Syntactic and Semantic Transfer with F-Structures*

Michael Dorna*, Anette Frank†, Josef van Genabith‡ and Martin C. Emele*

*IMS, Universität Stuttgart	†Xerox Research Centre Europe	‡Dublin City University
Azenbergstr. 12	6, chemin de Maupertuis	Computer Applications
D–70174 Stuttgart	F–38240 Meylan	Dublin 9, Ireland
{dorna,emele}@ims.uni-stuttgart.de	Anette.Frank@xrce.xerox.com	josef@compapp.dcu.ie

Abstract

We present two approaches for syntactic and semantic transfer based on LFG f-structures and compare the results with existing co-description and restriction operator based approaches, focusing on aspects of ambiguity preserving transfer, complex cases of syntactic structural mismatches as well as on modularity and reusability. The two transfer approaches are interfaced with an existing, implemented transfer component (**Verbmobil**), by translating f-structures into a term language, and by interfacing f-structure representations with an existing semantic based transfer approach, respectively.

1 Introduction

Target and source levels of representation in transfer-based machine translation (MT) are subject to often competing demands: on the one hand, they need to abstract away from particulars of language specific surface realization to ensure that transfer is as simple and straightforward as possible. On the other hand, they need to encode sufficiently fine-grained information to steer transfer. Furthermore, target and source representations should be linguistically well established and motivated levels of representation. Finally, from a computational perspective they need to be sensible representations for both parsing and generation. LFG f-structures are abstract, "high-level" *syntactic* representations which go some way towards meeting these often irreconcilable requirements.

Correspondence-based transfer on f-structures has been proposed in (Kaplan et al., 1989). A closer look at translation problems involving structural mismatches between languages – in particular head switching phenomena (Sadler and Thompson, 1991) – led to the contention that transfer is facilitated at the level of semantic representation, where structural differences between languages are often neutralized. Structural misalignment is treated in semantics construction involving a restriction operator (Kaplan and Wedekind, 1993) where f-structures are related to (possibly sets of) disambiguated semantic representations.

Given the high potential of semantic ambiguities, the advantage of defining transfer on semantic representations could well be counterbalanced by the overhead generated by multiple disambiguated structures as input to transfer. This and the observation that many semantic (and syntactic) ambiguities can be preserved when translating into a target language that is ambiguous in similar ways, sheds light on the issue of the properties of representations for the task of defining transfer.

In principle, the problem of semantic ambiguity in transfer can be tackled in a number of ways. Packed ambiguity representation techniques (Maxwell III and Kaplan, 1993) could be integrated with the approach in (Kaplan and Wedekind, 1993). In the linear logic based semantics of (Dalrymple et al., 1996) scope ambiguities are accounted for in terms of alternative derivations of meaning assignments from a set of meaning constructors. Ambiguity preserving semantic transfer can be devised on sets of meaning constructors rather than disambiguated meanings (Genabith et al., 1998). Transfer on packed representations is considered

in (Emele and Dorna, 1998).

In the present paper we consider alternative approaches to transfer on underspecified – syntactic or semantic – representations, focusing on issues of modularity, reusability and practicality, interfacing existing implemented approaches in a flexible way. At the same time, the proposals readdress the issue of what is an appropriate level of representation for translation, in view of the known problems engendered by structural mismatches and semantic ambiguity.

We first show how the underlying machinery of the semantic-based transfer approach developed in Dorna and Emele (1996b) can be ported to *syntactic* f-structure representations. Second, we show how the underspecified *semantic* interpretation approach developed in Genabith and Crouch (1997) can be exploited to interface f-structure representations directly with the named semantic-based transfer approach. Third, we compare the two approaches with each other, and with co-description and restriction operator based approaches.

2   Syntactic Transfer

This section presents a simple bidirectional translation between LFG f-structures and term representations which serve as input to and output of a transfer component developed within the Verb*mobil* project (Dorna and Emele, 1996a). The term representation is inspired by earlier work (Kay et al., 1994; Caspari and Schmid, 1994) which uses terms as a quasi-semantic representation for transfer and generation.

The translation between f-structures and terms is based on the correspondence between directed graphs representing f-structures and the functional interpretation of these graphs (cf. (Johnson, 1991)). Given an arc labeled f which connects two nodes n_1 and n_2 in a graph, the same can be expressed by a function $f(n_1) = n_2$. An f-structure is the set of such feature equations describing the associated graph. Instead of feature equations $f(n_1) = n_2$ we use the relational notation $f(n_1, n_2)$.

Using this idea f-structures can be converted into sets of terms and vice versa.[1] F-structure

---

[1]For motivation why we prefer term representations

PRED features and their "semantic form" values are given special treatment. Instead of introducing PRED terms we build unary relations with the semantic form predicate name as functor (see Example (1)). The resulting representation is similar to a Neo-Davidsonian style event semantics (Parsons, 1991) but uses syntactic roles. For a formalization of the f-structure–term correspondence see Appendix A.

(1) a. $\begin{bmatrix} \text{SUBJ} & [\text{PRED} & \text{HANS}]\boxed{2} \\ \text{PRED} & \text{KOCHEN}\langle\uparrow \text{SUBJ}\rangle \\ \text{ADJN} & \{[\text{PRED} & \text{GERNE}]\boxed{3}\} \end{bmatrix}\boxed{1}$

  b. *Hans kocht gerne*

  c. { kochen(n1),
        SUBJ(n1,n2), Hans(n2),
        ADJN(n1,n3), gerne(n3) }

Consider the simple head switching example involving the German attitude adverb *gerne* and the English verb *like* (see (1b) and (3b)). (1a) is the LFG f-structure for the German sentence (1b).[2] (1c) is the set of terms representing (1a).

Transfer works on source language (SL) and target language (TL) sets of terms representing predicates, roles, etc. like the ones shown in (1c). The mapping is encoded in transfer rules as in (2). For a rule to be applied, the set on the SL side must be a matching subset of the SL input set. If this is the case, we remove the covering set from the input and add the set on the other side of the rule to the TL output. Transfer is complete, if the SL set is empty.

(2) a. { kochen(E) } <-> { cook(E) }.

  b. { SUBJ(E,X) } <-> { SUBJ(E,X) }.

  c. { Hans(X) } <-> { Hans(X) }.

  d. { ADJN(E,X),gerne(X) } # { SUBJ(E,Y) }
       <-> { like(X),XCOMP(X,E),SUBJ(X,Y) }.

The transfer operator <-> is bidirectional. Upper case letters in argument positions are logical variables which will be bound to nodes at runtime. Because of the variable sharings on both sides of a rule we work on the same nodes of a graph. The result is a graph rewriting process.

---

over feature structures for transfer, see (Emele and Dorna, 1998).

[2]For presentational purposes we leave out morphosyntactic information in f-structures here and in the following examples.

342

The head switching rule (2d) shows two components on its lefthand side: the part to the right of # is a test on a copy of the original input. The test binds the variable Y at runtime when applying the rule from left to right. In the reverse direction (and in general), TL tests are ignored.

Applying the rule set in (2) to (1c), we get (3c). We now use the correspondence between f-structures and term representations to construct the TL f-structure. The result is (3a) representing the English sentence (3b).

$$(3)\quad a. \quad \begin{bmatrix} \text{SUBJ} & \begin{bmatrix}\text{PRED} & \text{Hans}\end{bmatrix}_{[2]} \\ \text{PRED} & \text{like}\langle\uparrow \text{SUBJ},\uparrow \text{XCOMP}\rangle \\ \text{XCOMP} & \begin{bmatrix}\text{SUBJ} & \begin{bmatrix}\text{PRED} & \text{Hans}\end{bmatrix}_{[2]} \\ \text{PRED} & \text{cook}\langle\uparrow \text{SUBJ}\rangle \end{bmatrix}_{[1]} \end{bmatrix}_{[3]}$$

b. *Hans likes cooking*

```
c. { like(n3)
      SUBJ(n3,n2), Hans(n2),
      XCOMP(n3,n1), cook(n1),
                    SUBJ(n1,n2) }
```
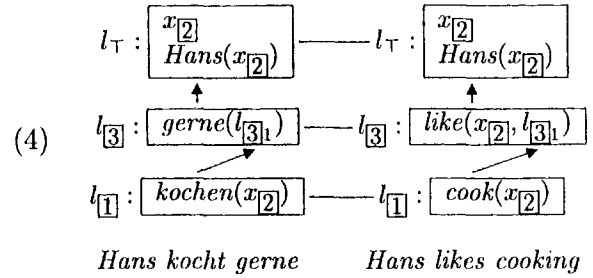
3 Semantic Transfer

Semantic-based transfer as detailed in (Dorna and Emele, 1996a; Dorna and Emele, 1996b) is based on rewriting underspecified *semantic* representations. The representations (Bos et al., 1996) are UDRS variants (Reyle, 1993). F-structures are abstract *syntactic* representations. They do, however, encode basic predicate–argument relations, and this is essentially semantic information. It turns out that there are important structural similarities between f-structures and UDRSs: f-structures can be "read" as UDRSs and hence be assigned an underspecified truth-conditional interpretation (Genabith and Crouch, 1997).[3] Appendix B gives a relational formulation of the correspondence between f-structures and UDRSs. The UDRS representations are processed by semantic-based transfer. The resulting system is bi-directional. Consider again the simple head switching case discussed in (1) and (3) above. (4) shows the corresponding UDRSs.

The structural mismatch between the two f-structures has disappeared on the level of UDRS representations and transfer is facilitated.[4]

---

[3]A similar corespondence between f-structures and QLFs (Alshawi and Crouch, 1992) has been shown in (Genabith and Crouch, 1996).

[4]In the implementation, a Neo-Davidsonian style en-

$$(4)\quad \begin{array}{ccc} l_T : \boxed{\begin{array}{c} x_{[2]} \\ Hans(x_{[2]}) \end{array}} & \longrightarrow & l_T : \boxed{\begin{array}{c} x_{[2]} \\ Hans(x_{[2]}) \end{array}} \\[4pt] l_{[3]} : \boxed{gerne(l_{[3]1})} & l_{[3]} : \boxed{like(x_{[2]}, l_{[3]1})} \\[4pt] l_{[1]} : \boxed{kochen(x_{[2]})} & \longrightarrow & l_{[1]} : \boxed{cook(x_{[2]})} \end{array}$$

Hans kocht gerne   *Hans likes cooking*

4 Embedded Head Switching and Multiple Adjuncts

How do the two approaches fare with embedded head switching and multiple adjuncts? Due to space limits we will not discuss straightforward cases where ambiguites represented in underspecified representations are carried over into the target language. Examples of this type involve quantificational and plural NPs, negation, or adjunct sets. Instead, we concentrate on complex cases where a source language ambiguity needs to be resolved in target language.

4.1 Embedded Head-Switching

The syntactic transfer rules (2) are supplemented by (5). The complex rule for *gerne* in (5) overrides[5] (2d) and the COMP rule in (5). For each additional level of embedding triggered by head switching adjuncts a special rule is needed.

```
(5)  { vermuten(E) } <-> { suspect(E) }.
     { Ede(X) } <-> { Ede(X) }.
     { COMP(E,X) } <-> { COMP(E,X) }.
     { gerne(X),ADJN(E,X),COMP(E1,E) } #
     { SUBJ(E,Y) } <->
     { like(X),XCOMP(X,E),SUBJ(X,Y),COMP(E1,X) }.
```

By contrast, on the level of UDRSs head switching has disappeared and transfer is facilitated. Figure 1 shows the transfer correspondence between terms and UDRSs.

---

coding of predicate argument relations is used. The subject of the target `like` relation is determined by the following transfer rule:

```
{ L:gerne(L1) } # { L2 ≤ L1, L2:agent(A) }
<-> { L:like(A,L1) }.
```

≤̇ is the transitive closure over subordination constraints ≤. Here and in the following we do not give set representations of UDRSs and transfer rules. Instead, we provide a graphical representations of standard UDRSs to better illustrate the structural mismatches discussion.

[5]For the treatment of overriding see, e.g., the specificity criterion in (Dorna and Emele, 1996a).
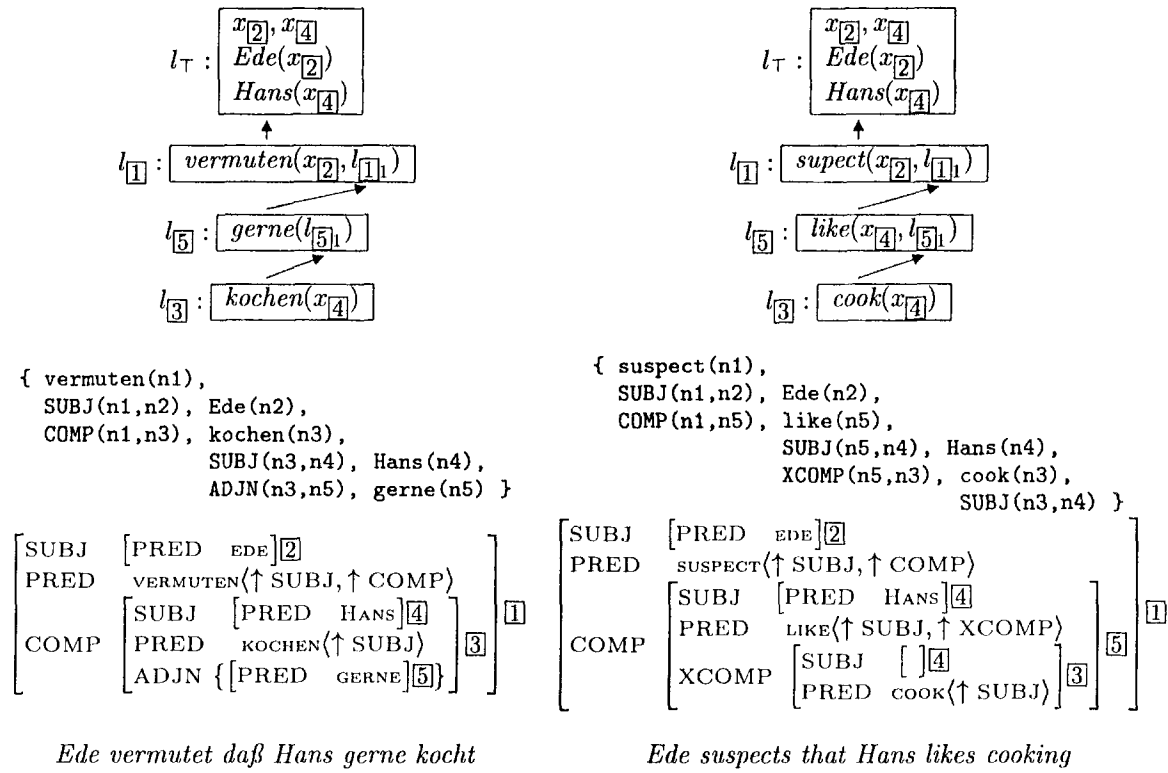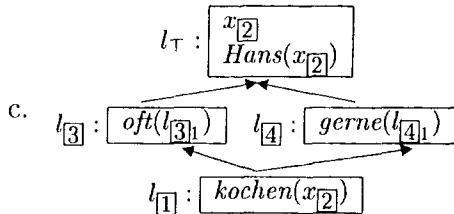
Left diagram:

$$l_\top : \boxed{\begin{array}{l} x_{[2]}, x_{[4]} \\ Ede(x_{[2]}) \\ Hans(x_{[4]}) \end{array}}$$

$$l_{[1]} : \boxed{vermuten(x_{[2]}, l_{[1]1})}$$

$$l_{[5]} : \boxed{gerne(l_{[5]1})}$$

$$l_{[3]} : \boxed{kochen(x_{[4]})}$$

```
{ vermuten(n1),
  SUBJ(n1,n2), Ede(n2),
  COMP(n1,n3), kochen(n3),
          SUBJ(n3,n4), Hans(n4),
          ADJN(n3,n5), gerne(n5) }
```

$$\begin{bmatrix} \text{SUBJ} & [\text{PRED} \quad \text{EDE}]{[2]} \\ \text{PRED} & \text{VERMUTEN}\langle\uparrow\text{SUBJ},\uparrow\text{COMP}\rangle \\ \text{COMP} & \begin{bmatrix} \text{SUBJ} & [\text{PRED} \quad \text{HANS}]{[4]} \\ \text{PRED} & \text{KOCHEN}\langle\uparrow\text{SUBJ}\rangle \\ \text{ADJN} & \{[\text{PRED} \quad \text{GERNE}]{[5]}\} \end{bmatrix}{[3]} \end{bmatrix}{[1]}$$

Ede vermutet daß Hans gerne kocht

Right diagram:

$$l_\top : \boxed{\begin{array}{l} x_{[2]}, x_{[4]} \\ Ede(x_{[2]}) \\ Hans(x_{[4]}) \end{array}}$$

$$l_{[1]} : \boxed{supect(x_{[2]}, l_{[1]1})}$$

$$l_{[5]} : \boxed{like(x_{[4]}, l_{[5]1})}$$

$$l_{[3]} : \boxed{cook(x_{[4]})}$$

```
{ suspect(n1),
  SUBJ(n1,n2), Ede(n2),
  COMP(n1,n5), like(n5),
          SUBJ(n5,n4), Hans(n4),
          XCOMP(n5,n3), cook(n3),
                  SUBJ(n3,n4) }
```

$$\begin{bmatrix} \text{SUBJ} & [\text{PRED} \quad \text{EDE}]{[2]} \\ \text{PRED} & \text{SUSPECT}\langle\uparrow\text{SUBJ},\uparrow\text{COMP}\rangle \\ \text{COMP} & \begin{bmatrix} \text{SUBJ} & [\text{PRED} \quad \text{HANS}]{[4]} \\ \text{PRED} & \text{LIKE}\langle\uparrow\text{SUBJ},\uparrow\text{XCOMP}\rangle \\ \text{XCOMP} & \begin{bmatrix} \text{SUBJ} & [\;]{[4]} \\ \text{PRED} & \text{COOK}\langle\uparrow\text{SUBJ}\rangle \end{bmatrix}{[3]} \end{bmatrix}{[5]} \end{bmatrix}{[1]}$$

Ede suspects that Hans likes cooking

Figure 1: Embedded Head Switching Example

4.2 Multiple Adjuncts

Consider the sentences in (6).

(6) a. *Oft kocht Hans gerne*

   b. *Hans kocht gerne oft*

   c. *Often Hans likes cooking*

   d. *Hans likes cooking often*

(6a) is ambiguous between (6c) and (6d), (6b) can only mean (6d). (6c) and (6d) are not ambiguous. (6a) is represented by f-structure (7a).

(7) a.
$$\begin{bmatrix} \text{SUBJ} & [\text{PRED HANS}]{[2]} \\ \text{PRED} & \text{KOCHEN}\langle\uparrow\text{SUBJ}\rangle \\ \text{ADJN} & \left\{ \begin{array}{l} [\text{PRED OFT}]{[3]} \\ [\text{PRED GERNE}]{[4]} \end{array} \right\} \end{bmatrix}{[1]}$$

   b.
```
{ kochen(n1),
  SUBJ(n1,n2), Hans(n2),
  ADJN(n1,n3), oft(n3),
  ADJN(n1,n4), gerne(n4) }
```

   c.
$$l_\top : \boxed{\begin{array}{l} x_{[2]} \\ Hans(x_{[2]}) \end{array}}$$

$$l_{[3]} : \boxed{oft(l_{[3]1})} \quad l_{[4]} : \boxed{gerne(l_{[4]1})}$$

$$l_{[1]} : \boxed{kochen(x_{[2]})}$$
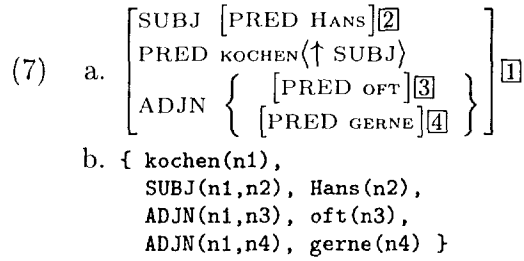
The corresponding term representation is (7b) and, in the absence of further constraints, we get a flat scopally underspecified UDRS (7c). Let (6a) be our translation candidate. For *syntactic transfer*, adding rules (9) to the ones introduced in (2) leads to (8a).

(8) a.
```
{ like(n4),
  SUBJ(n4,n2), Hans(n2),
  XCOMP(n4,n1), cook(n1),
  SUBJ(n1,n2),
  ADJN(n1,n3), often(n3) }
```

   b.
$$\begin{bmatrix} \text{SUBJ} & [\text{PRED HANS}]{[2]} \\ \text{PRED} & \text{LIKE}\langle\uparrow\text{SUBJ},\uparrow\text{XCOMP}\rangle \\ \text{XCOMP} & \begin{bmatrix} \text{SUBJ} & [\;]{[2]} \\ \text{PRED} & \text{COOK}\langle\uparrow\text{SUBJ}\rangle \\ \text{ADJN} & \{[\text{PRED OFTEN}]{[3]}\} \end{bmatrix}{[1]} \end{bmatrix}{[4]}$$

   c.
$$l_\top : \boxed{\begin{array}{l} x_{[2]} \\ Hans(x_{[2]}) \end{array}}$$

$$l_{[4]} : \boxed{like(x_{[2]}, l_{[4]1})}$$

$$l_{[3]} : \boxed{often(l_{[3]1})}$$

$$l_{[1]} : \boxed{cook(x_{[2]})}$$
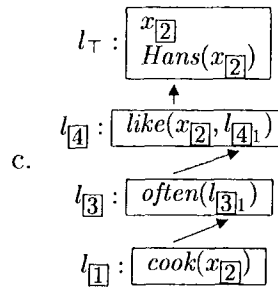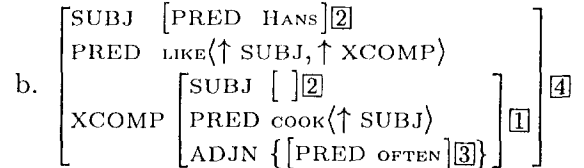
(9) { ADJN(E,X) } <-> { ADJN(E,X) }.
    { oft(E) } <-> { often(E) }.

(8a) corresponds to only one of the English translations, namely (6d), of (6a). As in the correspondence-based approach (Kaplan et al., 1989), *often* can only be assigned wide scope over *like* if the transfer formalism allows reference to and rewriting of partial nodes. In the present case the two terms kochen(n1), SUBJ(n1,n2) could then be rewritten as the complement of *like*, XCOMP(n4,n1), whereas ADJN(n1,n3) is rewritten as ADJN(n4,n3) or ADJN(n1,n3).[6]

The target f-structure for English must resolve the relative scope between *like* and *often* ((8b) and (10)).

$$(10)\quad \begin{bmatrix} \text{SUBJ} & [\text{PRED HANS}]_{\boxed{2}} \\ \text{PRED} & \text{LIKE}\langle\uparrow\text{SUBJ},\uparrow\text{XCOMP}\rangle \\ \text{XCOMP} & \begin{bmatrix} \text{SUBJ} & [\ ]_{\boxed{2}} \\ \text{PRED} & \text{COOK}\langle\uparrow\text{SUBJ}\rangle \end{bmatrix}_{\boxed{1}} \\ \text{ADJN} & \{[\text{PRED OFTEN}]_{\boxed{3}}\} \end{bmatrix}_{\boxed{4}}$$

Semantic transfer on the source UDRS (7c) preserves the underspecification and leads to (11).

(11)

However, (11) is not in the direct f-structure – UDRS correspondence with (10) and (8b). Instead, the correspondences on the enumerations of the scoping possibilities of (11) yield (10) and (8b) as required.

By contrast, the reading of (6b) is restricted by the surface order in which the two adverbials occur. On the semantic level this is reflected in terms of corresponding subordination constraints (12). The target UDRS corresponds to f-structure (8b).

---

[6]As an alternative, we can get both readings if we define special rules for adverbials in head switching contexts, giving them wide or narrow scope relative to the head switching adverbial. A narrow scope rule is already given in (9). A wide scope rule would be {ADJN(E,X)} # {HS(E1), XCOMP(E1,E)} ↔ {ADJN(E1,X)} where HS(E1) is a "marker" on the switched adverbial's node E1.

(12)

In LFG linearization effects can be captured in terms of f-precedence constraints $\prec_f$ as in (13).

$$(13)\quad \begin{bmatrix} \text{ADJN} & \{[\text{PRED }\alpha]_{\boxed{2}}, [\text{PRED }\beta]_{\boxed{3}}\} \\ & \{\boxed{2} \prec_f \boxed{3}\} \end{bmatrix}_{\boxed{1}}$$

Semantic subordination and f-precedence constraints can then be linked as in (14).

$$(14)\quad \boxed{1} \prec_f \boxed{J} \triangleleft\triangleright l_{\boxed{J}} \leq l_{\boxed{1}1}$$

With (14) the head switching – multiple adjunct interaction is correctly resolved in semantic-based transfer. Similarly, in syntactic transfer, the precedence constraint (13) can be used to steer translation to f-structure (8b).

5 Discussion

We have presented two alternative architectures for transfer in LFG. In both cases, transfer is driven by the transfer module developed and implemented by Dorna and Emele (1996a). In the case of syntactic transfer, transfer is defined on term representations of f-structures. In the case of semantic transfer, transfer is defined on UDRS translations of f-structures. F-structure, term and UDRS correspondences are defined in the Appendix. The transfer rules are bi-directional, as are the f-structure–term and f-structure–UDRS correspondences.

Co-description based approaches (Kaplan and Wedekind, 1993) require annotation of source and target lexica and grammars. By contrast, both approaches presented here support modular grammar development: they don't involve additional coding in the grammar specifications.

An important issue, noted above, is the problem of ambiguities and ambiguity preserving transfer. F-structures and UDRSs are underspecified syntactic and semantic representations, respectively. Both support ambiguity preserving transfer to differing degrees (NP scope, operators, adjuncts). F-structure based syntactic represen-

tations may come up against structural mismatches in transfer. The original co-description based approach in (Kaplan et al., 1989) faced problems when it came to examples involving embedded head-switching and multiple adjuncts (Sadler and Thompson, 1991), which led to the introduction of a restriction operator, to enable transfer on partial f-structures or semantic structures (Kaplan and Wedekind, 1993). One might suppose that the need to refer to partial structures is an artifact of the correspondence-based approach, which doesn't allow the mapping from a single node of the source f-structure to distinct nodes in the target f-structure without violation of the functional property of the τ correspondence. On closer inspection, though, the rewriting approach to syntactic f-structure–term translations presented above suffers from the very same problems that were met by the correspondence-based approach in (Kaplan et al., 1989).

By contrast, transfer on the semantic UDRS representations does not suffer from such problems. Head switching is dealt with in the construction of semantic representations. Underspecified semantic representations in the form of UDRSs (or related formalisms) offer the following advantages for transfer: they abstract away from cross-language configurational variation to facilitate transfer. Unlike the original restriction operator approach (Kaplan and Wedekind, 1993) whenever possible they avoid the detour of multiple transfer on disambiguated representations. At the same time they provide a flexible encoding of information essential to steer transfer.

Of course, semantics does not come for free nor does it always blend as seamlessly with syntactic representations as one would hope for. Semantics has to be encoded in the grammar or defined in terms of correspondences as below. System design has to address the question where to do what at which cost. Semantic representations pay off when they are useful for a number of tasks: evaluation (as against a database), inference and transfer. Even more so when existing resources can be interfaced qua semantic representations: in our case the tested transfer methodology and resources developed in (Dorna and Emele, 1996a).

References

H. Alshawi and R. Crouch. 1992. Monotonic semantic interpretation. In *Proceedings of ACL*, pages 32–39, Newark, Delaware.

J. Bos, B. Gambäck, C. Lieske, Y. Mori, M. Pinkal, and K. Worm. 1996. Compositional Semantics in Verbmobil. *Coling'96*, pages 131–136, Copenhagen, Denmark.

R. Caspari and L. Schmid. 1994. Parsing und Generierung in TrUG. Verbmobil Report 40, Siemens AG, December.

M. Dalrymple, J. Lamping, F.C.N Pereira, and V. Saraswat. 1996. A deductive account of quantification in lfg. In M. Kanazawa, C. Pinon, and H. de Swart, editors, *Quantifiers, Deduction and Context*, pages 33–57. CSLI Publications, No. 57.

M. Dorna and M. C. Emele. 1996a. Efficient Implementation of a Semantic-based Transfer Approach. *ECAI'96*, Budapest, Hungary.

M. Dorna and M. C. Emele. 1996b. Semantic-based Transfer. *Coling'96*, Copenhagen, Denmark.

M. C. Emele and M. Dorna. 1998. Ambiguity Preserving Transfer Using Packed Representations. *Coling'98*, Montréal, Canada.

J. van Genabith and R. Crouch. 1996. Direct and underspecified interpretations of lfg f-structures. In *COLING 96, Copenhagen, Denmark*, pages 262–267.

J. van Genabith and R. Crouch. 1997. On interpreting f-structures as udrss. In *ACL-EACL-97, Madrid, Spain*, pages 402–409.

J. van Genabith, A. Frank, and M. Dorna. 1998. Transfer Constructors. *LFG Conference '98, Brisbane, Australia*.

M. Johnson. 1991. Features and Formulae. *Computational Linguistics*, 17(2):131–151.

R. M. Kaplan and J. Wedekind. 1993. Restriction and Correspondance-based Translation. *EACL'93*, pages 193–202, Utrecht, The Netherlands.

R. Kaplan, K. Netter, J. Wedekind, and A. Zaenen. 1989. Translation by Structural Correspondences. *EACL'89*, pages 272–281, Manchester, UK.

M. Kay, M. Gawron, and P. Norwig. 1994. *Verbmobil: a Translation System for Face-to-Face Dialogs*. Number 33 in CSLI Lecture Notes. University of Chicago Press.

John T. Maxwell III and Ronald M. Kaplan. 1993. The interface between phrasal and functional constraints. *Computational Linguistics*, 19(4):571–590.

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

U. Reyle. 1993. Dealing with Ambiguities by Underspecification: Construction, Representation and Deduction. *Journal of Semantics*, 10(2):123–179.

L. Sadler and H. S. Thompson. 1991. Structural Non-correspondence in Translation. *EACL'91*, pages 293–298, Berlin, Germany.

A F-Structures and Terms

A 2-place relation between f-structures and sets of terms is defined below. $\boxed{i}$ are references to feature structures which are mapped into node constants ni used in terms. Γ are features (grammatical functions), and φ are f-structures. Predicates occur as $\Pi\langle\rangle$ if they do not subcategorize for an argument, else as $\Pi\langle\uparrow\Gamma_1,\dots,\uparrow\Gamma_n\rangle$.

1. (simple predicates)
$$\langle[\text{PRED}\quad \Pi\langle\rangle]\boxed{i},\quad \{\Pi(ni)\}\rangle$$

2. (complex predicates)
$$\left\langle\begin{bmatrix}\text{PRED} & \Pi\langle\uparrow\Gamma_1,\dots,\uparrow\Gamma_n\rangle\\ \Gamma_1 & \varphi_1\boxed{i_1}\\ \dots & \\ \Gamma_n & \varphi_n\boxed{i_n}\end{bmatrix}\boxed{i_0},\right.$$
$$\{\Pi(ni_0),\Gamma_1(ni_0,ni_1),\dots,\Gamma_n(ni_0,ni_n)\}$$
$$\cup\, T_1\cup\dots\cup T_n\rangle$$
$$\Longleftrightarrow \langle\varphi_1\boxed{i_1},T_1\rangle\wedge\dots\wedge\langle\varphi_n\boxed{i_n},T_n\rangle$$

3. (set values)
$$\langle[\text{ADJN}\quad\{\alpha_1\boxed{i_1},\dots,\alpha_m\boxed{i_n}\}]\boxed{i_0},$$
$$\{\texttt{ADJN}(ni_0,ni_1),\dots,\texttt{ADJN}(ni_0,ni_n)\}\,\cup$$
$$T_1\cup\dots\cup T_n\rangle$$
$$\Longleftrightarrow \langle\alpha_1\boxed{i_1},T_1\rangle\wedge\dots\wedge\langle\alpha_n\boxed{i_n},T_n\rangle$$

B F-Structures and UDRSs

In (Genabith and Crouch, 1997) the correspondence between f-structures and UDRSs was defined in terms of translation functions $\tau:$ and τ^{-1} between subsets of the f-structure and UDRS formalisms. Below we give a relational formulation of the correspondence $\lhd\rhd$ with a treatment of simple (scopal) adjuncts:[7]

1.
$$\begin{bmatrix}\text{PRED} & \Pi\langle\uparrow\Gamma_1,\dots,\uparrow\Gamma_n\rangle\\ \Gamma_1 & \varphi_1\boxed{I_1}\\ \dots & \\ \Gamma_n & \varphi_n\boxed{I_n}\\ \text{ADJN} & \{\alpha_1\boxed{J_1},\dots,\alpha_m\boxed{J_m}\}\end{bmatrix}\boxed{I}\quad\lhd\rhd$$
$$\{l_{\boxed{I}_\perp}:\Pi(\eta_{\boxed{I_1}},\dots,\eta_{\boxed{I_n}}),l_{\boxed{I}_\perp}\le l_{\boxed{I}^\top}\}\cup S$$
$$\cup\, A_1\cup\dots\cup A_m\cup F_1\cup\dots\cup F_n$$
$$\Longleftrightarrow$$
$$\Pi\langle\uparrow\Gamma_{\boxed{I_1}},\dots,\uparrow\Gamma_{\boxed{I_n}}\rangle\ \overset{\boxed{I}}{\lhd\rhd}\ \{\Pi(\eta_{\boxed{I_1}},\dots,\eta_{\boxed{I_n}})\}$$
$$\cup\, S\wedge\bigwedge_{i=1}^{n}\varphi_i\boxed{I_i}\ \overset{\boxed{I}}{\lhd\rhd}\ U_i\wedge\bigwedge_{i=1}^{m}\alpha_i\boxed{J_i}\ \overset{\boxed{I}}{\lhd\rhd_\alpha}\ A_i$$

2.
$$\begin{bmatrix}\text{SPEC} & Q\\ \text{PRED} & \Pi\langle\rangle\end{bmatrix}\boxed{I}\quad\overset{\boxed{J}}{\lhd\rhd}$$
$$\left\{\begin{array}{l}l_{\boxed{I}}:l_{\boxed{I}_1}Qx_{\boxed{I}}l_{\boxed{I}_2},l_{\boxed{I}_1}:x_{\boxed{I}},\\ l_{\boxed{I}_1}:\Pi(x_{\boxed{I}}),l_{\boxed{I}}\le l_\top,l_{\boxed{J}_\perp}\le l_{\boxed{I}_2}\end{array}\right\}$$

3.
$$\begin{bmatrix}\text{SPEC} & \wedge\\ \text{PRED} & \Pi\langle\rangle\end{bmatrix}\boxed{I}\quad\overset{\boxed{J}}{\lhd\rhd}$$
$$\left\{\begin{array}{l}l_{\boxed{I}}:x_{\boxed{I}},l_{\boxed{I}}:\Pi(x_{\boxed{I}}),\\ l_{\boxed{I}}\le l_\top,l_{\boxed{J}_\perp}\le l_{\boxed{I}}\end{array}\right\}$$

4.
$$[\text{PRED}\quad\Pi\langle\rangle]\boxed{I}\quad\overset{\boxed{J}}{\lhd\rhd}$$
$$\{l_\top:x_{\boxed{I}},l_\top:\Pi(x_{\boxed{I}}),l_{\boxed{J}_\perp}\le l_\top\}$$

5.
$$\begin{bmatrix}\text{PRED} & \Pi\langle\uparrow\Gamma_1,\dots,\uparrow\Gamma_n\rangle\\ \Gamma_1 & \varphi_1\boxed{I_1}\\ \dots & \\ \Gamma_n & \varphi_n\boxed{I_n}\\ \text{ADJN} & \Sigma\end{bmatrix}\boxed{I}\quad\overset{\boxed{J}}{\lhd\rhd}\ U\ \Longleftrightarrow$$
$$\begin{bmatrix}\text{PRED} & \Pi\langle\uparrow\Gamma_1,\dots,\uparrow\Gamma_n\rangle\\ \Gamma_1 & \varphi_1\boxed{I_1}\\ \dots & \\ \Gamma_n & \varphi_n\boxed{I_n}\\ \text{ADJN} & \Sigma\end{bmatrix}\boxed{I}\quad\lhd\rhd\ U$$

6.
$$\Pi\langle\uparrow\Gamma_1\boxed{I_1},\dots,\uparrow\Gamma_n\boxed{I_n}\rangle\ \overset{\boxed{I}}{\lhd\rhd}$$
$$\{\Pi(\eta_{\boxed{I_1}},\dots,\eta_{\boxed{I_n}})\}\cup S$$

holds iff there is a lexically specified map between subcategorizable grammatical functions in LFG semantic form and argument positions in the corresponding UDRT predicate, e.g.:

$$\begin{array}{l}\{l_{\boxed{I_2}^\top}\le l_{\boxed{I_2}_\perp}\}\cup\\ \{like(\quad x_{\boxed{I_1}},\quad l_{\boxed{I_2}_\perp}\quad)\}\\ \qquad\qquad\uparrow\qquad\qquad\uparrow\\ \text{LIKE}\langle\ \uparrow\text{SUBJ}\boxed{I_1},\ \uparrow\text{XCOMP}\boxed{I_2}\ \rangle\end{array}$$

7.
$$[\text{PRED}\quad\Pi\langle\rangle]\boxed{I}\quad\overset{\boxed{J}}{\lhd\rhd_\alpha}$$
$$\{l_{\boxed{I}}:\Pi(l_{\boxed{I}_1}),l_{\boxed{I}}\le l_{\boxed{J}^\top},l_{\boxed{J}_\perp}\le l_{\boxed{I}_1}\}$$

F-structures and UDRSs are in the $\lhd\rhd$ relation iff their components are $\lhd\rhd$ related (clause 1). $\lhd\rhd$ relates f-structure tags and UDRS labels. Clausal tags $\boxed{I}$ introduce a local top $\boxed{I}^\top$ and a local bottom $\boxed{I}^\perp$. The global top is $\top$. For readability, tops and bottoms are suppressed in the example translations. η refers to discourse referents or labels. S in clause 1 is a set of subordination constraints induced lexically by embedding verbs (clause 6). Clauses 2 – 4 relate quantificational, indefinite and proper name f-structure and UDRS components, clause 5 embedded clauses. Clause 7 translates simple adjuncts.

---

[7]In LFG adjuncts do not subcategorize the material they modify nor are they subcategorized by that material.