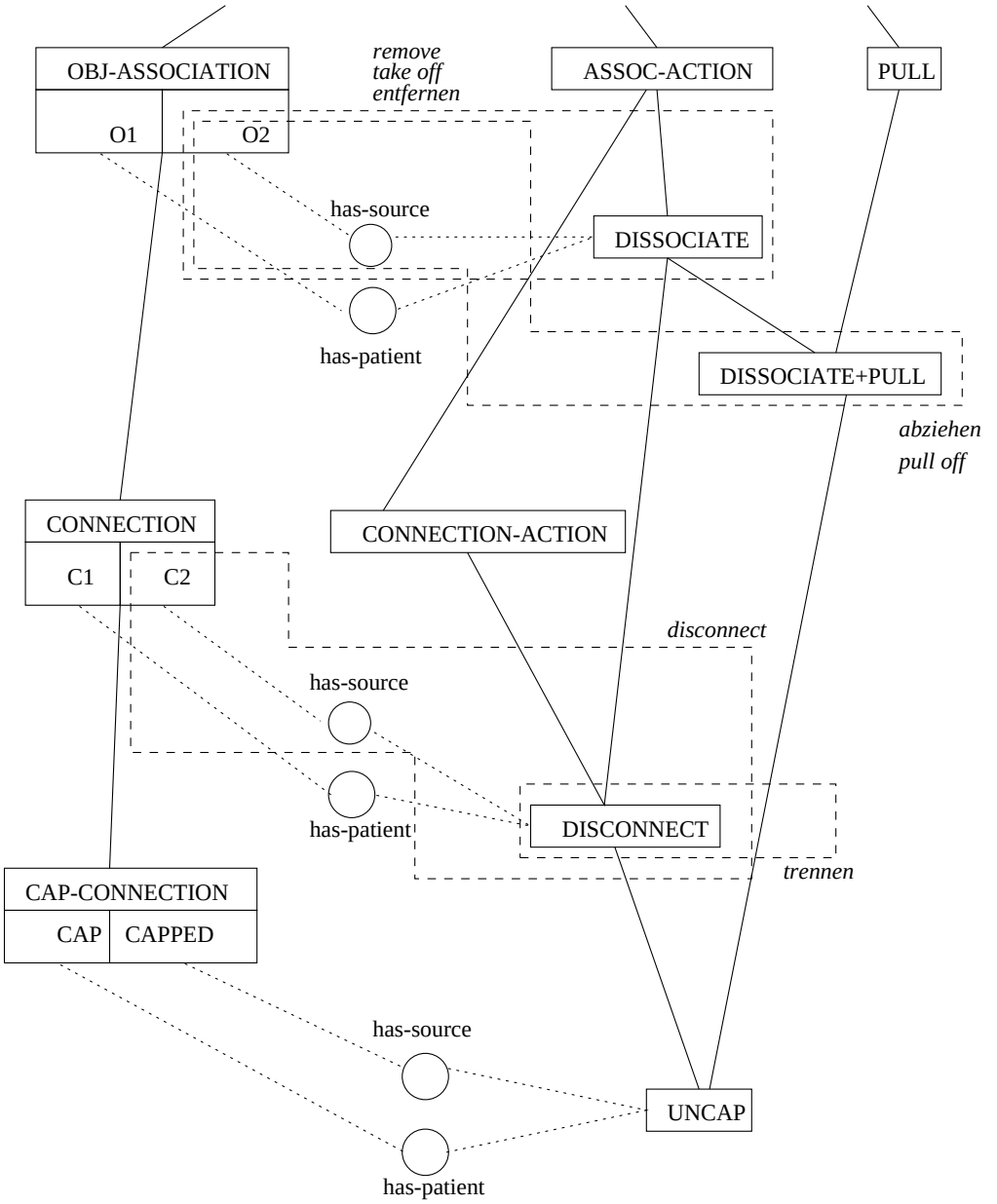

Fig. 3. Extract from 'connection' part of KB

- items with the same denotation, but different connotational features: *die, kick the bucket*;
- items with similar denotation, differing in terms of their semantic specificity and possibly connotationally: *darn, fix*;
- items that incorporate elements of the proposition that could otherwise be expressed separately: *affect adversely, impair*.

From this pool, the generator can subsequently select a set of items that covers the whole proposition, employing a preference function that evaluates the connotational features, and that can be combined into a grammatical utterance.

Since we have defined the link between concepts and relations and lexical items as *instances*, the task of finding all lexical options reduces to one of database retrieval and can therefore conveniently exploit the functionality of LOOM. The proposition to be expressed is an instantiated concept with a number of roles and their fillers, which in turn may have some properties (i.e., additional roles and fillers). To find all the applicable lexical items, we traverse the participating concept-instances in turn and invoke a function FIND-LEXITEMS that retrieves those lexical items whose instance-definition subsumes that of the instance to be verbalized.

The behavior of FIND-LEXITEMS, when applied to instance I of concept C is characterized as follows: It constructs a LOOM database query term that finds all instances of lexical items whose type is more general than or the same as C and which have no role associated to them that I does not have; otherwise, the lexical item would imply more than is warranted by the proposition. But conversely, I may very well have roles that are not defined for a lexical item, yet the item may be appropriate; in this case, the item is more general, i.e., it conveys *less* than warranted by the proposition—which, for some reason or another, might be desired, as in the *disconnect/abziehen* example. The final constraint for FIND-LEXITEMS is that for all of Is roles, if they are also defined for the lexical item, the type of the filler must be subsumed by that of the item's filler. This ensures that selectional restrictions imposed by the lexical item, if any, are obeyed.

More formally, the function can be expressed as follows. Let I denote the instance that we apply FIND-LEXITEMS to, and $t(I)$ a function that returns the type of instance I. $C_1 \succeq C_2$ denotes subsumption, i.e., C_1 is more general than C_2; $R(i_1, i_2)$ means that relation R holds between two instances. Then we are looking for the set of interface instances i such that three conditions are fulfilled:

$$\{i|\ t(i) \succeq t(I) \tag{1}$$

$$\wedge\ \forall R\ [R(i,x) \rightarrow R(I,y)] \tag{2}$$

$$\wedge\ \forall R\ [R(I,x) \rightarrow [R(i,y) \rightarrow t(y) \succeq t(x)]]\ \} \tag{3}$$

The lexical option finder applies FIND-LEXITEMS to each concept-instance in the proposition and thereby also finds options for incorporation. It maps the elements of the input proposition to the instances that serve in the definitions of the COVERING relation and can thereby return the set of applicable lexical items together with those parts of the proposition they cover.

```
> (tellm (cap wire_d) (capped sparkplug_d))
OK

> (tellm (:about unplug_d unplug
         (has-patient wire_d) (has-source sparkplug_d)))
|I|MY-EVENT(UNPLUG)

>(find-lexitems (fi unplug_d) 'english)
((("remove")     (|I|UNPLUG_D(UNPLUG)
                  |I|SPARKPLUG_D(CAPPED)
                  |I|HAS-SOURCE_D(HAS-SOURCE)))
 (("remove")     (|I|UNPLUG_D(UNPLUG)))
 (("take off")   (|I|UNPLUG_D(UNPLUG)
                  |I|SPARKPLUG_D(CAPPED)
                  |I|HAS-SOURCE_D(HAS-SOURCE)))
 (("take off")   (|I|UNPLUG_D(UNPLUG)
                  |I|HAS-SOURCE_D(HAS-SOURCE)))
 (("disconnect") (|I|UNPLUG_D(UNPLUG)
                  |I|SPARKPLUG_D(CAPPED)
                  |I|HAS-SOURCE_D(HAS-SOURCE)))
 (("disconnect") (|I|UNPLUG_D(UNPLUG)))
 (("pull off")   (|I|UNPLUG_D(UNPLUG)
                  |I|SPARKPLUG_D(CAPPED)
                  |I|HAS-SOURCE_D(HAS-SOURCE)))
 (("pull off")   (|I|UNPLUG_D(UNPLUG)
                  |I|HAS-SOURCE_D(HAS-SOURCE))))

>(find-lexitems (fi unplug_d) 'german)
((("entfernen") (|I|UNPLUG_D(UNPLUG)
                 |I|SPARKPLUG_D(CAPPED)
                 |I|HAS-SOURCE_D(HAS-SOURCE)))
 (("entfernen") (|I|UNPLUG_D(UNPLUG)))
 (("trennen")   (|I|UNPLUG_D(UNPLUG)))
 (("abziehen")  (|I|UNPLUG_D(UNPLUG)
                 |I|SPARKPLUG_D(CAPPED)
                 |I|HAS-SOURCE_D(HAS-SOURCE))))
```

Fig. 4. Sample run of lexical option finder

Figure 4 shows a sample run that begins with defining the proposition to be expressed. Two object instances are created: `wire_d` of type `cap`, and `sparkplug_d` of type `capped` (compare figure 3). The event instance `unplug_d` is of type `unplug` and has two roles filled by the objects previously created. Now, the lexical option finder is applied twice to the event-proposition, first to find the applicable English lexemes, then the German ones. The output is a list of pairs consisting of a lexical item and the list of those instances of the proposition that the item covers. E.g., the first lexical item found, `remove`, covers the event concept

`unplug_d` as well as the spark plug and the `has-source` relation connecting the two (this covering corresponds to the dashed box at the top in figure 3).

Items appearing twice have optional coverings, and the choice is left to the generator. Here, the second `remove` covers only the event concept itself, so if the generator takes this option, it has to express the sparkplug explicitly. In practice, of course, more synonyms, differing in connotation, may be found, depending on the richness of the lexicon.

4 Three representation levels for lexicalization

4.1 Summary of the approach

For multilingual generation from a common knowledge base, we need to go beyond the common procedure of mapping lexical items one-to-one onto concepts. We have presented a flexible way of associating lexical items with *configurations* of concepts and roles, and a lexical option finder that determines all items whose denotation is a subset of the proposition to be expressed. With the COVERING relation we have a measure for determining when all the semantic material has been verbalized. We also use it to handle incorporation, which may be optional as well, and by means of the COVERING statements the set of lexical options implicitly holds the information on the various possibilities of dividing the input structure into verbalizable pieces, a task that has been called the 'chunking problem'.

The effect of linking lexical items to concepts *and* roles is that we can represent more fine-grained semantic distinctions than those made by the concepts only: similar lexical items all map onto the same—fairly general—semantic predicate, and their differences are represented with associated roles and fillers. The lexical option finder determines the set of all semantically appropriate items, ranging from specific to general meaning. This scheme, driven by the desire to keep the conceptual KB free from language-specific knowledge, ensures that the lexicalization process is independent of the specificity of the input proposition as constructed by some application program: The generator will always find a general word, and in those cases where the input contains more detailed semantic information, it will also come up with appropriate specific lexical items.

4.2 Representation levels

We distinguish three different levels for representing the information necessary for lexicalization in generation: The **lexicon entry** for a word or phrase holds the syntactic features needed by the grammar as well as stylistic attributes (formality, genre, etc.) to be used by a preferential word choice process. A language-independent **conceptual representation** (knowledge base) contains the propositional content of the utterance to be generated; the grain-size of the concepts is determined by the requirements of the underlying application program, which may also use the KB for reasoning purposes. To mediate between KB and lexicon entries, we define **interface instances** that represent the level of lexical

semantics by mapping a conceptual configuration to one or more lexicon entries (synonyms). The definition of an interface instance corresponds to the conditions necessary for using the associated word(s), and the COVERING role points to those parts of the definition that are actually expressed by the word. These may or may not coincide: in the instances for *go* and *drive* in fig. 2 the complete definition is covered, whereas the various DIE instances in fig. 1 would not cover the selectional restrictions for the EXPERIENCER (the COVERING role is not listed in the figure)—the filler of that role needs to be expressed separately. Accordingly, there are two different kinds of selectional restrictions: some in the concept definitions in the KB, and some in the interface instances. The KB restricts the experiencer of DIE to be of the type ANIMATE-BEING, and interface instances for specific words further restrict this type to ANIMAL or HUMAN.

Lexicalization, in the approach outlined in this paper, means mapping a language-neutral conceptual representation to a language-specific word-based representation that covers all aspects of the original one. The process is driven by the lexicon: by (efficiently) retrieving the interface instances associated with the concepts of the proposition, we chunk the message into pieces corresponding to words or phrases; and a preferential choice process can make the final decisions and put together specifications that can serve as input to language-specific surface generators, which take care of grammatical constraints (in the application where the lexical option finder is being used, the TECHDOC system [Rösner, Stede 1992], this function is performed by the PENMAN sentence generator [Penman 1989]).

The benefit of assigning such a prominent role to the lexicon in the generation process lies in the amount of linguistic variation a generator can achieve. For the same semantic representation, LOF determines the range of possible verbalizations, where variety stems from synonyms (*die – kick the bucket*), from words that can incorporate the meaning of other words (*drive – go by car*), from words that map case roles differently (*fill – pour*), and from words of different specificity (*dog – poodle*), which can also yield fairly different cross-linguistic lexicalizations of the same event (*disconnect – abziehen*). The set of lexical options, including words with identical, different, or overlapping denotation, is a prerequisite for genuine lexical *choice*, which is needed for tailoring generator output to the user and the utterance situation—a task that most language generators so far have "compiled out", i.e. encoded implicitly by mapping concepts directly to lexical items.

Acknowledgements

Financial support from FAW Ulm, the Natural Sciences and Engineering Research Council (NSERC) of Canada, and the Information Technology Research Centre (ITRC) of Ontario is gratefully acknowledged. For their helpful comments on earlier drafts of this paper, I thank Graeme Hirst, Dietmar Rösner, Brigitte Grote, and several anonymous reviewers.

References

[Bierwisch, Schreuder 1992] M. Bierwisch, R. Schreuder. "From Concepts to Lexical Items". In: *Cognition* 42(1-3), 1992.

[Cruse 1986] D.A. Cruse. *Lexical Semantics.* Cambridge University Press, 1986.

[DiMarco, Hirst, Stede 1993] C. DiMarco, G. Hirst, M. Stede. "The Semantic and Stylistic Differentiation of Synonyms and Near-Synonyms". In: *Working notes of the AAAI Spring Symposium on Building Lexicons for Machine Translation*, Stanford University, March 1993.

[Dorr 1993] B. Dorr. "Interlingual Machine Translation: A Parameterized Approach". In: *Artificial Intelligence* 63, pp. 429-492, 1993.

[Emele et al. 1992] M. Emele, U. Heid, S. Momma, R. Zajac. "Interactions between Linguistic Constraints: Procedural vs. Declarative Approaches". In: *Machine Translation* 7(1-2), 1992.

[Garza-Cuarón 1991] B. Garza-Cuarón. *Connotation and Meaning.* Mouton de-Gruyter, Berlin/New York 1991.

[Horacek 1990] H. Horacek. "The Architecture of a Generation Component in a Complete Natural Language Dialogue System". In: R. Dale, C. Mellish, M. Zock (eds.): *Current Research in Natural Language Generation.* Academic Press, London 1990.

[Hovy 1988] E.H. Hovy. *Generating Natural Language under Pragmatic Constraints.* Lawrence Erlbaum, Hillsdale (NJ) 1988.

[Levelt 1989] W.J.M. Levelt. *Speaking. From Intention to Articulation.* MIT Press, Cambridge 1989.

[MacGregor, Bates 1987] R.M. MacGregor and R. Bates. "The Loom Knowledge Representation Language". University of Southern California/ISI, Tech. Rep. ISI/RS-87-188.

[Marcus 1987] M. Marcus. "Generation Systems Should Choose Their Words". In: Y. Wilks (ed.): *Theoretical Issues in Natural Language Processing.* New Mexico State Univ., Las Cruces 1987.

[McDonald 1991] D.D. McDonald. "On the Place of Words in the Generation Process". In: C.L. Paris, W.R. Swartout, W.C. Mann (eds.): *Natural Language Generation in Artificial Intelligence and Computational Linguistics.* Kluwer, Dordrecht, 1991.

[Miezitis 1988] M.A. Miezitis. "Generating Lexical Options by Matching in a Knowledge Base". Technical Report CSRI-217, Dept. of Computer Science, University of Toronto, 1988.

[Nirenburg, Nirenburg 1988] S. Nirenburg and I. Nirenburg. "A Framework for Lexical Selection in Natural Language Generation". In: *Proceedings of the 12th International Conference on Computational Linguistics (COLING)*, pp. 471–475, Budapest 1988.

[Nogier, Zock 1992] J.F. Nogier, M. Zock. "Lexical Choice by Pattern Matching". In: *Knowledge Based Systems* 5(3), 1992.

[Novak 1991] H.-J. Novak. "Integrating a Generation Component into a Natural Language Understanding System". In: O. Herzog, C. R. Rollinger (eds.): *Text Understanding in LILOG.* Springer, Berlin/Heidelberg 1991.

[Novak 1993] H.-J. Novak. "Die LILOG-Ontologie aus Generierungssicht". In: G. Klose, E. Lang, Th. Pirlein (eds.): *Ontologie und Axiomatik der Wissensbasis von LILOG.* Springer, Berlin/Heidelberg 1992.

[Penman 1989] *The Penman documentation.* Unpublished documentation for the Penman language generation system. University of Southern California/ISI, 1989.

[Reiter 1990] E. Reiter. "Generating Descriptions that Exploit a User's Domain Knowledge". In: R. Dale, C. Mellish, M. Zock (eds.): *Current Research in Natural Language Generation.* Academic Press, London 1990.

[Rösner, Stede 1992] D. Rösner, M. Stede. "TECHDOC: A System for the Automatic Production of Multilingual Technical Documents". In: G. Görz (Ed.): *KONVENS 92* – Proceedings of the First German Conference on Natural Language Processing. Springer, Berlin/Heidelberg 1992.

[Schwarze, François 1985] C. Schwarze, J. François. "Heilen und Reparieren". In: C. Schwarze (ed.): *Beiträge zu einem kontrastiven Wortfeldlexikon Deutsch – Französisch.* Narr, Tübingen 1985.

[Sondheimer, Cumming, Albano 1990] N. Sondheimer, S. Cumming, and R. Albano. "How to Realize a Concept: Lexical Selection and the Conceptual Network in Text Generation". In: *Machine Translation* 5(1), pp. 57–78, 1990.

[Stede 1993] M. Stede. "Lexical Choice Criteria in Language Generation". In: *Proceedings of the Sixth Conference of the European Chapter of the ACL*, pp. 454–459, Utrecht 1993.

[Stede 1995] M. Stede. "Lexicalization in Natural Language Generation: A Survey". In: *Artificial Intelligence Review* 8:309-336, 1995.

This article was processed using the LaTeX macro package with LLNCS style