

'Relaxed' Compositionality in Machine Translation

Doug Arnold (Essex)
Steven Krauwer (Utrecht)
Louis des Tombe (Utrecht)
Louisa Sadler (Essex)

Dept. of Language & Linguistics,
University of Essex,
Wivenhoe Park, Colchester, C04 3SQ, UK.

R.U.Utrecht,
Trans 14, 3512 JK, Utrecht,
Netherlands.

Abstract

An approach to translation is described that embodies certain principles about translation, in particular, the principle of 'compositionality', with the capacity for dealing with problematic/exceptional and apparently 'non-compositional' phenomena in such a way that the treatment of both 'regular' phenomena, and other problem cases, is not affected. The discussion focusses on the translation between a class of adverbs in Dutch (e.g. 'graag') and the corresponding complex sentential structures in English ('like to'). A detailed discussion of the phenomenon is included, including aspects that are not adequately treated here.

Introduction

In this paper we present a partial hypothesis about MT by describing a machine that embodies certain principles. For example, that translation is basically a 'linguistic' relation, best understood as a relation between 'argument structures', that it requires independent, and linguistically well motivated characterisations of source and target languages, that it is basically 'compositional', and 'symmetric' or 'reversible'.

The hypothesis is partial in at least two ways. First, we neglect many important aspects of MT, such as the need to involve 'real world' knowledge in choosing among linguistically possible translations, and the need to take account of linguistic units larger than the sentence. Second because the principles themselves are still somewhat informal and impressionistic.

We are particularly interested in the way in which one can combine an approach to MT which embodies relatively strong principles with the treatment of 'irregular', 'exceptional' or otherwise problematic phenomena. In this paper we describe a particular version of compositionality ('relaxed' compositionality) which is promising in this respect: it maintains the basic insight of compositionality (that the translation of a whole is some reasonably straightforward function of the parts taking account of the way they are combined together), but allows a straightforward account to be given of apparently 'non-compositional' phenomena. For exemplification, we discuss the well-known, but still problematic 'like-graag' cases.

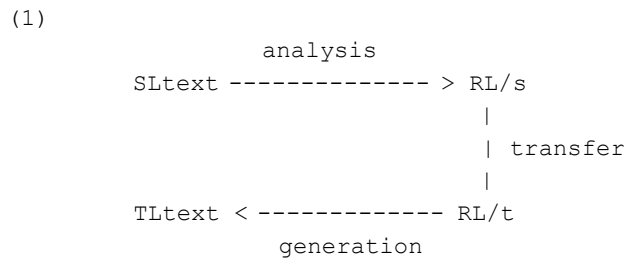
In itself, a treatment of a single problem case, or even a group of problem cases is not very interesting. It is always possible to find some account of one problem in isolation, but this is neither interesting nor practical, because it takes no account of the interaction of the problem with other problems, and with general principles.

The paper is in three sections. Section 1 describes the basic machine, and the underlying principles, including an idea of 'strict' compositionality. Section 2 introduces the idea of 'relaxed' compositionality in the context of a discussion of the problem of translating 'graag' and similar adverbs into English. The final section provides some more descriptive discussion of this problem, noting inadequacies of our treatment, and suggests a further extension to the machine.

1. The Core Machine

We describe here a transfer-based architecture for multilingual translation. In such a system, translation involves relating source language (SL) text to an abstract, but still in some way

source language oriented, representation language RL/s,
 'transferring' expressions of this abstract representation to
 expressions of a target language (TL) oriented representation
 RL/t, which is then related to the target language text.[1]:

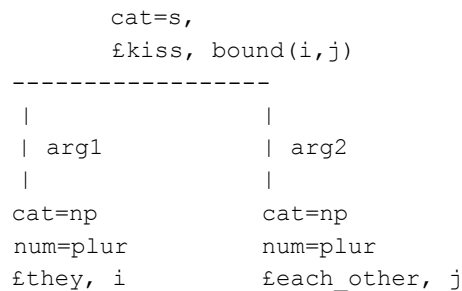


We call expressions in these representation languages Interface Structures (ISes), and take the basic idea to be one of semantic dependency or argument structure. Each construction is assumed to consist of a governor, (predicate, or head) with a number of dependents (arguments or modifiers). Dependents are said to fill 'slots' in the 'frame' of the head.

This makes ISes linguistic, rather than 'real world' knowledge representations. We think this is appropriate, since we think that translation is an irreducibly linguistic relation, and that the role of 'real world' knowledge is in fact to choose between 'linguistically possible' translations, e.g. to choose between alternative pairings of argument structures. However, we have no proposal for doing this at present, and the ideas about translation we present are correspondingly incomplete.

The following exemplifies:

- (2) They kissed each other
 (3)



(For exemplification, we have omitted attributes such as definiteness, tense/time reference, person, etc, and simplified in other small ways.) In a dependency framework, constructions are assumed always to have a lexical head, here indicated by '£', and marked on the root of the construction. Branches are labelled to indicate dependency relations to the head. Node labels are

feature bundles in the normal way. The anaphoric relation between 'they' and 'each other' is represented by means of the annotation 'bound(i,j)' mentioning the indices on the two nodes).

This approach decomposes the problem of translating texts into the problems of Analysis and Generation, and the problem of 'translating' abstract representations. Two problems arise:

First, consider the task of writing a Generation component for a language L in such a situation. In the worst case, the input to generation would be defined by the union of all other analysis components, and all transfer components that have L as their target. This seems to make the task impossible, since the writer of the monolingual component cannot realistically be expected to have that sort of knowledge of other monolingual analysis components. In the worst case, adding a new source language would lead to a radical redefinition of the generation component. There should be some independent, and linguistically well-motivated, definition of the class of ISes that are input to the generation component of L, some definition of the notion 'well-formed expression' in IS of L. Moreover, it is natural if the input to generation should be the same as (or at least very similar to) the output of analysis (cf Krauwer & Des Tombe (1984)).

Thus we are led to introduce a constructivist principle: each IS/i is given an independent definition by means of a 'grammar' (G), which generates IS/i in the conventional sense. We take the core of a G to consist of (i) declarations defining the notion 'well-formed feature bundle', and (ii) sets of 'constructors', which correspond to lexical frames or argument structures, specifying the requirements that a lexical head puts on its dependents.

```
(4)      fkiss = (cat=s).[arg1 = (cat=np),
                arg2 = (cat=np),
                *mod ]
```

For example (4) would be the (simplified) lexical entry for the verb 'kiss': 'kiss' has two obligatory arguments, both nps, and any number of modifiers, of any category at all [2]; the category of a construction headed by 'kiss' is s. A representation such as (2) can be generated by applying (3) to representations built by the constructors fthey, and feach_other, and filling in various features.

The second problem is finding a systematic basis for the translation relation (considered as an ISxIS relation). Here we think two ideas are useful. First, the idea that the translation relation is generally 'reversible' in the sense that 'expression2' is a translation of 'expression1' just in case 'expression1' is a

translation of 'expression2'. Second, the idea that the translation relation is basically 'compositional'. Intuitively, compositional translation of a simple, unanalysable SL expression involves finding the corresponding TL item. Compositional translation of a complex SL expression involves translating its parts, and combining the translations together in a way that reflects the way the parts were combined in the SL. Given the constructivist idea just described, this has a natural interpretation. For an SL expression E, constructed by applying a constructor CO to sub-expressions C1,...CN, then the translation of E can be found by translating C1,...CN, and applying the corresponding TL constructor to the translations:

(5) If E is an expression CO : C1,...,CN, then
 Trans(E) = Trans(CO) : Trans(C1),...,Trans(CN),

This would give the following general format for translation rules (t-rules):

(6)

```

sl_constructor: arguments
<=>
tl_constructor: arguments'
```

When constructors are taken to be lexical dependency frames, compositional translation involves simply relating those frames. For example, translating between English and French as in (7) a and b involves a rule like (8).

(7) a. John likes Mary.
 b. Marie plait a Jean.

(8)

```

£like(cat=s). [ 1!arg1,
                2!arg2
                3!*mod ]
<=>
£plaire(cat=s).[2!arg1,
                1!arg2,
                3!*mod ]
```

£like and £plaire are the names of English-IS and French-IS constructors, and arg1, arg2, and mod are slots in their frames. The rule indicates that the arg1 of 'like' corresponds to the arg2 of 'plaire', and vice-versa. One can think of one side of the rule as an instruction to decompose a source language structure, and the other side as an instruction to build a target language structure. For example, from English=>French, it could be read as: (i) take a structure built by the £like constructor, decompose it into 3 parts, whatever fills the arg1 slot, whatever

fills the arg2 slot, and whatever things fill the mod(ifier) slot; (ii) translate these parts; and (iii) recombine the translations by applying the fplaire constructor, in such a way that flike 's arg1 becomes fplaire 's arg2, and vice-versa.

Notice that this interpretation incorporates the constructivist principle: the target language side of a translation rule is essentially a call to the target language G to build a structure. So translation rules are guaranteed to produce only structures which are well-formed according to the independent definition of the target G.[3]

We think analysis and generation are simplified and made more coherent if further levels of representation are recognised. We assume the relation between these levels is handled by the same apparatus as is used for relating ISes (i.e. the idea of 'translation' is generalized slightly). In addition to IS, we currently assume a level of surface relational structure (broadly analogous to LFG f-structure (Kaplan and Bresnan 1982)), and a level of surface configurational structure.[4]

There are a number of extensions to the core machine as described above.

It often happens that a structure can be translated in several ways. In particular, it may be that it can be translated by a relaxed compositional rule (see section 2 below), and also translated strictly compositionally. Sometimes this reflects a genuine ambiguity. But in other cases it is undesirable, e.g. in the domain of morphology, Dutch 'kenner' should be translated as 'someone who knows', not as '*knower', which is what a normal compositional translation rule would give. For this reason, the basic machine is provided with a rudimentary control mechanism, which allows the user to state that if one rule applies other rules must be prevented from applying (they must be 'suppressed'). E.g. the 'kenner' \Leftrightarrow 'someone who knows' rule would suppress the strictly compositional translation rule.

Anaphoric relations (wh-trace, control, raising, pronominal) are expressed by means of coindexation between slots in a descriptor (and thus between nodes in a representation) (van Noord et al 1988). These are not handled by the ordinary translation or G apparatus. Instead, there are specialist coindexation rules which operate as part of the G. Stating an anaphoric relation involves stating a structural relation (for example, c-command) which holds between descriptors, and an anaphoric type, which indicates which features are affected by the relation (the transparency), and whether the relation is eg. one-one or one-many. (9) exemplifies the structural relation for English subject control:

9)
control(a,b) = (cat=s,control=subj).
 [a!arg1,(cat=s).
 [b!arg1= OPEN]]

The anaphor rules work by checking items that stand in given structural relations (e.g. a subject of a control verb, and an unfilled subject slot in an complement clause), and assigning an indication of binding, and by giving identical values to features that have been declared to be 'transparent' for the particular relation in anaphorically related nodes.

The basic idea is that anaphoric relations are preserved in translation, but that at any level, new annotations may be added, and existing ones may be checked for validity.

Given these extensions, the major problem with the view of translation given above is that many cases appear to be only partially compositional according to the definition. There are many cases where it is not plausible to analyse source and target structures as having the same kind of 'constructor-to-constructor' structure, or where such an analysis distorts and undermines the linguistic basis of the representation languages. Section 2 presents a case of this sort, discussing the relaxation of strict compositionality adopted to deal with such cases in a systematic way.

2. Relaxed Compositionality

The core translation machine described above makes two major claims with respect to the basis for translation:

- (1) that translation is basically compositional
- (2) that (1) is true of dependency representations

This section discusses a family of potentially problematic cases of 'non-compositional' translation, motivating an alternative 'relaxed' version of compositionality.

It is well known that languages differ greatly in their choice of strategy for the expression of sentential features such as habituality and modality. For example, Dutch and English differ with respect to the freedom given to sentential adverbials in the expression of these and other sentential features. In the examples below, what is a sentential modifier (an adverb) in Dutch is treated as the head of a verbal construction in English (in these cases, a control/raising verb). [5]

- (1) 'like-to':
- a. Jan kust Marie graag.
Jan kisses Marie 'likingly'
 - b. Jan likes to kiss Marie.
- (2) 'habitual used-to':
- a. In het weekend ging Jan gewoonlijk uit.
in the weekend went Jan habitually out
 - b. fit weekends, Jan used to go out.
- (3) 'formerly':
- a. Daar stond vroeger een molen.
there stood formerly a mill
 - b. There used to be a mill there.
- (4) 'perfective used-to':
- a. Wij woonden vroeger in Amsterdam.
we lived formerly in Amsterdam
 - b. We used to live in Amsterdam. [6]
- (5) 'happened-to':
- a. Jan is toevallig thuis.
Jan is accidentally at home
 - b. Jan happens to be at home.

We will assume that the t-rules must relate structures like (6) and (7).

- (6) $\text{\$kussen}$
- ```

| arg1 | | mod
| | |
\jan \marie \graag

```
- (7)  $\text{\$like, bound(i,j)}$
- ```

-----
| arg1           | arg2
\jan,i           & kiss
                -----
                | arg1   | arg2
                |        |
                OPEN,j   \marie
                (i.e. \jan)

```


In dependency terms, what we see in these cases is that English and Dutch 'invert' the relations between 'like/graag' and 'kiss/kussen': where Dutch applies the frame of £kussen to £graag, English applies the frame of £like to £kiss.

An informal, one directional, characterisation of what happens here is easy:

- (8)
- (a) the Dutch adverb translates as a control verb;
 - (b) the Dutch subject becomes subject of the control verb;
 - (c) the Dutch verb's tense becomes the tense of the control verb;
 - (d) the rest of Dutch construction (the verb and remaining features, arguments, and modifiers) translates as the complement of the control verb.

What is problematic is giving a precise characterisation, which is compatible with other general principles, and which does not distort or complicate the analysis of other phenomena. This relation can only be stated in a constructor-to-constructor system at the cost of introducing special constructors for the adverbs in question. But this is undesirable. For example, one would not like an analysis where 'like-to' is analysed as an adverb in English, because this would eliminate any possibility of a sensible and general account of phenomena as diverse as subject verb agreement, and the distribution of reflexive pronouns in the generation of English. Likewise, treating Dutch adverbs such as *graag* as two place predicates, on the basis of their translation into English, while other adverbs are treated as sentence modifiers, may be thought to distort the grammar of Dutch in an unwelcome manner. [7]

If we reject the possibility of special constructors which have the effect of making the IS representations similar in these problematic cases, we must relax the notion of compositionality to deal with these cases as described informally above. We can assume that the English IS G will contain entries like the following:

```
£like(cat=s, control=subj). [arg1=(cat=np),  
                             arg2=() ]
```

```
£kiss(cat=s). [arg1=(cat=np),  
              arg2=(cat=np),  
              mod*]
```

and the control rule given in section 1. Dutch IS will contain entries like:

```
fkussen(cat=s).[arg1=(cat=np),  
                arg2=(cat=np),  
                mod*]
```

```
fgraag(cat=adv).[]
```

As an alternative to dealing with this problem case 'vertically' by making the dependency representations similar, we might attempt to capture all of (8) above in one t-rule. This approach involves writing a (rather complicated) t-rule for each adverb/verb pair. Notice however, that as a general approach to such non-compositionalities, this solution is unworkable. This is because the 'rest of the Dutch construction' (see (8d) above) may itself contain problem cases. For example, several of these adverbs can co-occur, as in (9) and (10), so the treatment must be in some sense 'recursive'.

(9)

- a. Vroeger zwom Jan graag.
 formerly swam Jan likingly
- b. Jan used to like to swim.
- c. *Jan liked to used to swim.

(10)

- a. Jan zwom gewoonlijk graag.
 Jan swam habitually 'likingly'
 Jan used to like to swim.

An attempt to write a set of strictly compositional t-rules would lead to one rule being added for each combination of adverbs.

The solution we propose seeks to minimize the interaction between problematic cases and the 'normal' compositional cases. The basic idea is that it is often possible to split a problematic structure up into an 'exceptional' part, and a 'regular' part, in such a way that translating the 'regular' part is a strictly compositional process, and then to recombine the translations in some appropriate way 'licensed' by the target G. A first approximation is that if 'graag' is removed from the Dutch representation (6), and if there are no other problems, then the remainder can be translated compositionally. The translation of 'graag' and the remainder can then be recombined under the control of the target generator. The process for English to Dutch translation is analogous.

What this involves is 'relaxing' the notion of strict compositionality by allowing constructions to be split up into parts from which they could have been built, irrespective of whether they were actually built from those parts according to the source

language G. This implies that the translation is performed on the basis of the representation as it stands, that is, the actual method of construction of the representation does not affect the translation process.

Our treatment of the like/graag case therefore involves splitting the construction into three parts:

```

the np arg1 and the tense of v
the adverb
the verbal construction

```

The translation of each of these parts is stated by t-rule:

```

r1 = r! ((cat=s, !tense=X). [!arg1])
<=>
      r! ((cat=s, !tense=X). [!arg1])

```

```

r2 = r!((cat=s).[!mod=£graag])
<=>
      !£like((cat=s). [r!arg2])

```

```

r3= !kussen((cat=s).[r!*])
<=>
      !kiss((cat=s>). [r!*])

```

Rule 1 extracts the arg1 and specifies that its translation is arg1 of the translation of the rest of the construction, and extracts the tense and effectively copies it across (by inserting the feature tense with the same value ('X' is a variable) in the translation. Rule 2 extracts the adverb and translates it as a like construction in which the translation of the head of the construction containing graag satisfies the arg2 slot. The last rule extracts the dependents of the predicates, which will be translated separately, and states that 'kiss' translates as 'kussen'.

Given a subject control anaphora rule in the target generator along the lines of section 1 above, the following results:

```

(11)      £like
          -----
          | arg1          | arg2
          £jan,i          £kiss
                               -----
                               | arg1      | arg2
                               |          |
                               OPEN,j     £marie

```

For each adverb of this type, we add a t-rule along the lines of rule 2 above.

Cases involving more than one 'special' part (for example, the sentence with two adverbs in (10) above) are straightforward. The translator contains a rule for gewoonlijk/used-to such as:

```
r4 = r!((cat=s). [!mod=fgewoonlijk])
<=>
!fused-to((cat=s). [r!arg2])
```

The fact that used-to lacks non-finite forms will ensure that the target generator produces (12) rather than (13) (but see section 3 below):

- (12) Jan used to like to swim.
- (13) Jan liked to use to swim.

This approach to the like/graag problem has introduced a relaxation of compositionality which involves seeing the problematic example as made up of a compositional part and a 'special' part. Only the 'special' part has been treated exceptionally - the rest of the construction undergoes the normal t-rules. No modification which affects the intuitive basis of the Dutch grammar or the English grammar has been made. Furthermore, the interaction of non-compositionalities does not lead to further special rules.

The crucial difference between compositional and 'non-compositional' cases seems to be that the latter requires structures to be decomposed and built in ways that do not exactly mirror the analyses provided by the relevant Gs. However it turns out that the decomposition and recompositions are still rather limited, and 'natural' with respect to the source and target Grammars. For though the decomposition (construction) process does not always involve basic expressions or constructors of the source (or target) language, it always yields expressions that can be built from (into) basic expressions. Thus it has been possible to define a model which handles apparent non-compositional cases in such a way that the translations of expressions are still reasonably natural or straightforward functions of basic expressions, and their mode of combination (ie. compositionally).

3. Problems

The exemplification of adverb to verb translation in the previous section reflects the treatment that has been implemented in small demonstration system that has been built at Essex and Utrecht, but it is inadequate or incomplete in several important ways. We

first discuss some linguistic issues relating to our use of 'like/graag' to exemplify the notion of relaxed compositionality, and then go on to discuss a more serious inadequacy, making a proposal for a change to the core machine.

In section 2 we showed that a notion of relaxed compositionality would permit translation between a certain class of Dutch adverbs and English sentential constructions without modifying the Gs in any unintuitive way. The treatment had the advantage of being modular, in the sense that it permitted us both to treat only the 'special' part as 'special' in the t-rules, and to deal straightforwardly with the interaction of such phenomena.

However, the linguistics of the example were simplified in a number of respects. Firstly, though we have given an account of where tense is realised in the gov-modifier translations, we have assumed that the actual value of the tense feature is simply preserved. This is inadequate: tense and aspect systems in different languages vary greatly, and some representation based on time reference is clearly to be preferred (van Eynde 1987a, 1987b, Rigler 1988).

Secondly, we assumed that these adverbs and verbs have only one translation. However, there are alternative translations where the English and Dutch structures are much closer:

(1)
Jan houdt ervan te zwemmen.
Jan likes it-of to swim
Jan likes to swim.

(2)
Jan houdt van zwemmen.
Jan likes of swim(inf)
Jan likes swimming.

(1) and (2) exemplify the fact that English 'like to' is not necessarily translated by 'graag' in Dutch. The difference in surface form in (1) follows from the unavailability in Dutch of sentential complementation with verb-preposition complexes of this sort (houden strongly governs the preposition van), and is thus quite independent of the point at issue. A dummy argument *er* (variously analysed as a PP or a NP) is inserted, and the structure corresponding to the English sentential complement is a sentential adjunct. [83 (2) illustrates the use of a 'nominalised infinitive' in the translation of 'to swim' (recall that van, as a preposition, cannot take a sentential argument. [9]

The similarity that exists between the Dutch and English examples in (1) means that these structures can be translated compositionally, and given appropriate rules, producing these translations

and the alternatives discussed in section 2 is unproblematic.

The problem arises in choosing between the alternative translations that will then be produced, the method for choosing between alternative translations was implemented in an earlier version of this core machine, but it is still not clear how one can make the best choices in different contexts. [10]

A potentially more serious problem arises since the translation of 'graag' (or the German 'gern') as 'like to' is not always possible:

(3)

Jan was bang dat Marie het vervelend zou vinden
om af te wassen maar ze deed het graag.
John was afraid that Marie would find it tiresome to wash
up but she did it willingly/with pleasure.

(4)

Ich habe es gern gemacht.
I have it willingly done
I did it willingly.

The relevant distinction between cases such as (3) and (4) on the one hand, and those discussed in section 2 on the other, seems to involve the presence or absence of habituality in the time reference. The treatment exemplified here does not take account of this distinction, and is thus descriptively incomplete.

We have assumed the adverb-verb rules apply in English-Dutch transfer, taking this to be a relation between argument structures. But this is probably inadequate for 'graag' itself, which is '(surface) subject oriented', not 'arg1', or 'agent' oriented (it requires an animate subject).

There are several possibilities here. One would be to preserve an indication in the IS representation of which np is surface subject. This would look more convincing in the context of a general theory of the representation and treatment in translation of 'information structure' or 'functional sentence perspective'. A more immediately available alternative is to treat these adverbs in translation to/from a level where information about surface grammatical relations is already available, i.e. in Dutch analysis/generation, between surface syntactic relational structure and IS treating graag etc. as 2 place predicates at IS, i.e. very much like their English verb translations (so that transfer would become very simple). Only obvious modifications to the rules given above would be required. One would like to find other non-translational justification for introducing an IS representation of this type, given the view outlined in section 1

that Gs are autonomous, and motivated on monolingual grounds.
[11]

The treatment in section 2 assumed that English generation will filter any attempt to produce:

(5) *John liked to used to read books

because used (to) is morphologically defective, lacking any non-finite forms of the kind required in the complement of like (to). This is open to question. Some dialects allow:

(6) Did you used to read books?

where used (to) appears in a context otherwise reserved for a non-finite (base) form. A more adequate solution might involve looking for a semantic account of what is wrong with (5), or the analysis of to + verb as a special form of the verb – which used to certainly lacks. However, supposing no more adequate monolingual account of this could be given, a slightly less elegant account it still possible. If the rules that deal with the different problematic adverbs are ordered then, for example, ordering the *gewoonlijk* rule before the *graaq* rule will ensure that *gewoonlijk* is extracted first, and so given widest scope.

This is unattractive, since the linguist has explicitly to take account of the interaction of rules. But it does not seem disastrous, since the rules involved are intuitively closely related, and it is not unreasonable for them to be considered as a group.

Finally, we discuss an aspect of this analysis which raises some fundamental issues for the user of the system concerning the linguistic analyses (s)/he can express most naturally. On the basis of this discussion, we propose an extension to the system.

In order to get the *arg1* into the highest clause, so that it becomes subject of, for example, used to, it is necessary to extract and translate it before the other arguments of the predicate are translated. To see why this is a problem requires some background.

The framework of section 1 offers the linguist two basic strategies, which we will call frame to frame, and linking rule.

In the frame-frame strategy t-rules are essentially pairs of frames with annotations expressing correspondences between the various slots. For example:

```

(7)
      !flike(cat=v). [1!arg1,
                    2!arg2,
                    3!*mod ]
      <=>
      !fplaire(cat=v). [ 2!arg1,
                       1!arg2,
                       3!*mod ]

```

In this way it is easy to state idiosyncratic correspondences, but in the absence of a mechanism for stating redundancies across t-rules, there is no way of capturing the regularities that seem to exist. Conversely, the linking rule strategy involves using t-rules to state correspondences between predicates (govs), and separately between slots, for example:

```

(8)
!fkussen <=> !fkiss

(r! (cat=v). [!arg1])
<=>
(r! (cat=v). [!arg1])

```

This strategy is useful where there is little if any idiosyncratic variation in the roles way predicates assign (thus, e.g. it would be suitable for relating representation languages based on genuine semantic relations (theta roles), where there should be essentially no variation between predicates, for example, the hypothesis would be that agents always translate as agents.

We think it is an open, and very interesting question which of these strategies is appropriate, and under what conditions. For examples, the linking rule strategy seems to require some notion of default translation to operate successfully.

The problem with the existing treatment is that it excludes the frame-frame strategy generally: at least the arg1 (subject) *is* translated by a linking rule on the analysis in section 2. This is not without its attractions, for example, the idea that the syntactic behaviour of an arg1 (deep subject) or surface subject is somehow not conditioned by the predicate of the clause is a familiar one (subjects are not sub-categorised, because external to vp, for example). But given that there are idiosyncracies in correspondences between frames affecting the arg1 (subject) the approach leads to rules which combine exceptions, which is precisely what we set out to avoid. Moreover, the basic idea behind our treatment of semi-compositionality is to remove problematic parts of structures until what is left can be translated compositionally. This obviously motivates removing the special adverbs, but hardly motivates removing the subject.

What one would like is to extract the special adverbs and be left with the 'sentence nucleus' as in (3), the translation of which can be inserted as the complement of like (to) giving (10).

```
(3)          £kussen
           -----
           | arg1      |
           |            |
           £jan        £marie
```

```
(10)       £like
           -----
           | arg1      | arg2
           OPEN        £kiss
                               -----
                               | arg1  | arg2
                               |      |
                               £jan   £marie
```

The first problem with this is that it contains an empty pronominal c-commanding a lexical np, which will be discarded by English generation, as violating the anaphoric rules (the empty pronominal will remain unbound, and the representation will ultimately be discarded). Another facet of this same problem is that such a rule will not be reversible, because (10) will never be produced by English analysis.

Modifying the anaphoric rules of English is not an option given our discussion in section 1. above. What one would like is an operation which 'flips' or interchanges antecedents and anaphors. There is plenty of evidence that such a device is independently necessary to deal adequately with the analysis of, for example, raising, idioms and 'directional' processes like gapping. [12]

What distinguishes the adverb-verb cases from the others here is that it is not very general – rather than formulating some general rule to readjust structures, one would like to make this some sort of annotation on particular t-rules, although the extent to which such a process is generally required is a topic for further research. To do this, we will exploit an existing facility for indexing items in t-rules which is separate from the 'translation index' used in the examples above (it is used in the existing system it is used in a limited way for adding Linear Precedence and Anaphoric conditions to t-rules).

In Dutch-English what one would like is for translation to produce something like (10) from (9), then flip the nodes labelled OPEN and £jan, producing a configuration which the English control rule will handle correctly. English to Dutch is the reverse: the control coindexation is already in place, if the nodes are

flipped, then £jan will be represented as an argument of £kiss, and translation can replace £like and its empty argument with £graag. Thus, in Dutch-English we have: translate, flip, control; in English-Dutch we have control, flip, translate.

The rule that one would like to annotate might be something like (11), this mentions the arg1 of the Dutch verb as 'context' (it is not marked with a '!' which would extract it and cause it to be translated), and inserts/extracts an empty arg1 for £like. A 'flip' annotation can refer directly to the arg1 of £like, but to can only refer to the node that will contain £Jan as 'the translation of the arg1 in Dutch'. For this we introduce the notation tran(I), where I is a (non-translation) index. Thus we can write:

```
r!(cat=s). [ A-arg1, !&graag]
<=>
!£like(cat=s). [B-!arg1=OPEN,  r!arg2]
           {flip(B,tran(A1))}
```

This looks satisfactory for Dutch-English. Notice that for English-Dutch the 'flip' annotation must be applied before the body of the t-rule is executed, and that tran(I) must now mean 'the object that translates as I'. Thus though the rule remains 'reversible', the way it is written is clearly 'oriented'.

Footnotes

[1] The work reported here has been carried out in the context of the Eurotra machine translation project. It develops and modifies the ideas presented in Arnold et al 1986, Arnold & des Tombe 1987. We should emphasise, however, that it is not part of the mainstream Eurotra research or development work, and in particular, that the system we describe differs in many critical ways to the 'official' Eurotra software.

The system we describe has been implemented in c-prolog, and several small-medium scale modules have been written for fragments of Dutch, German and English. It is still experimental, however, and the work we report here is very much 'work in progress'. The design of the system, and the basic ideas about its application to linguistic description and translation are the result of collaborative work with: Gertjan van Noord, Joke Dorepaal and Coby Verkuyl (of the University of Utrecht), Dominique Petitpierre (ISSCO, University of Geneva), and Andrew Betts (Essex). Dominique Petitpierre and Gertjan van Noord wrote the great majority of the prolog code which the system uses.

[2] There should not be a single 'modifier' slot at IS: distinctions between, e.g. time, place, manner modifiers should be

recognised. Nothing hangs on this here.

[3] This idea of compositionality amounts to a very strong version of the 'rule-to-rule' hypothesis, and corresponds roughly to that used in Rosetta (Landsbergen 1386). It is stronger than the normal rule-to-rule hypothesis, which (e.g. in Montague's PTQ) pairs syntactic rules with arbitrary expressions of Intensional Logic (rather than pairing them with formation rules of Intensional Logic). This is not the place for a proper comparison of Rosetta and the system described here, but it may be useful to note four fundamental properties which set this system apart: (i) the use of abstract ISeS distinct from the surface grammar of the languages; (ii) the centrality of the idea of 'dependency'; (iii) the emphasis on the internal linguistic coherence of representation languages (Rosetta explicitly seeks to 'tune' source and target grammars, at the expense of internal naturalness, if necessary); (iv) the idea of a relaxed version of compositionality (see below).

[4] Since this is not a dependency level, the normal apparatus of constructors corresponding to lexical frames is not directly applicable. Instead, we take constructor to correspond to phrase structure rule at this level; its 'slots' will be the branches that connect the mother to the daughters, and apparatus for indicating linear precedence (LP) between branches is added.

[5] A similar problem also arises in German/English or German/French translation, for example:

Ich schwimme gern.
I swim 'likingly'
I like to swim.

J'aime nager.
I-like swim
I like to swim.

[6] This also occurs with *vroeger* with a perfective verb form:

Wij hebben vroeger in Amsterdam gewoond.
We have formerly in Amsterdam lived
We used to live in Amsterdam.

[7] But this is the treatment adopted in Rosetta and argued for in Landsbergen 1985, and Appelo and de Jong 1987, where it is pointed out that the impossibility of weather-it in subject position provides some monolingual justification for treating *graag* as a two place predicate.

[8] The same constraint is seen in

Jan hoopt erop dat Marie het schilderij ziet.
Jan hopes it-on that Marie the painting sees
Jan hopes that Marie sees the painting.

From its status as an adjunct follows the fact that you cannot extract a wh-item from the embedded sentential construction: extraction in Dutch, as in English, is ungrammatical from positions within adjuncts.

[9] We do not consider here translations for 'used to' involving 'plegen' ('be accustomed to'), which we take to be extremely formal if not obsolete:

In het weekend placht Jan uit te gaan.
in the weekend used Jan out to go
Jan used to go out at the weekend.

[10] For a description of the preference mechanism referred to here, see Petitpierre et al (1986).

[11] Semantically, this analysis of subject oriented adverbs is rather plausible (e.g. this is the sort of semantic representation proposed by Cresswell (1985) for the English adverb willingly, which is analogous), (see also the references in [7] above), but the incorporation of such 'deep' semantic treatments into a level of representation should not be adopted piecemeal.

[12] For example, Dutch and English gap in different directions.

References

D.J.Arnold, S.Krauwer, M. Rosner, L. des Tombe, & G.B.Varile (1986) "The <C, A>, T framework in Eurotra" in Proceedings of Coling 86 , Bonn.

Arnold,D.J. & des Tombe,L. (1987) "Basic Theory and Methodology in Eurotra" in Nirenburg, S. (ed.) pp 114-135.

Appelo, L. & F. de Jong (1987) "Synonymy and Translation", Philips Research M.S. 14.269.

Bresnan,J. (ed) (1982) The Mental Representation of Grammatical Relations MIT Press, Cambridge, Mass.

Cresswell, M.J. (1985): Adverbial Modification Reidel Publishing Company, Dordrecht.

van Eynde, Frank (1987a): "Time. A Unified Theory of Tense, Aspect and Aktionsart" (ms.), (Eurotra-Leuven).

van Eynde, Frank (1987b): "Tense in Text" (ms.), (Eurotra-Leuven).

- Kaplan, R. & J. Bresnan (1982) "Lexical Functional Grammar: a formal system for grammatical representation" in Bresnan (ed) pp173-281.
- Krauwert, S. & Des Tombe, L. (1984) "Transfer in a multi-lingual MT system" in Proceedings of COLING 1984 Stanford, Ca. pp464-487.
- Landsbergen, J. (1985) "Isomorphic Grammars and their use in the Rosetta Translation System", in King, M (ed) Machine Translation Today Edinburgh University Press.
- Landsbergen, J. (1986) "Montague Grammar and Machine Translation", in Whitelock et al (1987) pp 113-148.
- Nagao, M. (1987) "Role of Structural Transformation in a Machine Translation System", in Nirenburg, S. (ed.) pp262-277.
- Nirenburg, S. (ed.) (1987) Machine Translation: Theoretical and Methodological Issues CUP, Cambridge.
- van Noord, G. et al (1988) "A Treatment of Anaphora", ms. University of Utrecht.
- Petitpierre, D. et al (1986) "A model for Preference" in Proceedings of the 1986 European ACL , Copenhagen.
- Rigler, E. (1988) "Multilingual Application of a Theory of Tense and Aspect", Working Papers in Language Processing No. 4., Eurotra-Essex, University of Essex.
- Whitelock, P. et al (1987) Linguistic Theory and Computer Applications, Academic Press, London.