Gefördert vom
Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie

Verbmobil
Verbundvorhaben

Semantic-based Transfer

Michael Dorna and Martin Emele

Universität Stuttgart

Report 143
July 1996

July 1996

Michael Dorna and Martin Emele

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

Tel.: (0711) 121 - 1363
Fax: (0711) 121 - 1366

Gehört zum Antragsabschnitt: 12 Transfer

Die vorliegende Arbeit wurde im Rahmen des Verbundvorhabens Verbmobil vom Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie (BMBF) unter dem Förderkennzeichen 01 IV 101 U gefördert. Die Verantwortung für den Inhalt dieser Arbeit liegt bei den Autoren.

Semantic-based Transfer[1]

Michael Dorna and Martin C. Emele

Abstract

This article presents a new semantic-based transfer approach developed and applied within the Verb*mobil* Machine Translation project. We give an overview of the declarative transfer formalism together with its procedural realization. Our approach is discussed and compared with several other approaches from the MT literature. The results presented in this article have been implemented and integrated into the Verb*mobil* system.

1 Introduction

The work presented in this article was developed within the Verb*mobil* project (Kay et al., 1994; Wahlster, 1993). This is one of the largest projects dealing with Machine Translation (MT) of spoken language. Approximately 100 researchers in 29 public and industrial institutions are involved. The application domain is spontaneous spoken language in face-to-face dialogs. The current scenario is restricted to the task of appointment scheduling and the languages involved are English, German and Japanese.

This article describes the realization of a transfer approach based on the proposals of (Abb and Buschbeck-Wolf, 1995; Caspari and Schmid, 1994) and (Copestake, 1995). Transfer-based MT[2], see e.g. (Vauquois and Boitet, 1985; Nagao et al., 1985), is based on contrastive bilingual corpus analyses from which a bilingual lexicon of transfer equivalences is derived. In contrast to a purely lexicalist approach which relates bags of lexical signs, as in Shake-and-Bake MT (Beaven, 1992; Whitelock, 1992), our transfer approach operates on the level of semantic representations produced by various analysis steps. The output of transfer is a semantic representation for the target language which is input to the generator and speech synthesis to produce the target language utterance. Our transfer equivalences abstract away from morphological and syntactic idiosyncracies of source and target languages. The bilingual equivalences are described on the basis of semantic representations.

Since the Verb*mobil* domain is related to discourse rather than isolated sentences the model theoretic semantics is based on Kamp's Discourse Representation Theory, DRT (Kamp and Reyle, 1993). In order to allow for underspecification, variants of Underspecified Discourse Representation Structures (UDRS) (Reyle, 1993) are employed as semantic formalisms in the different analysis components (Bos et al., 1996; Egg and Lebeth, 1995; Copestake et al., 1995).

[1]This article is pulished in the *Proceedings of the 16th International Conference on Computational Linguistics, Copenhagen, 1996* (COLING-96).
We would like to thank our colleagues of the Verb*mobil* subproject Transfer, our IMS colleagues Ulrich Heid and C.J. Rupp and the anonymous reviewers of COLING-96 for useful feedback and discussions on earlier drafts of the paper.
[2]For a more detailed overview of different approaches to MT, see e.g. (Hutchins and Somers, 1992).

Together with other kinds of information, such as tense, aspect, prosody and morpho-syntax, the different semantic representations are mapped into a single multi-dimensional representation called Verb*mobil* Interface Term (VIT) (Dorna, 1996). This single information structure serves as input to semantic evaluation and transfer. The transfer output is also a VIT which is based on the semantics of the English grammar (cf. Copestake et al. (1995)) and used for generation (see Kilger and Finkler (1995) for a description of the generation component).

Section 2 of this paper sketches the semantic representations we have used for transfer. In section 3 we introduce transfer rules and discuss examples. In section 4 we compare our approach with other MT approaches. In section 5 we present a summary of the implementation aspects. For a more detailed discussion of the implementation of the transfer formalism see Dorna and Emele (1996). Finally, section 6 summarizes the results.

2 Semantic Representations

The different Verb*mobil* semantic construction components use variants of UDRS as their semantic formalisms, cf. (Bos et al., 1996; Egg and Lebeth, 1995; Copestake et al., 1995). The ability to underspecify quantifier and operator scope together with certain lexical ambiguities is important for a practical machine translation system like Verb*mobil* because it supports ambiguity preserving translations. The disambiguation of different readings could require an arbitrary amount of reasoning on real-world knowledge and thus should be avoided whenever possible.

In the following examples we assume an explicit event-based semantics (Dowty, 1989; Parsons, 1991) with a Neo-Davidsonian representation of semantic argument relations. All semantic entities in UDRS are uniquely labeled. A label is a pointer to a semantic predicate making it easy to refer to. The labeling of all semantic entities allows a flat representation of the hierarchical structure of argument and operator and quantifier scope embeddings as a set of labeled conditions. The recursive embedding is expressed via additional subordination constraints on labels which occur as arguments of such operators.

Example (1a) shows one of the classical Verb*mobil* examples and its possible English translation (1b).

(1) a. *Das paßt echt schlecht bei mir.*
b. *That really doesn't suit me well.*

The corresponding semantic representations are given in (2a) and (2b), respectively.[3]

[3]For presentation purposes we have simplified the actual VIT representations.

(2) a. `[l1:echt(l2), l2:schlecht(i1),`
`l3:passen(i1), l3:arg3(i1,i2),`
`l4:pron(i2), l5:bei(i1,i3), l6:ich(i3)]`

b. `[l1:real(l2), l2:neg(l7), l7:good(i1),`
`l3:suit(i1), l3:arg3(i1,i2),`
`l4:pron(i2), l5:arg2(i1,i3), l6:ego(i3)]`

Semantic entities in (2) are represented as a Prolog list of labeled conditions. After the unification-based semantic construction, the logical variables for labels and markers, such as events, states and individuals, are skolemized with special constant symbols, e.g. `l1` for a label and `i1` for a state. Every condition is prefixed with a label serving as a unique identifer. Labels are also useful for grouping sets of conditions, e.g. for partitions which belong to the restriction of a quantifier or which are part of a specific sub-DRS. Additionally, all these special constants can be seen as pointers for adding or linking information within and between multiple levels of the VIT.

Only the set of semantic conditions is shown in (2); the other levels of the multi-dimensional VIT representation, which contain additional semantic, pragmatic, morpho-syntactic and prosodic information, have been left out here. If necessary, such additional information can be used in transfer and semantic evaluation for resolving ambiguities or in generation for guiding the realization choices. Furthermore, it allows transfer to make fine-grained distinctions between alternatives in cases where the semantic representations of source and target language do not match up exactly.

Semantic operators like negation, modals or intensifier adverbials, such as *really*, take extra label arguments for referring to other elements in the flat list which are in the relative scope of these operators.[4]

This form of semantic representation has the following advantages for transfer:

- It is possible to preserve the underspecification of quantifier and operator scope if there is no divergence regarding scope ambiguity between source and target languages.
- Coindexation of labels and markers in the source and target parts of transfer rules ensures that the semantic entities are correctly related and hence obey any semantic constraints which may be linked to them.
- To produce an adequate target utterance additional constraints which are important for generation, e.g. sortal, topic/focus constraints etc., may be preserved.
- There need not be a 1 : 1 relation between semantic entities and individual lexical items. Instead, lexical units may be decomposed into a set of semantic

[4]For the concrete example at hand, the relative scope has been fully resolved by using the explicit labels of other conditions. If the scope were underspecified, explicit subordination constraints would be used in a special scope slot of the VIT. The exact details of subordination are beyond the scope of this paper, cf. Frank and Reyle (1995) and Bos et al. (1996) for implementations.

entities, e.g. in the case of derivations and for a more fine grained lexical semantics. Lexical decomposition allows us to express generalizations and to apply transfer rules to parts of the decomposition.

3 Our Transfer Approach

Transfer equivalences are stated as relations between sets of source language (SL) and sets of target language (TL) semantic entities. They are usually based on individual lexical items but might also involve partial phrases for treating idioms and other collocations, e.g. verb-noun collocations (see example (8) below). After skolemization of the semantic representation the input to transfer is variable free. This allows the use of logical variables for labels and markers in transfer rules to express coindexation constraints between individual entities such as predicates, operators, quantifiers and (abstract) thematic roles. Hence the skolemization prevents unwanted unification of labels and markers while matching individual transfer rules against the semantic representation.

The general form of a transfer rule is given by

```
SLSem, SLConds TauOp TLSem, TLConds.
```

where `SLSem` and `TLSem` are sets of semantic entities. `TauOp` is an operator indicating the intended application direction (one of `<->,->,<-`). `SLConds` and `TLConds` are optional sets of SL and TL conditions, respectively. All sets are written as Prolog lists and optional conditions can be omitted.

On the source language, the main difference between the `SLSem` and conditions is that the former is matched against the input and replaced by the `TLSem`, whereas conditions act as filters on the applicability of individual transfer rules without modifying the input representation. Hence conditions may be viewed as general inferences which yield either true or false depending on the context. The context might either be the local context as defined by the current VIT or the global context defined via the domain and dialog model. Those inferences might involve arbitrarily complex inferences like anaphora resolution or the determination of the current dialog act. In an interactive system one could even imagine that conditions are posed as yes/no-questions to the user to act as a negotiator (Kay et al., 1994) for choosing the most plausible translation.

If the translation rules in (3) are applied to the semantic input in (2a) they yield the semantic output in (2b). We restrict the following discussion to the direction from German to English but the rules can be applied in the other direction as well.

```
(3) a. [L:echt(A)] <-> [L:real(A)].
    b. [L:passen(E),L:arg3(E,Y),L1:bei(E,X)]<->
       [L:suit(E),L:arg2(E,X),L:arg3(E,Y)].
    c. [L:schlecht(E)],[L1:passen(E)] <->
       [L:neg(A),A:good(E)].
    d. [L:ich(X)] <-> [L:ego(X)].
    e. [L:pron(X)] <-> [L:pron(X)].
```

The simple lexical transfer rule in (3a) relates the German intensifier `echt` with the English `real`[5]. The variables `L` and `A` ensure that the label and the argument of the German `echt` are assigned to the English predicate `real`, respectively.

The equivalence in (3b) relates the German predicate `passen` with the English predicate `suit`. The rule not only identifies the event marker `E`, but unifies the instances `X` and `Y` of the relevant thematic roles. Despite the fact that the German *bei*-phrase is analysed as an adjunct, it is treated exactly like the argument `arg3` which is syntactically subcategorized. This rule shows how structural divergences can easily be handled within this approach.

(4)
```
[L:passen(E), L1:bei(E,X)] <->
[L:suit(E),   L:arg2(E,X)].
```

The rule in (3b) might be further abbreviated to (4) by leaving out the unmodified `arg3`, because it is handled by a single metarule, which passes on all semantic entities that are preserved between source and target representation. This also makes the rule for (3e) superfluous, since it uses an interlingua predicate for the anaphor in German and English.

The rule in (3c) illustrates how an additional condition (`[L1:passen(E)]`) might be used to trigger a specific translation of *schlecht* into *not good* in the context of *passen*. The standard translation of *schlecht* to *bad* is blocked for verbs like *suit*, that presuppose a positive attitude adverbial.[6] One main advantage of having such conditions is the preservation of the modularity of transfer equivalences because we do not have to specify the translation of the particular verb which only triggers the specific translation of the adverbial. Consequently, the transfer units remain small and independent of other elements, thus the interdependencies between different rules are vastly reduced. The handling of such rule interactions is known to be one of the major problems in scaling up MT systems.

A variation on example (1) is given in (5).

(5) a. *Das paßt mir echt schlecht.*
b. *That really doesn't suit me well.*

The translation is exactly the same, but the German verb *passen* takes an indirect object *mir* instead of the adjunct *bei*-phrase in (1). The appropriate transfer rule looks like (6a) which can be reduced to (6b) because no argument switching takes place and we can use the metarule again.

(6)
```
a.[L:passen(E),L:arg2(E,X),L:arg3(E,Y)]<->
  [L:suit(E),  L:arg2(E,X),L:arg3(E,Y)].
b. [L:passen(E)] <-> [L:suit(E)].
```

[5]The semantic predicate `real` abstracts away from the adjective/adverbial distinction.

[6]Instead of using a specific lexical item like *passen* the rule should be abstracted for a whole class of verbs with similar properties by using a type definition, e.g. `type(de,pos_attitude_verbs,[gehen,passen,...])`. For a description of type definitions see (11) below.

In a purely monotonic system without overriding it would be possible to apply the transfer rule in (6b) to sentence (1) in addition to the rule in (4) leading to a wrong translation. Whereas in the underlying rule application scheme assumed here, the more general rule in (6b) will be blocked by the more specific rule in (4).

The specificity ordering of transfer rules is primarily defined in terms of the cardinality of matching subsets and by the subsumption order on terms. In addition, it also depends on the cardinality and complexity of conditions. For the *passen* example at hand, the number of matching predicates in the two competing transfer rules defines the degree of specificity.

The following example illustrates how conditions are used to enforce selectional restrictions from the domain model. For example *Termin* in German might either be translated as *appointment* or as *date*, depending on the context.

(7) a. `[L:termin(X)] <-> [L:appointment(X)].`
b. `[L:termin(X)],`
`[sort(X)=<~temp_point] <-> [L:date(X)].`

The second rule (7b) is more specific, because it uses an additional condition. This rule will be tried first by calling the external domain model for testing whether the sort assigned to `X` is not subsumed by the sort `temp_point`. Here, the first rule (7a) serves as a kind of default with respect to the translation of *Termin*, in cases where no specific sort information on the marker `X` is available or the condition in rule (7b) fails.

In (8), a light verb construction like *einen Terminvorschlag machen* is translated into *suggest a date* by decomposing the compound and light verb to a simplex verb and its modifying noun.

(8)
```
[L:machen(E),L:arg3(E,X),
 L1:terminvorschlag(X)] <->
[L:suggest(E),L:arg3(E,X),L1:date(X)].
```

We close this section with a support verb example (9) showing the treatment of head switching in our approach. The German comparative construction *lieber sein* (lit.:*be more liked*) in (9a) is translated by the verb *prefer* in (9b).

(9) a. *Dienstag ist mir lieber.*
b. *I would prefer Tuesday.*

(10)
```
[L:support(S,L1),L2:experiencer(S,X)
 L1:lieb(Y),L1:comparative(Y)] <->
[L:prefer(S),L:arg1(S,X),L:arg3(S,Y)].
```

The transfer rule in (10) matches the decomposition of the comparative form *lieber* into its positive form *lieb* and an additional comparative predicate together with the support verb *sein* such that the comparative construction *lieber sein* (`Y` *ist* `X` *lieber*) is translated as a whole to the English verb *prefer* (`X` *prefers* `Y`).

4 Discussion

The main motivation for using a semantic-based approach for transfer is the ability to abstract away from morphological and syntactic idiosyncrasies of individual languages. Many of the traditional cases of divergences discussed, e.g. by Dorr (1994), are already handled in the Verb*mobil* syntax-semantics interface, hence they do not show up in our transfer approach. Examples include cases of categorial and thematic divergences. These are treated in the linking between syntactic arguments and their corresponding thematic roles.

Another advantage of a semantic-based transfer approach over a pure interlingua approach, e.g. Dorr (1993), or a direct structural correspondence approach, e.g. Slocum et al. (1987), is the gain in modularity by allowing language independent grammar development. Translation equivalences relating semantic entities of the source and target grammars can be formulated in a grammar independent bilingual semantic lexicon. In cases where the semantic representations of source and target language are not isomorphic, a nontrivial transfer relation between the two representations is needed. But it is clearly much easier to map between flat semantic representations than between either syntactic trees or deeply nested semantic representations

An interlingua approach presumes that a single representation for arbitrary languages exists or can be developed. We believe from a grammar engineering point of view it is unrealistic to come up with such an interlingua representation without a strict coordination between the monolingual grammars. In general, a pure interlingua approach results in very application and domain specific knowledge sources which are difficult to maintain and extend to new languages and domains. This holds especially in the Verb*mobil* context with its distributed grammar development.

Whereas our approach does not preclude the use of interlingua predicates. We use interlingua representations for time and date expressions in the Verb*mobil* domain. Similarly for prepositions, cf. Buschbeck-Wolf and Nübel (1995), it makes sense to use more abstract relations which express fundamental relationships like *temporal location* or *spatial location*. Then it is left to the language specific grammars to make the right lexical choices.

(11) a. `type(de,temp_loc,[an,in,um,zu]).`
b. *am* Dienstag, *im* Mai, *um* drei, *zu* Ostern
c. `type(en,temp_loc,[on,in,at]).`
d. *on* Tuesday, *in* May, *at* three, *at* Easter

The class definitions in (11a) and (11c) cluster together those prepositions which can be used to express a temporal location. The names `de` and `en` are the SL and TL modules in which the class is defined, `temp_loc` is the class name and the list denotes the extension of the class. (11b) and (11d) show possible German and English lexicalizations.

(12)
```
[temp_loc(E,X)],[sort(X)=<time] <->
[temp_loc(E,X)].
```

The interlingua rule in (12) identifies the abstract temporal location predicates under the condition that the internal argument is more specific than the sort `time`. This condition is necessary because of the polysemy of those prepositions. During compilation the SL class definition will be automatically expanded to the individual predicates, whereas the TL class definition will be kept unexpanded such that the target grammar might be able to choose one of the idiosyncratic prepositions.

Mixed approaches like Kaplan et al. (1989) can be characterized by mapping syntax as well as a predicate-argument structure (f-structure). As already pointed out, e.g. in (Sadler and Thompson, 1991), this kind of transfer has problems with its own multiple level mappings, e.g. handling of verb-adverb head switching, and does not cleanly separate monolingual from contrastive knowledge, either. In Kaplan and Wedekind (1993) an improved treatment of head switching is presented but it still remains a less general solution.

A semantic approach is much more independent of different syntactic analyses which are the source of a lot of classical translation problems such as structural and categorial divergences and mismatches. In our approach grammars can be developed for each language independently of the transfer task and can therefore be reused in other applications.

At first glance, our approach is very similar to the semantic transfer approach presented in Alshawi et al. (1991). It uses a level of underspecified semantic representations as input and output of transfer. The main differences between our approach and theirs are the use of flat semantic representations and the non-recursive transfer rules. The set-oriented representation allows much simpler operations in transfer for accessing individual entities (set membership) and for combining the result of individual rules (set union). Furthermore, because the recursive rule application is not part of the rules themselves, our approach solves problems with discontinuous translation equivalences which the former approach cannot handle well. A transfer rule for such a case is given in (4).

Our current approach is strongly related to the Shake-and-Bake approach of Beaven (1992) and Whitelock (1992). But instead of using sets of lexical signs, i.e. morpho-syntactic *lexemes* as in Shake-and-Bake, we specify translation equivalences on sets of arbitrary *semantic entities*. Therefore, before entering the transfer component of our system, individual lexemes can already be decomposed into sets of such entities, e.g. for stating generalizations on the lexical semantics level or providing suitable representations for inferences. For example, the wh-question word *when* is decomposed into `temp_loc(E,X), whq(X,R), time(R,X)` (lit.: *at which time*), hence no additional transfer rules are required. Similarly, German composita like *Terminvorschlag* are decomposed into its compounds, e.g. `termin(i2), n_n(i1,i2), vorschlag(i1)` where `n_n` denotes a generic noun-noun relation. As a result a compositional translation as *proposal for a date* is possible without stating any additional translation equivalences to the ones for the simplex nouns.

Another major difference is the addition of conditions which trigger and block the applicability of individual transfer rules. For instance in the specific translation of *schlecht* to *not good* as defined in (3c), without conditions, one would have to add the verb *passen* into the bag to test for such a specific context. As a consequence the translation of the verb needs to be reduplicated, whereas in our approach, the translation of the verb can be kept totally independent of this specific translation of the adverbial, because the condition functions merely as a test.

These examples also illustrates the usefulness of labeled conditions, because the negation operator can take such a label as an argument and we can use unification again to achieve the correct coindexation. If we would use a hierarchical semantics instead, as in the original Shake-and-Bake aproach, where the negation operator embeds the verb semantics we would have to translate `schlecht(e), passen(e)` into `not(suit(e), well(e))` in one rule because there is no coindexation possible to express the correct embedding without the unique labeling of predicates.

Finally, we have filled the lack of an adequate control strategy for Shake-and-Bake by developing a nonmonotonic control strategy which orders more specific rules before less specific ones. This strategy allows the specification of powerful default translations. Whereas without such an ordering special care is needed to prevent a compositional translation in cases where a more specific noncompositional translation also exists.

The same argument about control holds in comparison to the unification-based transfer approach on Mimimal Recursion Semantics (MRS) (Copestake et al., 1995; Copestake, 1995). In addition, we use matching on first order terms instead of feature structure unification. Full unification might be problematic because it is possible to add arbitrary information during rule application, e.g. by further unifying different arguments. The other main difference is our nonmonotonic control component whereas the MRS approach assumes a monotonic computation of all possible transfer equivalences which are then filtered by the generation grammar. It is difficult to judge the feasibility of their approach given the fact that only a limited coverage has been addressed so far.

5 Implementation

A more detailed presentation of the implementation aspects of our transfer approach can be found in Dorna and Emele (1996). The current transfer implementation consists of a transfer rule compiler which takes a set of rules like the one presented in section 3 and compiles them into two executable Prolog programs one for each translation direction. The compiled program includes the selection of rules, the control of rule applications and calls to external processes if necessary.

Because both the transfer input and the matching part of the rules consist of sets we can exploit ordered set operations during compilation as well as at runtime to speed up the matching process and for computing common prefixes which are shared between different rules.

The compiled transfer program is embedded in the incremental and parallel architecture of the Verb*mobil* Prototype. Interaction with external modules, e.g. the domain model and dialog module or other inference components, is done via a set of predefined abstract interface functions which may be called in the condition part of transfer rules. The result is a fully transparent and modular interface for filtering the applicability of transfer rules.

6 Summary

This paper presents a new declarative transfer rule formalism, which provides an implementation platform for a semantic-based transfer approach. This approach combines ideas from a number of recent MT proposals and tries to avoid many of the well known problems of other transfer and interlingua approaches.

The declarative transfer correspondences are compiled into an executable Prolog program. The compiler exploits indexing for more efficient search of matching rules. There is a nonmonotonic but rule-independent control strategy based on rule specificity.

Currently, the transfer component contains about 1700 transfer rules. Thanks to the set orientation and indexing techniques we did not encounter any scaling problems and the average runtime performance for a 15 word sentence is about 30 milliseconds.

Future work will include the automatic acquisition of transfer rules from tagged bilingual corpora to extend the coverage and an integration of domain specific dictionaries.

References

B. Abb and B. Buschbeck-Wolf. 1995. A Minimal Transfer Conception for Verbmobil. Verb*mobil* Report 84, Institute for Logic and Linguistics, IBM Informationssysteme GmbH Heidelberg, Germany.

H. Alshawi, D. Carter, M. Rayner, and B. Gambäck. 1991. Translation by Quasi Logical Form Transfer. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL '91)*, pages 161–168, Berkeley.

J. L. Beaven. 1992. Shake-and-Bake Machine Translation. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING '92)*, Nantes, France.

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)*.

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.

R. Caspari and L. Schmid. 1994. Parsing und Generierung in TrUG. Verb*mobil* Report 40, Siemens AG, Munich, Germany.

A. Copestake, D. Flickinger, R. Malouf, S. Riehemann, and I. Sag. 1995. Transfer and Minimal Recursion Semantics. In *Proceedings of the 6th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI '95)*, Leuven, Belgium.

A. Copestake. 1995. Semantic Transfer in Verbmobil. Verb*mobil* Report 93, IMS, Universität Stuttgart, Germany and CSLI, Stanford University.

M. Dorna and M. Emele. 1996. Efficient implementation of a semantic-based transfer approach. In *Proceedings of the 12th European Conference on Artificial Intelligence (ECAI '96)*.

M. Dorna. 1996. The ADT-Package for the Verb*mobil* Interface Term. Verb*mobil* Report 104, IMS, Universität Stuttgart, Germany. (to appear).

B. J. Dorr. 1993. Interlingual Machine Translation: a Parameterized Approach. volume 63(1-2) of *Artificial Intelligence*, pages 429–493.

B. J. Dorr. 1994. Machine Translation Divergences: a Formal Description and Proposed Solution. volume 20(4) of *Computational Linguistics*, pages 597–633.

D. Dowty. 1989. On the Semantic Content of the Notion of 'Thematic Role'. In G. Chierchia, B. Partee, and R. Turner, editors, *Property Theory, Type Theory and Natural Language Semantics*. Dordrecht, Reidel.

M. Egg and K. Lebeth. 1995. Semantic underspecification and modifier attachment ambiguities. In *Beiträge zur 5. Fachtagung der Sektion Computerlinguistik der DGfS*.

A. Frank and U. Reyle. 1995. Principle Based Semantics for HPSG. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics (EACL '95)*, Dublin, Ireland.

W. J. Hutchins and H. L. Somers. 1992. *An Introduction to Machine Translation*. Academic Press, London, UK.

Hans Kamp and Uwe Reyle. 1993. *From Discourse to Logic: An Introduction to Modeltheoretic Semantics of Natural Language*. Kluwer, Dordrecht, Holland.

R. M. Kaplan and J. Wedekind. 1993. Restriction and correspondance-based translation. In *Proceedings of the 6th Conference of the European Chapter of the Association for Computational Linguistics (EACL '93)*, pages 193–202, Utrecht, The Netherlands.

R. Kaplan, K. Netter, J. Wedekind, and A. Zaenen. 1989. Translation by Structural Correspondences. In *Proceedings of the 4th Conference of the European Chapter of the Association for Computational Linguistics (EACL '89)*, pages 272–281, Manchester, UK.

M. Kay, M. Gawron, and P. Norwig. 1994. *Verbmobil: a Translation System for Face-to-Face Dialogs*. CSLI Lecture Notes; No. 33. CSLI Center for the Study of Language and Information, Stanford University.

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

M. Nagao, U. Tsujii, and U. Nakamura. 1985. The Japanese Government Project for Machine Translation. volume 11(2-3) of *Computational Linguistics*, pages 91–110.

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. In *Jounal of Semantics*, volume 10(2), pages 123–179.

L. Sadler and H. S. Thompson. 1991. Structural Non-correspondence in Translation. In *Proceedings of the 5th Conference of the European Chapter of the Association for Computational Linguistics (EACL '91)*, pages 293–298, Berlin, Germany.

J. Slocum, W. S. Bennet, J. Bear, M. Morgan, and R. Root. 1987. METAL: The LRC Machine Translation System. In S. Michaelson and Y. Wilks, editors, *Machine Translation today: the state of the art*, pages 319–350. Edinburgh University Press.

B. Vauquois and C. Boitet. 1985. Automated Translation at Grenoble University. volume 11(1) of *Computational Linguistics*, pages 28–36.

W. Wahlster. 1993. Verbmobil — Translation of Face-to-Face Dialogs. In *Proceedings of the 4th Machine Translation Summit*, Kobe, Japan.

P. Whitelock. 1992. Shake-and-Bake Translation. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING '92)*, pages 784–791, Nantes, France.