Gefördert vom
bmb+f
Bundesministerium für
Bildung, Wissenschaft,
Forschung und Technologie

Verbmobil
Verbundvorhaben

Resolution on Demand

Bianka Buschbeck-Wolf

Universität Stuttgart

Vm

Report 196
May 1997

May 1997

Bianka Buschbeck-Wolf

Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart
Azenbergstraße 12
D – 70174 Stuttgart

Tel.: (0711) 121 - 1367
Fax: (0711) 121 - 1366
e-mail: `bianka@ims.uni-stuttgart.de`

Gehört zum Antragsabschnitt: 3.1 Konstrastive Phänomene und Kodierung

Die vorliegende Arbeit wurde im Rahmen des Verbundvorhabens Verbmobil vom Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie (BMBF) unter dem Förderkennzeichen 01 IV 701 N3 gefördert. Die Verantwortung für den Inhalt dieser Arbeit liegt bei der Autorin.

Abstract

Following the strategy of *resolution on demand*, the transfer component triggers inference processes in analysis and resolution components whenever it needs more information to solve a particular translation task. This report presents a collection of specific requests to discourse semantics, dialogue processing and context evaluation and illustrates them with a series of examples.[1]

[1]The work presented here is strongly influenced by discussions with my colleagues of the Verb*mobil* project. I would like to express my gratitude to Manfred Stede, Michael Dorna, Jan Alexandersson, Franz Beil, Johan Bos, Martin Emele, Elisabeth Maier, C.J. Rupp and Michael Schiehlen for most valuable comments on the topics of this report.

Contents

1 Introduction

Speech-to-speech translation is very challenging task. Spoken language is highly ambiguous because of a large number of ambiguous words, anaphora, deictic expressions and ellipsis. To achieve an acceptable translation quality ambiguity and anaphora resolution as well as ellipsis reconstruction are of major importance. In order to reduce the number of translation problems, it is most reasonable to focus on restricted application domains. In Verb*mobil*, we consider the domain of appointment scheduling and travel planning.

Since it is too expensive to resolve all ambiguities, anaphora, etc., we have developed the concept of *resolution on demand.* Besides the strategy of preserving scopal and attachment ambiguities by underspecification [Bos, 1995, Egg and Lebeth, 1995, Schiehlen, 1996] this technique minimizes the analysis efforts.

The need of particular resolution procedures can be best recognized within the transfer component [Dorna and Emele, 1996b], i.e., by the contrastive data. Here, the cases of alternative translation candidates or of lacking information are identified. Hence, it is the appropriate place for formulating requests to other components in order to obtain the information that is needed to solve particular translation tasks.

In order to choose one among different translation correspondences, we have to explore the contexts in which the one or the other target language (TL) expression is used. For the resolution of many translational ambiguities the local semantic context is sufficient (see Section 2.2). It allows to formulate restrictions on the sort or the semantic type of a predicate, on its scopal embedding, mood, number, aktionsart, etc., [Buschbeck-Wolf and Tschernitschek, 1996] as well as on prosodic information [Lieske et al., 1997].

However, there are cases in which the transfer component needs more global information to choose a particular TL correspondence. This is, e.g., information about the antecedent of an anaphor or an ellipsis, domain-specific world knowledge, speech act and discourse stage information [Alexandersson et al., 1997a], as well as information about the pragmatic function of discourse particles [Stede and Schmitz, 1997]. This information can be obtained either from the discourse semantics, the dialogue processing or the context evaluation component. By anchoring in the transfer rules specific requests to these three components, the transfer triggers the required resolution procedures. Figure 1 in Section 2 shows the corresponding interfaces of the transfer module by bold lines.

After a short description of the transfer module in Section 2, we present a collection of translation problems that require information from other components. We divide the phenomena into the following three parts:

1. Translation problems that concern the dialogue processing (see Section 3)
2. Translation problems that concern the discourse semantics (see Section 4)
3. Translation problems that concern the context evaluation (see Section 5)

We demonstrate each kind of requirement with some examples. It is not the aim of this report to present an exhaustive collection of all resolution requests of the transfer component. We consider to investigate, first of all, the language pair German–English. For a detailed discussion of the problems that arise when translating Japanese into German and vice versa, we refer to [Fujinami and Nanz, 1997] and [Fujinami, 1997].

2 Transfer

For better understanding of the following sections, let us briefly introduce our semantic-based transfer approach.

2.1 Interfaces of the Transfer Module

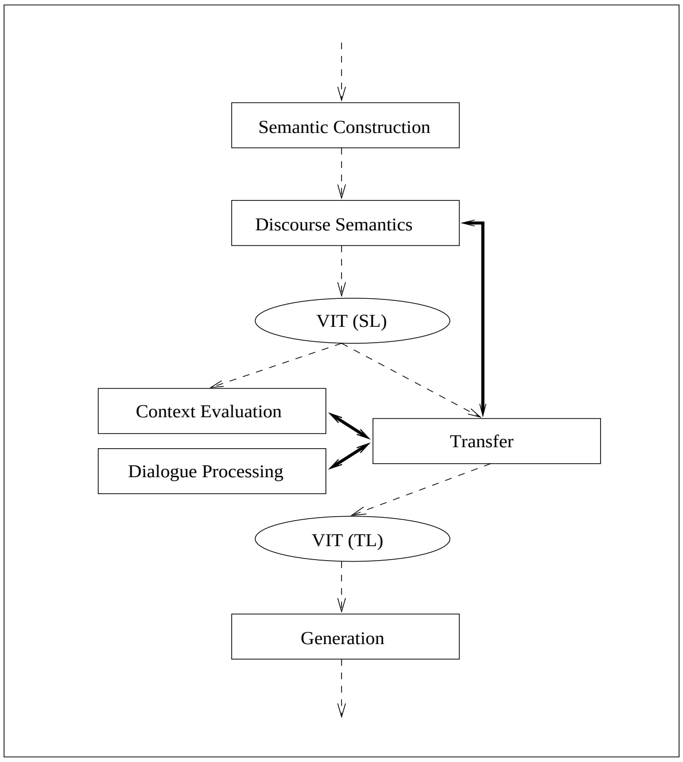

Figure 1: Interfaces of the Transfer Component

Figure 1 shows the embedding of the transfer module in the Verb*mobil* system. The semantic construction [Bos et al., 1996] produces underspecified semantic representations (see Section 2.2) that are partially resolved by the subsequent discourse semantics module. The latter provides the input to the transfer component. To choose the appropriate translations, the transfer component obtains additional information via interfaces from the dialogue processing that keeps track of the dialogue history and provides speech act information [Alexandersson et al., 1997b]; from the discourse semantics that resolves anaphora, ellipses and scope [Gambäck and Bos, 1997]; and from the context evaluation that supports the resolution of problematic lexical ambiguities [Quantz and Schmitz, 1994]. The transfer module reports its TL semantic representations to the generator which maps them to TL expressions [Kilger and Finkler, 1995].

2.2 Semantic Representation

The semantic representation together with additional information, such as number, tense, aspect, prosodic accent, sorts, etc., is encoded in a multi-dimensional data structure called Verb*mobil* Interface Term (VIT) [Dorna, 1996]. This single information structure serves as interface representation for all components that operate on semantic structures (see Figure 1).

Let us shortly describe the semantic slot of a VIT, since it is the part to which the transfer mappings apply. The semantic slot contains a set of semantic predicates. Each one has a unique label `l` which is used as an address for linking information within and between multiple levels of a VIT. Besides their label, referential predicates introduce an instance `i`. Argument roles and modifier relations are represented in a Neo-Davidsonian way [Parsons, 1991]. Semantic operators like quantifiers, modals or scopal adverbs take extra label arguments for referring to other elements which are in the relative scope of these operators.[2]

(1a) Wir wollten das Gespräch doch verlegen?
(1b) We wanted to put off the meeting, didn't we?

Consider example (1a). This utterance is assigned the representation in (2). The relative scope of the sentence mood operator `ynq(l0,l1)` (yes-no question), the control verb `wollen(l1,i1)` and the pragmatic adverb `doch(l2,l3)` has been fully resolved by using the explicit labels of other predicates, these are `l1,l2` and `l3`.

(2)
```
ynq(l0,l1),
wollen(l1,i1),
arg1(l1,i1,i2),
arg3(l1,i1,l2),
doch(l2,l3),
verlegen(l3,i3),
arg1(l3,i3,i2),
arg3(l3,i3,i4),
pron(l4,i2,sp_he),
def(l5,i4,l6),
gespraech(l6,i4)
```

[2]For representational details see [Kasper et al., 1997].

2.3 Transfer Rules

The transfer relates underspecified source language (SL) and TL semantic descriptions by the application of bilingual transfer rules.

(3) `SL_Sem # SL_Cond TauOp TL_Sem # TL_Cond.`

The general form of a transfer rule is shown in (3). It establishes the equivalence between sets of SL semantic entities `SL_Sem` and sets of TL semantic entities `TL_Sem`. The operator `TauOp` indicates in which direction a rule is applied, i.e., bi-directional ($\leftrightarrow$) or uni-directional ($\rightarrow$ or $\leftarrow$).[3]

(4) `wollen(L,I) <-> want(L,I).`

(4) shows a simple transfer rule for the unambiguous predicate *wollen*. The capitalized symbols `L` and `I` stand for logical variables which are bound to concrete values when applying a rule to a given input.

To resolve translational ambiguities, the rules are optionally provided with a condition part (`SL_Cond` and `TL_Cond`). The `#` sign separates it from the mapping part.[4] The condition part restricts the transfer mappings to the relevant context. It contains only tests. On the one hand, these are tests on particular semantic predicates, on their sort, their semantic class, etc., i.e., tests on information that is contained in the actual VIT representation. On the other hand, these are interface predicates that encode calls to other components for providing information that is not part of the actual VIT.

Consider the ambiguous verb *verlegen*, which gets translated, among others, into *put off*, *publish*, *move* and *misplace*. The different readings can be identified by sortal restrictions on the theme argument (`arg3`). The rule that is relevant for the translation of *verlegen* in (1) is shown in (5). Its mapping to *put off* is constrained to contexts in which the theme argument is realized by a noun of the sort `situation`.

(5a) `verlegen(L,I) # arg3(L,I,I1),sort(I1,situation) <-> put_off(L,I).`

2.4 Templates for Transfer Rules

For building the transfer rule bases we introduced a template mechanism that ensures the adaptation and reusability of transfer rules independently of the concrete front and back end of the transfer component [Buschbeck-Wolf and Dorna, 1997].

For the sake of simplicity, the examples given in the following sections are mostly formulated by the use of templates. Let us briefly introduce them.

---

[3] A rule application reduces the SL input by the set of semantic entities in `SL_Sem` if they form a matching subset of the input. On the other hand, the TL semantic entities `TL_Sem` are added to the TL output (see [Dorna and Emele, 1996a] for details).

[4] The condition to the left of `TauOp` restricts the application direction $\rightarrow$, and vice versa.

There are two kind of templates:

1. Templates for transfer equivalences
   (the template call is prefixed by the `@` operator)

2. Templates for frequently used conditions
   (the template call is prefixed by the `+` operator)

(6) shows the template that correspond to the rule in (4).

(6)
```
@verb(wollen,want,L,I).
```

(7) gives the definition of a template for the transfer of verbs. Its usage is restricted to the semantic type the predicate belongs to. In our case, it is `v` for verbs. The variables `SL` and `TL` are replaced by the incoming verbs when calling the template.

(7)
```
verb(v(SL),v(TL),L,I) := SL(L,I) <-> TL(L,I).
```

This is a very simple case. Templates are more useful for capturing complex changes in the semantic structure of the involved languages, such as argument switching, head or category switching. We do not want to explain the encoding of these phenomena in detail here, and refer to the examples in the following sections.

Condition templates are applied in the rule's condition part in order to fix frequently occurring restrictions more efficiently. They are used to express, e.g., that a predicate is of a specific sort or semantic type, is modified, is quantified or is embedded in a particular way, etc. As a result, the conditioned rule in (5) can be expressed by (8).

(8)
```
# +arg3_sort(I,situation) <-> true
@verb(verlegen,put_off,L,I).
```

The `true` predicate in either the SL or the TL condition part means that the rule application in the corresponding direction is unrestricted. A $\rightarrow$ or $\leftarrow$ between the SL and TL conditions fixes the mapping direction. This is sometimes necessary in order to suppress undesirable back translations.

2.5 Cascading Disambiguation

For the resolution of translational ambiguities, we follow the principle of *cascading disambiguation*, i.e., the local information available in the actual VIT is used first to constrain the transfer mapping. When it is exhausted, other components, which provide global contextual information, are requested. Let us demonstrate this principle with the verb *vorziehen*, which is either translated into *prefer* or into *schedule earlier*. In (9)–(12), there are some contexts in which its mapping to one of its equivalents is quite obvious.[5]

[5] The contextual trigger is marked in italics.

(9a) Ich würde *den Dienstag* **vorziehen**.
(9b) I would **prefer** Tuesday.

(10a) Ich würde das Treffen *gerne* **vorziehen**.
(10b) I would like to **schedule** the meeting **earlier**.

(11a) Wir *sollten* das Treffen **vorziehen**.
(11b) We should **schedule** the meeting **earlier**.

(12a) Ich würde es **vorziehen**, *am Montag zu kommen.*
(12b) I would **prefer** to come on Monday.

If the theme argument is a time expression, *prefer* is the only correspondence (9). This is captured by the rule in (13). The translation with *schedule earlier* is not feasible then, because it would require a kind of movable object as `arg3` and times, such as Tuesday, are fixed.

(13)
```
# +arg3_sort(I,time) <-> true
@verb(vorziehen,prefer,L,I).
```

In one of its interpretations, *vorziehen* is an attitude verb. If it is modified by an adverb that expresses the speakers attitude, it cannot have an attitude reading itself (10). This excludes *prefer* as translation correspondence (14).[6]

(14)
```
# attitude(L1,I) <-> true
@verb(vorziehen,schedule,L,I) @add_adx(early,I) @add_comp(I).
```

In (11) the *prefer* reading of *vorziehen* is odd for similar reasons as those discussed in the previous example. Here, it occurs within the scope of a modal verb, which also expresses an attitude. The corresponding rule is shown in(15). [7]

(15)
```
# modal(L1,I1,L2),L=<L2 <-> true
@verb(vorziehen,schedule,L,I) @add_adx(early,I) @add_comp(I).
```

If *vorziehen* has a propositional `arg3` realization (12), *prefer* is the appropriate equivalent (16).[8]

(16)
```
# +proposit_arg3(I) <-> true
@verb(vorziehen,prefer,L,I).
```

[6] The predicate `attitude` is an abstraction over all attitude adverbs. It is attached to the verbal predicate by coindexation of the instance `I`. The template `add_adx` introduces a modifier and `add_comp` a comparative predicate for it.

[7] The predicate `modal` checks the semantic type of the incoming verbs. The equation says that the label of `vorziehen` is below the label of the modal, i.e., within its scope.

[8] The template `proposit_arg3` tests the propositional argument realization.

However, these rules are not sufficient to cope with the ambiguity of *vorziehen*. If its `arg3` belongs to the sort `situation` (17) and it does not occur in one of the mentioned contexts (13)–(16), then the local semantic context does not allow to choose between the translations in (17b) and (17c).

(17) Ich würde *das Treffen* am Montag **vorziehen**.
(17a) I would **prefer** the meeting on Monday.
(17b) I would **schedule** the Monday meeting **earlier**.

To resolve this ambiguity, the context evaluation component is consulted. (18) shows the mapping of `vorziehen` to `prefer` under the condition that the German verb subsumes the concept `vorziehen_prefer`. The subsumption test `concept(L)=<vorziehen_prefer` is internally mapped onto a call to the context evaluation module for verifying the prefer reading of *vorziehen* in the given context. Since the ambiguity of this verb holds only if the theme argument `arg3` is of the sort `situation` the condition part contains a corresponding restriction. (19) shows the corresponding rule for the resolution of the move reading of *vorziehen*.

(18)
```
# +arg3_sort(I,situation),concept(L)=<vorziehen_prefer <-> true
@verb(vorziehen,prefer,L,I).
```

(19)
```
# +arg3_sort(I,situation),concept(L)=<vorziehen_move <-> true
@verb(vorziehen,schedule,L,I) @add_adx(early,I) @add_comp(I).
```

3 Transfer Requests to Dialogue Processing

3.1 Dialogue Stage

There are cases in which the information about the actual discourse stage allows to determine the appropriate translation.

(20a) Ja, gut, dann **wiederhole** ich jetzt nochmal.
(20b) All right, I **recapitulate**.

(21a) Könnten Sie das bitte **wiederholen**? (Ich hab's nicht verstanden.)
(21b) Could you **repeat** that, please? (I didn't get it.)

Both verbs *repeat* and *recapitulate* in (20) and (21) are possible translations of *wiederholen*. Usually, *recapitulate* is used to give a summary of a discussion or a part of it, while *repeat* means to do something again. In order to disambiguate *wiederholen*, we could utilize its place of occurrence in the dialogue. While *recapitulate* is usually uttered at the end of a successive appointment scheduling circle, i.e., in the closing part, *repeat* can be used at any stage of the dialogue.

(22a) Wir wollten **einen Termin vereinbaren/ausmachen/abmachen**.
(22b) We have to **make an appointment**.

(23a) Dann sollten wir noch **einen Termin vereinbaren/ausmachen/ abmachen**.
(23b) Then we still have to **fix/agree on a date/time**.

Another example is the translation of the expression *einen Termin ausmachen/vereinbaren/abmachen*. It displays a collocation-like behaviour in the Verb*mobil* domain, since the translation of these verbs and that of *Termin* are strongly interdependent. When used to open a dialogue, the whole expression is translated into *make an appointment* (22). Uttered within the negotiation phase, *fix a date/time* or *agree on a date/time* are the appropriate translations (23).[9]

In the dialogue processing component, meeting scheduling dialogues are divided into three main parts: an `opening`, a `negotiation` and a `closing` phase (see [Alexandersson et al., 1997a], Chapter 3). They are communicated by the predicate in (24).

(24) `dialog_phase(+DialogPhase)`

(25) and (26) show the rules that include a dialogue phase request for the considered examples in (20) and (22).

(25)
```
# dialog_phase(closing) <-> true
@verb(wiederholen,summarize,L,I).
```

(26)
```
# arg3(L,I,I1),indef(L2,I1,G,H),dialog_phase(opening) <-> true
@verb(ausmachen,make,L,I) @noun(termin,appointment,L1,I1).
```

3.2 Propositional Content

Consider the German preposition *bei*. For its occurrences in the Verb*mobil* domain, sortal constraints on its internal argument are, in most cases, sufficient to identify the intended meaning.[10] However, if this argument refers to a human being and the *bei*-PP modifies an attitude, we are faced with an ambiguity between the perspective reading (27a)–(28a) and the unspecified spatial interpretation (27b)–(28b) of the preposition *bei*.

(27) Geht/klappt das **bei Ihnen**?
(27a) Does it suit **you**?
(27b) Is it possible **at your place**?

(28) Das ist schlecht/ungünstig/unmöglich **bei mir**.
(28a) That is bad/inconvenient/impossible **for me**.
(28b) It is bad/inconvenient/impossible **at my place**.

[9] For the choice of the equivalent of *Termin* in (23a) (see Section 3.4).

[10] For instance, a pure spatial reading of *bei* can be identified if the internal argument refers to a thing or a location (e.g., *bei* Berlin - *near* Berlin), and a temporal-spatial one if it is a situation (e.g., *bei* der Vorlesung - *at* the lecture).

In case the theme of an attitude verb (27) or the subject of the copula with a predicative attitude adverb (28) is realized by an event-type pronoun, the reading of the *bei*-PP cannot be identified by anaphora resolution. Since the antecedent is a situation, both interpretations are feasible. To figure out which reading is intended we need information about the propositional content of the actual turn. If the *bei*-PP in the considered context forms part of an utterance in which a location is negotiated, we can heuristically derive that its spatial interpretation is the appropriate one. In any other case, the perspective reading is assumed.

The request of the propositional content is shown in (29). It is attached to the instance of the main event in the current utterance. Values of `PropContent` are the negotiated topics, such as time, location, duration, frequency of a meeting, etc.

(29) `prop_content(+Inst,+PropContent)`

The rules (30) and (31) capture the contexts in which the preposition *bei* occurs in (27) and (28). They both include a request on whether the negotiation topic is a location.[11]

In case of a negative response to the dialogue act request, the rules are not applicable. Then, default rules map *bei* in these contexts to *perspective*.

(30)
```
# sort(I1,person),abstr_passen(L1,I),arg3(L1,I,I2),
+sit_anaphor(I2),prop_content(I,location) -> true
@prep(bei,loc_derived,L,I,I1).
```

(31)
```
# sort(I1,person),support(L1,I,L2),attitude(L3,I2),L3=<L2,
+sit_anaphor(I2),prop_content(I,location) -> true
@prep(bei,loc_derived,L,I,I1).
```

3.3 Illocution or Dialogue Acts

Illocutional information is encoded in dialogue acts. It is important for the translation of discourse particles[12] as well as for generation purposes. Here, we demonstrate the influence of the dialogue act on the translation of the modal particle *vielleicht*.

(32a) Könnten Sie **vielleicht** einen Vorschlag machen?
(32b) Could you make a suggestion?

(33a) Würde Ihnen das **vielleicht** passen?
(33b) Would it suit you?

[11] `abstr_passen` is an abstract predicate that groups together the attitude verbs *gehen, passen* and *klappen* and `attitude` is a predicate type that abstracts over all attitude adverbs, such as *gut, möglich, schlecht*, etc., (see [Buschbeck-Wolf and Tschernitschek, 1996].

[12] One example is the translation of exclamative *bitte*. If it is used in an utterance with the dialogue act `thank` it is translated into *you are welcome*; if not–*please* is the appropriate correspondence. Another case that calls for dialogue act information is the transfer of the particle *ja*. It is mapped to *yes* only if it appears in an `accept` or `confirm` utterance. It corresponds to *well* if being used as an uptake particle. In other cases it is omitted in English or expressed by a question tag [Ripplinger and Alexanderson, 1996].

(34a) Wäre der Montag **vielleicht** möglich?
(34b) Would Monday be possible?

(35a) Ich würde **vielleicht** vorschlagen, Karfreitag oder Karsamstag.
(35b) I would suggest Good Friday or Good Saturday.

In general, the modal particle *vielleicht* is used to express the possibility of a state or event. Then, it is translated into *perhaps, maybe* or *possibly.* However, in spoken German, the literal meaning of *vielleicht* is often not appropriate. In suggestions and requests, it is also uttered for being polite (32a)–(35a). In these cases, it should not be translated (32b)–(35b), since English does not allow the same kind of lexicalization of politeness.

Let us consider this use of *vielleicht* in more detail. When it occurs in a request to make a suggestion (32), the speaker would not expect a yes-no answer, but a concrete proposal. Similarly, when inquiring the hearer's opinion to a made suggestion (33)–(34), the speaker wants to know whether the proposal suits the hearer or not. A "maybe" answer would not satisfy these requests. In (35), the literal meaning of *vielleicht* is odd for pragmatic reasons. When making a suggestion, the speaker usually picks only those times or places that suit him. Otherwise his proposal would not contribute to the negotiation goal. It would contradict the principles of relevance and cooperativity [Grice, 1975].

The translation problem consists in the identification of *vielleicht* as a politeness marker. According to our investigations, this happens when it is uttered in requests or suggestions. This information can be obtained from the dialogue act recognition.[13]

The dialogue act of the current utterance is requested by the predicate in (36). The values of `DialogAct` are defined in [Alexandersson et al., 1997a], Chapter 4.

(36) `dialog_act(+DialogAct)`

In (37)–(39) we show some of the transfer rules that include a dialogue act request in order to delete the particle *vielleicht.* With a restriction on the dialogue act alone, the rules would overgenerate. Therefore, they contain some additional contextual constraints. For the rule in (37), it is the context of making suggestions. (38) and (39) capture the contexts of attitude expressions, i.e., attitude verbs and copula constructions with predicative attitude adverbs (see Section 3.2).[14]

[13] Dialogue act information might also help to disambiguate ambiguous verbs. In case the German verb *annehmen* occurs with an theme argument realized by an event-type pronoun as in "Ich *nehme* es *an*." *accept* and *assume* are feasible translations. Since the antecedent of *es* might be a proposition in both cases, anaphora resolution does not solve the problem. But the dialogue act information does: appearing in an `accept` utterance, which is predicted from the previous context, the translation into *assume* is out.

[14] If *vielleicht* has focus on the suggested time as in "Würde Ihnen **vielleicht** der Montag passen?" (lit.: 'Would maybe/perhaps Monday suit you?'), the time expression gets some sort of vague interpretation. For this case, spoken English provides a similar way of expression: "Would it suit you, **maybe** on Monday?". Since this translation would require to much restructuring without much contribution to the communicative goal of the utterance, we prefer to remove the particle *vielleicht* in these cases, i.e., the translation "Would Monday suit you?" is produced.

```
(37) # (machen(L,I),arg3(L,I,I1),vorschlag(L2,I1);vorschlagen(L,I)),
       L=<H,dialog_act(request;suggest) -> true
     @del_scopal(vielleicht,L1,H).
```

```
(38) # abstr_passen(L,I),L=<H,dialog_act(request;suggest) -> true
     @del_scopal(vielleicht,L1,H).
```

```
(39) # support(L,I,L1),attitude(L2,I1),L2=<L1,L=<H,
       dialog_act(request;suggest) -> true
     @del_scopal(vielleicht,L3,H).
```

3.4 Temporal Referents

The translation of proper names for holidays is problematic if they are unknown in the culture of the addressee. Even if one has an idea about the origin of the holiday, this does not imply knowledge about its date or whether it is a public holiday. In order to avoid misunderstandings, we consider to translate particular holiday names in contexts as in (40) and (41) with the corresponding date or time span.

(40a) Am **Rosenmontag** habe ich keine Zeit.
(40b) On **February 10th** I won't be free.

(41a) Über **Fasching** bin ich im Urlaub.
(41b) From **February 10th** to **February 12th** I'll be on vacation.

For this kind of substitution, the transfer rule includes a request to the calendar model for the date (42) or the period (43) the holiday takes place. The values of the variables `Day`, `FromDay` and `ToDay` are the numbers that are used as designators of cardinal and ordinal numbers, and the values of the variables `Month`, `FromMonth` and `ToMonth` correspond to the designator values of the "month of year" predicate `mofy(L,I,D)` (see [Kasper et al., 1997]).

```
(42) get_date(+Inst,-Day,-Month)
(43) get_interval(+Inst,-FromDay,-FromMonth,-ToDay,-ToMonth)
```

(44) and (45) show the rules that cover the examples in (40) and (41). The first rule maps `rosenmontag` to the date on which it takes place in the considered year.[15] This substitution is restricted to the context that the holiday occurs as internal argument of a preposition with a temporal meaning (`temp_rel`). The rule in (45) maps `fasching` to the appropriate time period. The contextual restriction for this rule is hidden in the template definition.[16]

[15] holiday_date(X,I,D,M) := holiday(L,I,X) ↔ ord(L1,I,D),mofy(L,I,M) @add_group(L1,L).

[16] The holiday period has to appear as internal argument of a static temporal preposition (temp_loc) that is replaced by a temp_span_begin–temp_span_end construction, which is lexicalized by *from–to*.

holiday_interval(X,I,[D,M],[D1,M1]) := temp_loc(L,I1,I),holiday(L1,I,X) ↔
temp_span_begin(L,I1,I),temp_span_end(L5,I1,I2),ord(L2,I,D),mofy(L1,I,M),ord(L3,I2,D1),

(44) `# temp_rel(L1,I1,I),get_date(I,D,M) -> true`
`@holiday_date(rosenmontag,I,D,M).`

(45) `# get_interval(I,D,M,D1,M1)   -> true`
`@holiday_interval(fasching,I,[D,M],[D1,M1]).`

While the temporal referents for holidays can easily be obtained from the integrated calendar, the problem exemplified in (46) is harder to resolve.

(46) Dann müßten wir noch **einen Termin** finden.
(46a) We still have to find **a date** then.
(46b) We still have to find **a time** then.

The translation of the noun *Termin* in contexts like in (46) depends on the currently negotiated time interval, i.e., it shows, to some extent, anaphoric behaviour. The situation becomes clearer when the currently focused time interval is explicitly mentioned in the context as in (47) and (48).[17]

(47a) Dann müßten wir noch **einen Termin** *im Mai/in dieser Woche* finden.
(47b) Then we still have to find **a date** in May/this week.

(48a) Dann müßten wir noch **einen Termin** *am Montag* finden.
(48b) Then we still have to find **a time** on Monday.

To solve the translation problem in (46), we have to consider the process of time scheduling. It works on the principle of restricting step by step the time interval that suits both participants. If the meeting is scheduled from scratch, one usually starts with finding an appropriate month, then a week, followed by a date or a day of the week. Finally, the daytime and the concrete time are negotiated. Unfortunately, the recorded dialogues do not display this sequence of events in a regular way. Right at the beginning, one might propose a rather concrete time. For some events to be scheduled, the participants already have a fixed month or day. Thus, some of the expected steps can be skipped. However, the dialogue component that keeps track of the dialogue history can predict from the previously scheduled time interval the next one.

If (46) is uttered immediately following the participants' decision to schedule a meeting or after having fixed a month or a week, the translation in (46a) is preferred. After having agreed on a date one would expect that *Termin* refers to a daytime or an hour (46b). Here, the translation with *date* ought to be ruled out.[18]

mofy(L4,I2,M1),in_g(L4,G),in_g(L3,G) @add_group(L5,L) @add_group(L2,L1).

[17] However, for these cases the transfer does not require a special resolution, since the information that is necessary for disambiguation can be found in the local utterance context.

[18] One must admit that for (46a), in contrast to (47a), the translation with the noun *time* would also be feasible. Although in its prominent reading, it refers to a concrete time in terms of hours and minutes, it may also be used to denote an arbitrary time interval. We provide the underspecified noun *time* as a default translation.

To solve the translation problem that appears in (46), we have to know which time interval is currently under negotiation. This is expressed by the request in (49). As values we suggest `settle_date`, which covers the stages in which the month, the week and the day are settled, and `settle_time`, which indicates the stage in which daytimes and times are negotiated.

(49) `time_negotiation_stage(+Inst,+Stage)`

(50) and (51) show the corresponding transfer rules that integrate this call for evaluation.[19]

(50)
```
# time_negotiation_stage(I,find_date) <-> true
@noun(termin,date,L,I).
```

(51)
```
# time_negotiation_stage(I,find_time) <-> true
@noun(termin,time,L,I).
```

3.5 Temporal Perspective Points

In German, there seems to be a clear preference to use *nächst* and *kommend* to point to a time immediately following the speech time (52)–(53), while the use of *folgend* refers to a time that follows a future reference time (54).

(52a) ... vielleicht noch diese Woche oder **nächst**e Woche?
(52b) ... perhaps this or **next** week?

(53a) Wann würd's Ihnen denn passen? Ginge es **kommenden** Mittwoch?
(53b) When would it suit you? Would it suit you **next** Wednesday?

(54a) Vielleicht können wir gleich in der **folgenden** Woche das zweite Treffen machen.
(54b) Maybe we could have the second meeting right in the week **after**.

However, the Verb*mobil* dialogues do not exhibit a consistent usage of the German temporal-deictic adverbs. As shown in (55)–(57), *nächst* and *kommend* are also used to indicate a time following a temporal reference point in the future, and *folgend* to refer to a time immediately after the speech time. This leads to a translation problem. Depending on the speaker's temporal perspective point, the German adverbs have to be mapped onto other than their regular English equivalents, i.e., *nächst* and *kommend* might be translated into *following* and *after*, and *folgend* might correspond to *next*.

[19] These are, of course, not the only rules for the translation of *Termin*. They are applied after a set of other disambiguation rules.

(55a) Das wär' also der erste Termin, Samstag, siebter Mai. In der **nächsten** Woche ist ja ein Feiertag am Donnerstag.
(55b) So, the first date is Saturday, May seventh. Thursday the **following** week is a public holiday.

(56a) In der Woche vom zehnten an bin ich unterwegs und in der **kommenden** Woche kann ich erst ab Mittwoch.
(56b) In the week from the tenth on I'll be away, and in the **following** week I'll be free only from Wednesday on.

(57a) Heute haben wir Montag, den vierten, und ich würde vorschlagen, entweder gleich die **folgenden** fünf Tage oder ab Mittwoch, dem dreizehnten
(57b) Today is Monday the fourth and I would suggest the **next** five days or from Wednesday the thirteenth on.

In English, the use of the corresponding adverbs is more restricted than in German. For reference to a time immediately following the speech time, only *next* can be used, while *following* and *after* are used if a time is addressed from a future reference point. I.e., what we need for the transfer mapping is the information about the speaker's current temporal perspective when uttering *nächst*, *kommend* or *folgend*. This can be identified by consulting the dialogue history that keeps track of the introduced temporal referents. In the very beginning of a meeting scheduling circle, the speaker starts from the current time and refers with these adverbs to a day, week, month, etc., that immediately follows it. By every new proposal in the same circle, the speaker might either assume the current or a future reference point. This depends on the length of time interval focused by these adverbs. After having uttered: "Wie wär's am *nächsten* Freitag?" ('How about *next* Friday?'), in case of a negative response, the speaker may propose next: "Und wie sieht's *nächste* Woche aus?" ('How about *next* week?'). Here, the temporal perspective point for the longer time interval week remains the same. By proposing next a time interval of the same kind or a shorter one as before, the speaker's temporal perspective point changes to the last introduced time, which is in the future wrt. the speech time.

For accessing the information about the speaker's temporal perspective, the predicate in (58) is used. The values of `TempPerspect` are either `now` for reference to a time that immediately follows the speech time, or `fut` for reference to a time that succeeds a future reference point.

(58) `temp_perspect(+Label,+TempPerspect)`

(59) and (60) show the rules for the transfer of the temporal deictic adverbs.[20]

(59)
```
# temp_perspect(L,now) -> true
@mod(temp_deictic_adv,next,L,I).
```

[20] The used abstract predicates are defined as follows:
type(de,temp_deictic_adv,[naechst,kommend,folgend]).
type(en,abstr_following,[after,following]).

(60)
```
# temp_perspect(L,fut) -> true
@mod(temp_deictic_adv,abstr_following,L,I).
```

4 Transfer Requests to Discourse Semantics

Concerning discourse semantics, the transfer is first of all interested in the identification of antecedents of anaphors and ellipses. On the one hand, an antecedent itself might be required for a good translation (see (73)–(75) in Section 4.2). On the other hand, information about an antecedent might be necessary to verify the restriction part of a transfer rule. In case the context to be checked contains an anaphor, its antecedent has to be obtained to apply the transfer rule (see (62)–(61) in Section 4.1).

The call for identification of an antecedent is triggered by the transfer rule. The transfer automatically generates the corresponding request if an information required by the rule, such as the sort of a referential predicate, is lacking in the semantic input representation, or it finds a predicate of the kind `abstr_rel` or `abstr_nom` that mark verbal and nominal ellipses respectively.

4.1 Anaphora

In the Verb*mobil* domain we find a wide range of different kinds of anaphora. These are

- Personal pronouns
- Demonstrative pronouns
- Event-type pronouns
- Pronominal adverbs
- One-anaphora
- Definite descriptions

4.1.1 Personal Pronouns

Personal pronouns do not pose problems in our domain. About 85 % are first and second person pronouns. Thus, their reference to the speaker, to the hearer or to both of them can easily be identified. The only exception is *es* ('it'), that is ambiguous between a personal and an event-type pronoun (61). In a scenario with three participants we expect more problems since third person pronouns come into play, at least in German.

(61a)	A:	**Das Treffen** soll erst um drei beginnen.
	B:	Ich würde **es** *vorziehen.*
(61b)	A:	**The meeting** will start only at three o'clock.
	B:	I would *schedule* **it** *earlier.*

4.1.2 Demonstratives

Demonstratives are rather frequent in the Verb*mobil* domain. Fortunately, cases with extra-linguistic reference are rare and their reference can be determined by gender and number information (62). Only the demonstrative *das* is problematic, since it is also used as event-type pronoun (63).

(62a)	A:	Wir könnten es auch in **der ersten Maiwoche** machen.
	B:	Ja, **die** würde ich dann *vorziehen*.
(62b)	A:	We could also do it in **the first week of May**.
	B:	I would *prefer* **it**.

4.1.3 Event-type Pronouns

Event-type pronouns such as *es* and *das* are tricky. They often refer to complexly structured propositions. Then, no single referent can be picked out of the linguistic context as antecedent (63).

(63a)	A:	Wir könnten heute arbeiten und uns morgen ausruhen.
	B:	**Das** ist einen gute Idee.
(63b)	A:	We could work today and relax tomorrow.
	B:	**That**'s a good idea.

Let us give an example where anaphora resolution is essential for the application of transfer rules. In Section 2.5 we introduced the mapping rules for the ambiguous verb *vorziehen*. To apply these rules to the sentences in (61a) and (62a), the sorts of the antecedent of the pronoun *es* in (61a) and the demonstrative *die* in (62a) have to be identified. The transfer recognizes the resolution demand by the empty sort slot of the input representation and generates a request for the sort of the antecedent (64).

(64) `sort(+Inst,-Sort)`

4.1.4 Pronominal Adverbs

Pronominal adverbs are prepositions with a demonstrative as internal argument. By decomposing them into a preposition, a demonstrative and an unspecified nominal predicate (65), their translation is reverted to the translation of prepositions, which relies on the sort of the internal argument [Buschbeck-Wolf and Nübel, 1995]. Since in (66) it is anaphorically realized, the antecedent of `abstr_nom` has to recovered in order to apply the rules in (67)–(68).

(65) `vor(L1,I,I1),jen(L2,I1,G,H),abstr_nom(L3,I1),in_g(L3,G)`

(66) Mir wäre es lieber, wenn wir uns **davor** treffen.
(66a) I would prefer to meet you **before (it)**.
(66b) I would prefer to meet you **in front of it**.

(67)
```
# sort(I1,temporal) -> true
@prep(vor,temp_anterior,I,I1).
```

(68)
```
# sort(I1,entity) -> true
@prep(vor,loc_front,I,I1).
```

Another kind of adverbs that include a demonstrative meaning are deictic adverbs, such as *da* ('then, there'), *dort* ('there, then') and *hier* ('here'). In the Verb*mobil* dialogues, they are often used to refer to a time as well as to a location. Occasionally, this influences their translation (69).

(69) **Da** wär's mir recht.
(69a) It'll suit me **then**.
(69b) It'll suit me **there**.

To generate the appropriate English translation, the rules in (70)–(71) are applied. The conditions for these mappings are expressed by subsumption requests that trigger the resolution.

(70)
```
# concept(L)=<temporal <-> true
@mod(da_isect,then,L,I).
```

(71)
```
# concept(I)=<location <-> true
@mod(da_isect,there,L,I).
```

4.1.5 One-anaphora

One-anaphora always have to be resolved to generate the right gender of the German correspondence. (72a) contains two anaphoric *one* phrases. The first one corresponds to *date* and the second one to *meeting*. The German equivalents of these antecedents *Termin* (m,sg) and *Sitzung* (f,sg) determine the appropriate morphological form of *ein* and *weitere* in (72b).

(72a) Let us take that **date** for our first two-day **meeting**.
Now we still need **one** for **another one**.
(72b) Dann nehmen wir den **Termin** für unsere erste zweitägige **Sitzung**.
Jetzt brauchen wir noch **einen** für **eine weitere**.

4.1.6 Definite Descriptions

Definite descriptions as anaphora are relevant for the translation from Japanese into German. Japanese noun phrases do not have determiners that are essential for the translation into Germanic languages. The decision on whether an NP is definite or indefinite, can be made with its antecedent. If, in the preceding context, we can find a discourse referent that is anaphorically related to the NP in question, it has to be translated as a definite description. If no antecedent can be found, an indefinite description is more appropriate. However, there are exceptions to this simple heuristic. For a detailed analysis of the data, we refer to [Fujinami and Nanz, 1997].

4.2 Ellipsis

Concerning the language pair German–English, most ellipses can be preserved in the translation. However, sometimes the omitted material has to be reconstructed in order to provide a correct translation or to verify contextual restrictions of particular lexical items.

In the semantic representation, verbal ellipsis is represented by the underspecified predicate `abstr_rel` and noun ellipsis by `abstr_nom`. These predicates replace the antecedent and can be refined on demand.

4.2.1 V and VP Ellipsis

In German, modal verbs can be used with directional modifiers that are attached to the omitted verb. In English, the verb embedded by the modal cannot be left out. Thus, we need to reconstruct it for translation.

The insertion of the omitted verb is quite simple, if modals like *müssen* and *sollen* are used with locative directionals (73a). Here, an underspecified movement relation is required. Fortunately, English offers the verb *to go*, which fits this meaning (73b). The situation would be more difficult for languages that miss a verb with this general meaning. E.g., in German, one has to distinguish between *gehen* and *fahren*. For reconstruction we would have to know the kind of transportation.

(73a) Wir *müssen* _ nach Frankfurt.
(73b) We have **to go** to Frankfurt.

In (74)-(75), the insertion of the appropriate verb requires real inferences, since there are contextually different alternatives.

(74) Ich *möchte* _ zu Ihnen.
(74a) I want **to talk** to you.
(74b) I want **to see** you.

(75) Ich *möchte* _ zum WDR.
(75a) I want **to work** at the WDR radio station.
(75b) I want **to get** to the WDR radio station.

Consider the semantic representation for (75) in (76). The control verb *mögen* embeds a not further specified proposition `abstr_rel` via the hole `h1`. Its agent `arg1` is coindexed with the `arg2` of *mögen* that corresponds to its subject.

(76)
```
moegen(l1,i1),
arg2(l1,i1,i2),
arg3(l1,i1,h1),
abstr_rel(l2,i3),
arg1(l2,i3,i2),
zu(l3,i3,i4),
def(l5,i4,l8,h2),
named(l6,i4,wdr),
pron(l7,i2),
prontype(i2,std,sp),
dir(l3,yes),
in_g(l6,l8),
in_g(l1,l9),
in_g(l2,l10),
in_g(l3,l10),
leq(l10,h1)...
```

In order to translate the examples in (75), we have to refine the `abstr_rel` predicate. This is shown by the rules in (77)–(78). Here, `moegen` is translated into `want`[21] and `abstr_rel` is substituted by the verbal predicates `work` (77) and `get` (78) respectively. These kinds of refinements are only allowed in specific contexts. For (77) the embedded verb must take a directional preposition with an institution as internal argument. The concept of the underspecified predicate has to be subsumed by the concept of `arbeiten`. This should be verified by the discourse semantics.

(77)
```
# arg3(L,H),L1=<H,+is_prep(L2,I1,I2),dir(L2,yes),
  sort(I2,institution),concept(L1)=<arbeiten <-> true
@v_arg2TOv_arg1(moegen,want,L,I) @verb(abstr_rel,work,L1,I1).
```

(78)
```
# arg3(L,H),L1=<H,+is_prep(L2,I1,I2),dir(L2,yes),
  sort(I2,location),concept(L1)=<hinkommen <-> true
@arg2_arg1(moegen,want,L,I) @verb(abstr_rel,get,L1,I1).
```

Verbal ellipses have to be reconstructed if the transfer of a modifier depends on an omitted verb. Consider the examples in (79)–(80). For the translation of *where* and *here* into German it is decisive whether the verbal predicate denotes a movement or not.

(79a)	A:	I *am coming.*	B:	**Where, here**?
(79b)	A:	Ich komme.	B:	**Wohin, hierher**?
(80a)	A:	I*'ll be* at my office.	B:	**Where, here**?
(80b)	A:	Ich bin in meinem Büro.	B:	**Wo, hier**?

[21] `v_arg2TOv_arg1` is a template that performs an `arg2-arg1` shift.

The rules that relate *here* with *hierher* and *hier* are shown in (81)– (82). Since in utterances as (79)–(80) the sort of the modified verb is not available, a request to the discourse semantics for the sort of the antecedent is generated.

(81)
```
true <-> # sort(I,move_sit)
@mod(hierher,here,L,I).
```

(82)
```
true <-> # sort(I,~move_sit)
@mod(hier,here,L,I).
```

4.2.2 N Ellipsis

Noun ellipses display a similar behaviour as one-anaphora. In the Verb*mobil*domain we find the following types:

- Free quantifiers, e.g., *beide* ('both [of them]'), *einige* 'some (of them)', *drei* ('three [of them]')

  (83a) Bei mir würden **beide** _ passen.
  (83b) **Both of them** would suit me.

- Quantifiers/determiners with an adjective, e.g., *der nächste* ('the next [one]'), *alle anderen* ('all other [ones]'), *zwei ganze* ('two whole [ones]')

  (84) Ich würde **den anderen** _ *vorziehen*.
  (84a) I would *prefer* **the other one**.
  (84b) I would *schedule* **the other one** *earlier*.

- Quantifiers/determiners with an ordinal number, e.g., *das zweite* ('the second [one]'), *die beiden ersten* 9'the both first [ones]')

  (85a) **Die zwei ersten** _ haben mir mehr zugesagt.
  (85b) I liked **the two first ones** more.

N ellipses have often to be resolved for the verification of transfer conditions. E.g., the translation of *vorziehen* in (84) can only be decided wrt. its `arg3` realization (see the rules (13)–(19) in Section 2.5). Here, the resolution is triggered in the same way as in case of anaphorically realized arguments (see Section 4.1.3).

It is also desirable for generation to get an antecedent. This way, there is the option to choose between the anaphoric and nominal realization.

For the translation into Japanese, N ellipsis reconstruction is always necessary, since the lexicalization of quantifiers, ordinal and cardinal numbers depends on the type of the noun. For instance, cardinal numbers are added the suffix *ko* if small compact objects are counted, *dai* for small vehicles and *mai* for large ones, etc. For a detailed discussion see [Fujinami and Nanz, 1997].

4.2.3 Argument Dropping

In spontaneous speech, whole argument NPs might be omitted. Let's call these phenomenon "argument dropping". It occurs as:

- Topic drop (87)
- Implicit arguments (86)
- Zero anaphora

To get a well-formed translation, omitted arguments often have to be recovered. Sometimes, it suffices to insert a general pronoun that satisfies the verb's sortal restrictions on its argument, e.g., the event-type pronoun *it* in (86). Here an explicit noun is not required since one can assume the omitted argument to belong to the speaker's and hearer's common background.[22] However, for (87) no underspecified pronoun is available. To decide whether *he, she* or *it* is appropriate the previous context has to be considered.

(86a) Ich weiss _.
(86b) I know **it**.

(87) _ ist leider *nicht hier.*
(87a) Unfortunately **he/she** is *out of town.*
(87b) Unfortunately **it** is *not here.*

As we see in (87), the kind of omitted argument may also influence the translation of particular lexical items in its context. If *nicht hier sein* ('be not here') refers to a human being, in the Verb*mobil* domain, the translation into *be out of town* is preferred (87a). To apply the corresponding rule in (88) to the sentence in (87), we need the antecedent.

(88)
```
# sort(I,person),L=<H <-> true
@del_scop(neg,L1,H) @mod(hier,out_of_town,L,I).
```

Zero anaphora, i.e., omitted arguments, are frequent in spoken Japanese. The problem is that German does not allow that to the same degree. In most cases, the dropped arguments must not left out in the TL. For types of zero anaphora and resolution techniques, we refer to [Siegel and Metzing, 1994].

4.3 Resolution of Underspecified Scope

The resolution of underspecified scopal relations is necessary to verify restrictions on predicates that are bound in an underspecified way. To obtain the particular predicate, the corresponding label chain has to be instantiated.

[22] In this way, a lot of topic drops in copula constructions that include an experiencer PP can be tackled (see: _ ist okay für mich. '**It**'s okay with me').

There are some cases in which the scope of an operator is relevant for the translation, in particular for the surface order of the TL utterance. (89) shows the alternative scopes of the negation in case of head switching. For resolution, prosodic accent information might be of help.

(89)	Ich würde da lieber **nicht** kommen.
(89a)	I would prefer **not** to come.
(89b)	I would **not** prefer to come.

We must admit that, from a pragmatic point of view, the translations of (89) are equivalent, i.e., they would not be the reason for misunderstandings.

Another case is the translation of *immer* in contexts as in (90). Here, the translation differs wrt. the restrictor of the temporal quantificational adverb *immer* 'always'. The translation in (90a) assumes a quantification over all Mondays, while the one in (90b) reflects a quantification over all times of a particular Monday. For more examples, see [Lieske et al., 1997].

(90)	Am Montag kann ich **immer**.
(90a)	I'll be free **every Monday**.
(90b)	On Monday I'll be free **all the time/all day**.

5 Transfer Requests to Context Evaluation

In this section, we focus on lexical ambiguities whose resolution requires more information than is available from the local semantic context (see Section 2.5).

We regard two types of lexical ambiguities that force different translations:

- Ambiguities between different semantic classes (see Section 5.1)
- Ambiguities within a semantic class (see Section 5.2)

5.1 Resolution of Semantic Class Ambiguities

Beside scopal ambiguities, lexical-semantic ambiguities that require different semantic structures are represented in an underspecified way. Their representation is captured by semantic ambiguity classes. These are compact semantic representations that pack both feasible readings in one [Kasper et al., 1997]. If the transfer discovers the same lexical ambiguity in the TL it preserves the ambiguous representation. If not, it calls the context evaluation to instantiate one of the possible interpretations.

There is a couple of systematic lexical-semantic ambiguities that have to be resolved for German–English translation:

- Intersective vs. scopal adverbs
- Pronominal vs. scopal adverbs
- Quantificational vs. non-quantificational temporal adverbs

The call for specification of the underspecified lexical ambiguity `sem_class` is attached to the label of the ambiguous predicate (91). The values of the `SemClass` variable correspond to the semantic class and subclass abbreviations used in the semantic database [Kasper et al., 1997].

(91) `sem_class(+Label,+SemClass)`

In the following, we illustrate these ambiguities with some examples.

5.1.1 Intersective vs. Scopal Adverbs

(92)–(96) show some examples where the distinction between scopal adverbs (`scop`) and intersective modifiers (`isect`) is relevant for the translation of particular adverbs.[23]

Scopal adverbs are subdivided into focus particles (`fp`), modal sentence adverbs (`sa`) discourse relation adverbs (`dra`), modal particles (`mp`) and exclamatives (`excl`) [Bos and Schiehlen, 1997]. If the differentiation within the class of scopal adverbs is relevant for translation, the `sem_class` request is made more specific. Here, the classes of modal particles and sentence adverbs seem to be most interesting. They are often dropped in the translation, since English has an impoverished use of discourse particles and filler words (see Section 5.2.1).

(92a) Dann wird das Treffen **eben** nur vier Tage lang. (`scop/fp`)
(92b) The meeting will last only four days then.

(93a) Diesen Termin haben wir **eben** ausgeschlossen. (`isect`)
(93b) We have **just** excluded this date.

(94) Weil ich **selbst** nächste Woche nicht kommen kann. (`scop/fp` vs. `isect`)
(94a) Because I couldn't come **even** next week.
(94b) Because I **myself** couldn't come next week.

(95a) Wie sähe es in der Woche **direkt** nach Ostern aus? (`scop/fp`)
(95b) How about the week **right** after Easter?

(96a) Darf ich sie mal ganz **direkt** fragen? (`isect`)
(96b) May I ask you a **straight** question?

(97)–(98) give an example for the rules that are used to resolve the ambiguity between scopal adverbs and intersective modifiers. They include the `sem_class` request

[23] The adverbs *nun, gleich, erst* or *einfach* display a similar ambiguity.

introduced in (91). For the mapping to the refined predicate we make use of templates that instantiate its particular semantic representation.

(97)
```
# sem_class(L,isect) -> true
@isect_scopTOisect(eben_isect_scop,eben_isect,L,I).
```

(98)
```
# sem_class(L,scop) -> true
@isect_scopTOscop(eben_isect_scop,eben_scop,L,I).
```

5.1.2 Pronominal vs. Scopal Adverbs

The ambiguity between pronominal adverbs or anaphoric prepositions (`aprep` - a subclass of intersective modifiers (`isect`)), and discourse relational adverbs (`dra` - a subclass of scopal adverbs (`scop`)), has often to be resolved for German–English transfer (99)–(102).

(99a) **Dafür** fällt bei mir der 17. flach wegen einer Messe in Berlin. (`scop/dra`)
(99b) **But** I have got no time on the 17th **in turn** because of a fair in Berlin.

(100a) **Dafür** ist mein Zeitplan leider zu knapp bemessen. (`isect/aprep`)
(100b) Unfortunately, my time schedule is too tight **for that**.

(101a) **Dabei** hätte ich allerdings im September mehr Zeit. (`scop/dra`)
(101b) But I would have more time in September.

(102a) Lassen wir es dann **dabei**? (`isect/aprep`)
(102b) Do we leave it **like that**.

We exemplify the request for disambiguation by the rules for *dafür* in (103)–(104).

(103)
```
# sem_class(L,isect) -> true
@aprep_scopTOaprep(dafuer_aprep_scop,fuer,L,I).
```

(104)
```
# sem_class(L,scop) -> true
@aprep_scopTOscop(dafuer_aprep_scop,dafuer_scop,L,I).
```

5.1.3 Quantificational vs. Non-quantificational Temporal Adverbs

This ambiguity is inherent to a closed class of temporal adverbs, consisting of daytime adverbials (*morgens, mittags, nachmittags*, etc.) and week-day adverbials (*montags, dienstags, mittwochs* etc.).

(105) **Morgens** kann ich nicht teilnehmen. (`def` vs. `qua`)
(105a) I cannot participate **in the morning**.
(105b) I cannot participate **in the mornings/on every morning**.

The utterance in (105) means that the speaker cannot participate either on a specific morning (105a) or on every morning (105b).

The underspecified decomposed representation of the adverb *morgens* is shown in (106). It is represented as temporal modifier with the incorporated noun as internal argument. The ambiguity is captured by the underspecified quantifier `jed_def`. The appropriate reading has to be instantiated with the corresponding `sem_class` requests in (107)–(108).

(106)
```
temp_loc(L1,I1,I2),jed_def(L2,I2,L3,H),morgen(L3,I2)
```

(107)
```
# sem_class(L,def) -> true
@quant_chg(jed_def,def,L,I).
```

(108)
```
# sem_class(L,qua) -> true
@quant_chg(jed_def,qua,L,I).
```

5.2 Ambiguities within a Semantic Class

A lot of translational ambiguities remain within a semantic class. To a large extent, ambiguities of verbal, adjectival, adverbial and prepositional predicates can be handled in the transfer mappings by constraining them to the appropriate context (see Section 2.5). However, there are cases in which the immediate context is not sufficient. Consider the example in (109). For the modifier *eher*, in utterances that express an attitude to a time (109), two interpretations are feasible. They result in the translations in (109a) and (109b).

(109) Am Montag wäre es bei mir **eher** möglich.
(109a) Monday would suit me **better**.
(109b) It would suit me **earlier** on Monday.

The rules that map *eher* to the *better* and *earlier* are shown in (110) and (111), respectively. They include a subsumption test on the underlying concept that has to be verified by the context evaluation component.

(110)
```
# concept(L)=<eher_prefer <-> true
@mod(eher,good,L,I) @add_comp(I).
```

(111)
```
# concept(L)=<eher_temporal <-> true
@mod(eher,early,L,I) @add_comp(I).
```

In the following two sections, we focus on translational ambiguities in the domain of discourse particles and nouns.

5.2.1 Disambiguation of Discourse Particles

Let us consider some examples that show the different discourse functions of the particle *doch* [Stede and Schmitz, 1997].

(112a) **Doch**, ich komme am Montag.
(112b) **Yes**, I'll come on Monday.

(113a) Wir wollten uns **doch** am Montag treffen?
(113b) We wanted to meet on Monday, **didn't we?**

(114a) Dann würde das **DOCH** gehen.
(114b) Then, it would be possible, **after all**.

Sentence-initial *doch* in (112) is used to deny the utterance of another speaker and to reaffirm one's own previously expressed opposite opinion. In (113), *doch* signals the return to a previously made arrangement. The speaker reminds the hearer of a scheduled meeting and expects his approving response. With the prosodically marked *doch* in (114), the speaker refers to a previous dialogue stage. Something that was impossible before turned out to be feasible at the utterance time.

All three readings of *doch* in (112)–(114) need to be translated differently. While the reading in (112) can be recognized by its sentence initial occurrence (116), for (113) and (114) the transfer problem consists in the identification of the pragmatic function of *doch*. It is communicated by the predicate in (115). The values of `ParticleClass` correspond to the classes of pragmatic functions.

(115) `particle_class(+Label,+ParticleClass}`

[Stede and Schmitz, 1997] developed a classification of discourse particles wrt. their discourse function. They consider the particle *doch* to be either a coherence marker which should not be translated or a pointer to something previously uttered (`particle_class(L,given)`). In case the latter function is identified, the translation of *doch* seems to differ wrt. the sentence mood. In yes-no questions, it is expressed by a question tag in English. In declarative or imperative sentences, it is mapped onto *after all* if it bears prosodic accent. The transfer rules in (117)-(118) integrate in these kinds of restriction in their condition parts.

(116)
```
# sent_init(L) <-> true
@particle(doch,yes,L).
```

(117)
```
# particle_class(L,given),sent_mood(ynq) <-> true
@particle(doch,quest_tag,L).
```

(118)
```
# particle_class(L,given),sent_mood(decl;imp),pros_accent(L) <-> true
@particle(doch,after_all,L).
```

5.2.2 Concept Refinement for Nouns

In contrast to verbal and modifier predicates, the disambiguation of nouns is notoriously difficult, since it is impossible to manually fix all contexts in which the one or the other translation is preferred.[24] For this task, it seems to be more reasonable to rely on statistical information [Kameyama et al., 1993], [Gale et al., 1992] and [Almuallim et al., 1996]).

Consider the German noun *Gespräch*. It corresponds to *meeting* or *discussion* when it is a kind of organized event, while the other equivalent *conversation* denotes a more casual event or refers to its course. There are several contexts which force the one or the other interpretation. Co-occurring with verbs, such as *organize, plan* or *prepare*, *meeting* or *discussion* seem to be appropriate translations, while in other contexts *conversation* is used (see Table 1). Similarly, modifiers can identify one of the alternatives (see Table 2).

sich in ein *Gespräch* **einschalten**	**join** a *conversation*
ins *Gespräch* **kommen**	**get into** a *conversation*
ein *Gespräch* **fortsetzen**	**continue** a *conversation*
ein *Gespräch* **vorbereiten**	**prepare** a *meeting/discussion*
ein *Gespräch* **organisieren**	**organize** a *meeting/discussion*
ein *Gespräch* **planen**	**plan** a *meeting/discussion*

Table 1: Co-occurrences of *Gespräch* with verbal predicates

ein **stockendes** *Gespräch*	a **faltering** *conversation*
ein **nettes** *Gespräch*	a **nice** *conversation*
ein *Gespräch* **unter vier Augen**	a **private** *conversation*
ein **angesetztes** *Gespräch*	a **scheduled** *meeting/discussion*
ein **vertagtes** *Gespräch*	a **postponed** *meeting/discussion*
ein **dringendes** *Gespräch*	an **urgent** *meeting/discussion*

Table 2: Co-occurrences of *Gespräch* with modifiers

In the following, we will not distinguish between the different origins of noun ambiguities, such as SL inherent polysemy, specialization in the TL or underspecified relationality of relational nouns, since they do not seem to correspond to particular resolution procedures. We just list a couple of translational ambiguities that have to be resolved by the context evaluation component.

[24] Especially problematic is the resolution of translation mismatches [Kameyama et al., 1991], i.e., cases where the TL lexicalizes a conceptual distinction that is not made in the SL. Since the contexts in which these words occur are very similar it is hard to make a decision. For examples see (137)–(138).

Essen - Lunch/Dinner/Food

In German and English, we find different names for meals depending on the daytime they happen, e.g., *Mittagessen-lunch, Abendessen-dinner.* However, instead of the explicit compound, Germans often use only *Essen* (119a)–(120a), which leaves underspecified the daytime. This causes problems for its translation into English which does not provide a similar general expression. Here, we have to distinguish between *lunch* (119b) and *dinner* (120b). In order to choose the correct translation we have to know the time scheduled for the meal. It might be inferred from the utterance context. In (119a) the time is explicitly mentioned, in (120a) the utterance time provides a solution.

The third translation alternative for *Essen* is *food.* It is used if the concrete object is meant (121).

(119a) Wenn ich noch Zeit vor vierzehn Uhr zum **Essen** habe, kein Problem.
(119b) If I have time to have **lunch** before two p.m., no problem.

(120a) Bis gleich zum **Essen** beim Griechen.
(120b) See you for **dinner** at the Greek restaurant.

(121a) Das **Essen** ist dort immer außerordentlich gut.
(121b) The **food** is just excellent there.

(122)–(124) show the translation mappings of *Essen.* The contextual restrictions are formulated as requests on whether the ambiguous predicate subsumes one of the more special instances.[25].

(122)
```
# concept(L)=<mittagessen <-> true
@noun(essen,lunch,L,I).
```

(123)
```
# concept(L)=<abendessen <-> true
@noun(essen,dinner,L,I).
```

(124)
```
# concept(L)=<essen_konkret <-> true
@noun(essen,food,L,I).
```

Stelle - Place/Position

The noun *Stelle* is ambiguous already in German. It denotes either a particular area or position in space or a regularly paid position. These interpretations result in the translations in (125b) and (126b).

(125a) Was war noch mal die **Stelle**, wo wir uns letztes Mal getroffen haben?
(125b) What was the **place** there we met last time?

[25]For the sake of simplicity, in the following, we ignore the TL condition parts.

(126a) Ich habe eine gute **Stelle** gefunden.
(126b) I've found a good **position**.

(127)
```
#concept(L)=<platz_lok <-> true
@noun(stelle,place,L,I).
```

(128)
```
# concept(L)=<anstellung <-> true
@noun(stelle,position,L,I).
```

Geschäft - Business/Job/Office/Shop

The examples (129)–(132) display the different translations of *Geschäft*.

If *Geschäft* refers to buying and selling, commerce or trade, *business* is the correct translation (129). If it means a piece of work or a task, its English correspondence is *job* (130). Furthermore, *Geschäft* might denote a location (131)–(132). In this case, English lexicalizes the difference between rooms or buildings used as a working and business place (131b) and rooms or buildings where goods or services are sold to the public (132b).

(129a) Ich mache **Geschäfte** mit dieser Firma.
(129b) I'm doing **business** with this company.

(130a) Könnten sie dieses **Geschäft** für mich erledigen?
(130b) Could you do this **job** for me?

(131a) Wir können das Treffen bei mir im **Geschäft** machen.
(131b) We can meet at my **office**.

(132a) Die **Geschäfte** schließen ja erst um acht.
(132b) The **shops** are closing only at eight.

(133)
```
# concept(L)=<geschaeft_prozess1 <-> true
@noun(geschaeft,business,L,I).
```

(134)
```
# concept(L)=<geschaeft_prozess2 <-> true
@noun(geschaeft,job,L,I).
```

(135)
```
# concept(L)=<geschaeft_lokation1 <-> true
@noun(geschaeft,office,L,I).
```

(136)
```
# concept(L)=<geschaeft_lokation2 <-> true
@noun(geschaeft,shop,L,I).
```

Morning - Morgen/Vormittag

With *Morgen* and *Vormittag*, German lexicalizes a conceptual distinction that is not present in English (137)–(138). German speakers divide the *morning* into an earlier (137b) and a later part (138b). Whereas in (137a) and (138a) the context indicates a particular time interval, in most other cases, it does not. Then it is difficult to make a decision.

(137a) Fine, let us meet on Tuesday **morning** at nine for breakfast.
(137b) Gut, dann treffen wir uns Dienstag **Morgen** um neun zum Frühstück.

(138a) Tomorrow I am completely booked up in the **morning**, right until four p.m.
(138b) Morgen bin ich am **Vormittag**, also gerade bis vierzehn Uhr völlig ausgebucht.

(139)
```
# concept(L)=<morgen <-> true
@noun(morning,morgen,L,I).
```

(140)
```
# concept(L)=<vormittag <-> true
@noun(morning,vormittag,L,I).
```

Platz - Time/Slot/Place/Seat

The noun *Platz* denotes primarily a spatial area (143)–(144). However, in the Verb*mobil* domain, it is more frequently used to refer to a time interval (141)–(142). With temporal reference, it is translated into either *time* (141b) or *free slot* (142b). In its spatial interpretation *place* and *seat* occur as translation correspondences.

(141a) ...oder halt die Woche drauf, da ist auch noch **Platz**.
(141b) ...or the week after, there is still some **time** left.

(142a) Da ist noch jede Menge **Platz** bei mir im Terminkalender.
(142b) There are still a lot of **free slots** in my calendar.

(143a) Oder treffen wir uns an einem neutralen **Platz**.
(143b) Or shall we meet at a neutral **place**.

(144a) Nehmen sie hier bitte **Platz**.
(144b) Take this **seat** please.

(145)
```
# concept(L)=<zeit <-> true
@noun(platz,time,L,I).
```

(146)
```
# concept(L)=<luecke <-> true
@noun(platz,slot,L,I)@add_adx(free,I).
```

(147)
```
# concept(L)=<platz_lok <-> true
@noun(platz,place,L,I).
```

(148)
```
# concept(L)=<sitzplatz <-> true
@noun(platz,seat,L,I).
```

Raum - Room/Time/Area

Similar to *Platz*, *Raum* also occurs with a temporal interpretation which is reflected in its English translation *time* (150). However, the spatial reading is predominant in (149) and (151). The correspondence *room* is used for reference to a part of a building, and *area* for a spatial region with a geographical dimension.

(149a) Ich reserviere den **Raum** für drei Stunden.
(149b) I book the **room** for three hours.

(150a) ...um noch genügend **Raum** zu haben für Eventualitäten.
(150b) ...to have enough **time** if anything unexpected should arise.

(151a) Das ist im **Raum** München.
(151b) This is in the **area** of Munich.

(152)
```
# concept(L)=<zimmer <-> true
@noun(raum,room,L,I).
```

(153)
```
# concept(L)=<zeit <-> true
@noun(raum,time,L,I).
```

(154)
```
# concept(L)=<region <-> true
@noun(raum,area,L,I).
```

Reise - Trip/Journey

Finally, regard the example in (155) and (156). In English one has to choose between *trip* and *journey*, depending on the distance, duration and regularity of the travelling. While *trip* is a short one, *journey* indicates a long distance or a short one travelled regularly.

(155a) Wir wollten eine gemeinsame **Reise** planen.
(155b) We agreed to plan a joint **trip**.

(156a) ...und am fünften geht es dann los auf die große **Reise**.
(156b) ...and on the fifth we will start out on the great **journey**.

(157)
```
# concept(L)=<kurze_reise <-> true
@noun(reise,trip,L,I).
```

(158)
```
# concept(L)=<lange_reise <-> true
@noun(reise,journey,L,I).
```

6 Summary

In this report, we have presented a collection of ambiguities and translational problems the resolution of which goes beyond the inference potential of the transfer component. We have clustered them wrt. the components that might provide the corresponding information.

Concerning the *dialogue component*, transfer is interested in the propositional content and the illocution of utterances, in the dialogue stage and dialogue history as well as in time reference information.

The contribution of *discourse semantics*, we see in the resolution of anaphora, the reconstruction of ellipsis as well as in the instantiation of underspecified scope.

From the *context evaluation* we expect the resolution of problematic lexical ambiguities that result in different translations.

References

[Alexandersson et al., 1997a] J. Alexandersson, B. Buschbeck-Wolf, T. Fujinami, E. Maier, N. Reithinger, B. Schmitz, and M. Siegel. 1997a. Dialogue Acts in Verbmobil-2. Verb*mobil* Report 204.

[Alexandersson et al., 1997b] J. Alexandersson, N. Reithinger, and E. Maier. 1997b. Insights into the Dialogue Processing of VERBMOBIL. In *Proceedings of the 5th Conference on Applied Natural Language Processing*, Washington, D.C.

[Almuallim et al., 1996] Hussein Almuallim, Ysuhiro Akiba, Takefumi Yamazaki, Akio Yokoo, and Shigeo Kaneda. 1996. Two Methods for Learning ALT-J/E Translation Rules from Examples and a Semantic Hierarchy. In *Proceedings of the 15th Internatioanl Conference on Computational Linguistics, Kyoto, Japan*, pages 57–63.

[Bos and Schiehlen, 1997] J. Bos and M. Schiehlen. 1997. Klassifikation der deutschen Partikeln in Verbmobil. Verb*mobil* Memo, Universität des Saarlandes, Universität Stuttgart, July.

[Bos et al., 1996] J. Bos, B. Gambäck, C. Lieske, Y. Mori, M. Pinkal, and K. Worm. 1996. Compositional Semantics in Verbmobil. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING '96).*

[Bos, 1995] J. Bos. 1995. Predicate Logic Unplugged. In *Proceedings of the 10th Amsterdam Colloquium*, University of Amsterdam, Amsterdam, Holland.

[Buschbeck-Wolf and Dorna, 1997] B. Buschbeck-Wolf and M. Dorna. 1997. Using Hybrid Methods and Resources in Semantic-based Transfer. In *Proceedings of the 2nd International Conference "Recent Advances in Natural Language Processing" (RANLP'97)*, Tzigov Chark, Bulgaria, September.

[Buschbeck-Wolf and Nübel, 1995] B. Buschbeck-Wolf and R. Nübel. 1995. Die Behandlung übersetzungsambiger Präpositionen im Transfer des Verbmobil-Demonstrators. Verb*mobil* Report 87, IBM, Heidelberg and IAI, Saarbrücken, Germany.

[Buschbeck-Wolf and Tschernitschek, 1996] B. Buschbeck-Wolf and C. Tschernitschek. 1996. What You Always Wanted to Know about Semantic Transfer. Verb*mobil* Report 114, Institute for Logic and Linguistics, IBM Informationssysteme GmbH Heidelberg, Germany.

[Dorna and Emele, 1996a] M. Dorna and M. Emele. 1996a. Efficient Implementation of a Semantic-based Transfer Approach. In *Proceedings of the 12th European Conference on Artificial Intelligence (ECAI '96)*.

[Dorna and Emele, 1996b] M. Dorna and M. Emele. 1996b. Semantic-based Transfer. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING '96)*.

[Dorna, 1996] M. Dorna. 1996. The ADT-Package for the Verb*mobil* Interface Term. Verb*mobil* Report 104, IMS, Universität Stuttgart, Germany.

[Egg and Lebeth, 1995] M. Egg and K. Lebeth. 1995. Semantic underspecification and modifier attachment ambiguities. In J. Kilbury and R. Wiese, editors, *Integrative Ansätze in der Computerlinguistik. Beiträge zur 5. Fachtagung der Sektion Computerlinguistik der Deutschen Gesellschaft für Sprachwissenschaft (DGfS)*, pages 19–24, Düsseldorf, Germany.

[Fujinami and Nanz, 1997] T. Fujinami and C. Nanz. 1997. The 101 Translation Problems between Japanese and German/English. Verb*mobil* Report 199, IMS, Universität Stuttgart, Germany.

[Fujinami, 1997] T. Fujinami. 1997. Combining Situated Reasoning with Semantic Transfer Minimally. In *Proceedings of the Workshop on Spoken Language Translation*, Madrid, Spain.

[Gale et al., 1992] W.A. Gale, K.W. Church, and D. Yarowsky. 1992. Using Bilingual Materials to Develop Word Sense Disambiguation Methods. *Proceedings of the 4th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-92)*, Montreal.

[Gambäck and Bos, 1997] B. Gambäck and J. Bos. 1997. Semantic-Head Based Resolution of Scopal Ambiguities. Verb*mobil* Report 194, Universität des Saarlandes, Germany.

[Grice, 1975] H.P. Grice. 1975. Logic and Conversation. In P. Cole and J.L. Morgan, editors, *Syntax and Semantics*, volume 3, Speech Acts, pages 41–58. New York.

[Kameyama et al., 1991] M. Kameyama, R. Ochitani, and S. Peters. 1991. Resolving Translation Mismatches with Information Flow. *Proceedings of the ACL*, Berkeley.

[Kameyama et al., 1993] M. Kameyama, S. Peters, and H. Schütze. 1993. Combining Logic-Based and Corpus-Based Methods for Resolving Translation Mismatches. *Working Notes of the AAAI Spring Symposium Series*, Berkeley.

[Kasper et al., 1997] W. Kasper, J. Bos, M. Schiehlen, and C. Thielen. 1997. Definition of the Abstract Semantic Classes. Verb*mobil* Technical Report 61.

[Kilger and Finkler, 1995] A. Kilger and W. Finkler. 1995. Incremental Generation for Real–Time Applications. Research Report RR-95-11, DFKI GmbH, Saarbrücken, Germany.

[Lieske et al., 1997] C. Lieske, J. Bos, M. Emele, B. Gambäck, and C.J. Rupp. 1997. Giving Prosody a Meaning. In *Proceedings of Eurospeech'97.*

[Parsons, 1991] T. Parsons. 1991. *Events in the Semantics of English.* MIT Press, Cambridge, Mass.

[Quantz and Schmitz, 1994] J. Quantz and B. Schmitz. 1994. Knowledge-Based Disambiguation for Machine Translation. Verb*mobil* Report 44, Technische Universität Berlin.

[Ripplinger and Alexanderson, 1996] B. Ripplinger and J. Alexanderson. 1996. Disambiguation and Translation of German Particles in Verb*mobil*. Verb*mobil* Memo 70.

[Schiehlen, 1996] M. Schiehlen. 1996. Semantic Construction from Parse Forests. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING'96).*

[Siegel and Metzing, 1994] M. Siegel and D. Metzing. 1994. Nullpronomina und die Organisation von Wissensquellen für den Transfer Japanisch–Englisch. Verb*mobil* Memo 12.

[Stede and Schmitz, 1997] M. Stede and B. Schmitz. 1997. Discourse Particles and Routine Formulas in Spoken Language Translation. In *Proceedings of the ACL/EACL'97 Workshop on Spoken Language Translation*, Madrid.