# THE MAPPING UNIT APPROACH TO SUBCATEGORIZATION

*Robert Bobrow, Robert Ingria, David Stallard*

BBN Systems and Technologies Inc.
10 Moulton Street, Mailstop 6/4C
Cambridge, MA 02138

## ABSTRACT

We present the "mapping unit" approach to representing subcategorization information, a computational framework for encoding subcategorization information which has been developed and implemented for BBN's DELPHI system (the NL component of the HARC spoken language system). The advantage of our approach to subcategorization lies in its flexibility, a flexibility which in turn offers greater robustness of coverage with respect to unanticipated variations of a verbal argument pattern, and easier extension of coverage to new patterns. It handles in a quite natural way argument order variation, optionality of arguments, and metonymy.

## INTRODUCTION

"Subcategorization" refers to the constraints that a verb (or other syntactic head) places on the type and relative order of the phrases that serve as its arguments and (by some definitions) the effect of these argument phrases on the meaning of the whole clause. It is thus an important issue for any system which seeks to understand natural language, or to distinguish grammatical natural utterances from ungrammatical ones.

Many past approaches have sought to represent subcategorization declaratively, often using an approach based on the unification of feature values. Such approaches as Definite Clause Grammar [11], Categorial Grammar [1], PATR-II [13], and lexicalized TAG [12] include in one form or another a notion of "subcategorization frame" that specifies a sequence of complement phrases and constraints on them. Some have also advocated using the feature system to encode semantic information (as for example [9]), and this has recently characterized our own approach [3].

"Mapping unit" subcategorization is partly inspired by these approaches, but it handles several kinds of variation in natural language utterances which cause difficulty for them. These forms of variation are not in any sense marginal phenomena, but are instead repeatedly seen in the naturally derived data for the ATIS SLS common task domain. The phenomena fall into three classes:

The first is variation in argument order, as seen in

fly from Denver to Boston
fly to Boston from Denver

Such variation can be handled by the frame approach, but only at the cost of specifying one frame for each order. Besides such lexically-specific variation of order, other sources of order variation include interpolation of elements traditionally considered adjuncts ("What flights leave at 3 pm from Denver") and heaviness effects ("Show on the screen the fares and departure times of all the flights from Boston to Dallas").

The second is the optionality of arguments, and the different consequences thereof, including zero anaphora:

"What restrictions apply?"
(= apply TO SOMETHING IN CONTEXT)

default value:

"Show the flights."
(= show the flights TO ME)

existential quantification:

"fly to Boston" (AT WHATEVER TIME)

and independent truth-conditions:

"The Rockettes kicked."

Each verb that has optional arguments tends to have different preferences for what to do with the omitted argument places, as the above examples make clear. The frame approach can handle them, but again only at the cost of specifying multiple frames.

The third and final type of variation is the metonymic coercion of arguments, as seen below:

"What wide-body jets serve dinner?"
(= "What FLIGHTS on wide-body jets serve dinner?"
aircraft themselves do not "serve meals")
"What airlines fly to Dallas?"
(= "What airlines HAVE FLIGHTS to Dallas?"
airlines themselves don't "fly")

In both examples there is a superficial clash of types which is meant to be reconciled through the interposition of an implicit binary relation between the objects having those types. Our work postulates a distinction between two kinds of metonymy: "referential", where the argument is taken to be an indirect reference to an object of the proper type, and "predicative", where only the argument slot of the predicate is coerced and the referent is taken literally. This distinction will be discussed in more detail below.

Most verbs in the ATIS corpora ("fly", "arrive", etc.) have flight, source, destination, time of day, and day of the week arguments, most of which are not obligatory and can occur in almost any order. The number of frames necessary is combinatorially impractical, and to this situation the phenomenon of metonymic coercion, which makes the variation potentially open-ended, only provides the final blow. A fundamentally different framework from that of subcategorization frames is needed.

## THE "MAPPING UNIT" INFORMATION STRUCTURES

The central idea of the mapping unit approach is that there are several different types of subcategorization constraints, which ought to be represented as separate constraints, rather than enumerating the "cartesian product" of their possible combinations in fixed patterns.

The basic building block is the "mapping unit", a structure which represents the constraints on a particular phrasal argument and the contribution this argument makes to the semantics of the clause. Mapping units do not "know" whether they are optional or not, or in what order they occur with respect to other mapping units; this information is instead represented in the grammar and in a larger structure called a "map", of which the mapping unit is a component.

The following is an example of a mapping unit:

```
SUBJECT
(NP :trans)
(FLIGHT :trans)
(= FLIGHT-OF :trans)
```

Each mapping unit has the four components shown here: a grammatical relation (SUBJECT, DIRECT-OBJECT, OTHER-PP etc), a syntactic pattern, a type requirement, and semantic role information. The syntactic pattern is a unification pattern, and thus retains all the advantages of being able to handle partial information that are associated with unification. The syntactic pattern (in the example just NP) also includes slots for semantic translation (":trans").

The semantic type requirement (here "FLIGHT") is also enforced by unification, but separately, so that a failure due to semantic type clash can be distinguished from one that violates syntactic constraints. In this way, if the type requirement is not met, the mapping unit can be metonymically coerced, filling the semantic role not with the original complement translation but with an indefinite object related to it via a binary relation that resolves the clash. For example, the unit above could be coerced to accept an object of type AIRLINE via the binary relation FLIGHT-AIRLINE-OF, which maps flights to their airlines, thus handling the utterance "What airlines fly to Dallas".

A final separate representation is the contribution the semantics of the argument makes to the semantics of the clause, which is indicated by a semantic role constraint. The role is set (with the equality symbol "=") in the case of ordinary complement arguments and restricted (with "<") in the case of iterable temporal or locative adjunct modifiers ("in Harvard Square at Out of Town News next to the foreign magazine section"). A semantic role can only be set once in any given clause (which of course does not exclude it from being set to a conjunctive element), but can be restricted arbitrarily many times.

The mapping units are combined in a larger structure called a "map", of which the following is a (much reduced) example for the verb "fly":

```
((FLY1 FLIGHT-OF FLIGHT
       ORIG-CITY CITY
       DEST-CITY CITY)
 SUBJECT
 (NP :trans)
 (FLIGHT :trans)
 (= FLIGHT-OF :trans)

 OTHER-PP
 (PP (FROM) (NP :trans))
 (CITY :trans)
 (= ORIG-CITY :trans)

 OTHER-PP
 (PP (TO) (NP :trans))
 (CITY :trans)
 (= DEST-CITY :trans)
 completion (AND (FILLED FLIGHT-OF DEST-CITY)
                 (FILLED-OR-ANAPHOR ORIG-CITY))
 translation (p-and (flight-dest FLIGHT-OF
                                 DEST-CITY)
                    (flight-orig FLIGHT-OF
                                 ORIG-CITY)
                    (flight-departure-time
                              FLIGHT-OF
                              TIME-OF-DAY))
```

Every map has four components:

1. a labeled-argument predicate with typed roles
2. a collection of "mapping units"
3. a completion condition
4. a translation rule for the labeled-argument predicate

The labeled-argument predicate—in the example 'FLY1'—is the representation of the verb's "meaning", and has an assigned set of typed semantic roles which can appear in any application of the predicate (but which are not necessarily required to appear in every application).

The completion condition must be satisfied by any complete clause with the verb as head, and includes requirements on the instantiation of semantic roles.[1] In the example map, the FILLED completion predicate requires that the roles FLIGHT-OF and DEST-CITY be filled by a literal argument to the verb, while the FILLED-OR-ANAPHOR completion predicate allows the role ORIG-CITY to be be implicitly filled by a discourse entity. This condition allows "What flights fly

[1] This requirement, in effect, implements the Functional Completeness Condition of LFG [4].

to Denver from Boston?", "What flights fly from Boston to Denver" and "What flights fly to Denver?" but forbids "What flights fly?".

Other completion predicates include FILLED-OR-DEFAULT, which specifies a default value for a role, FILLED-OR-EXISTS, which generates an existential quantification over the range type if the role is unfilled, and GRAMMAR-REL-FILLED, which requires that a particular grammatical relation have been assigned. The unqualified optionality of a semantic role is indicated simply by leaving it out of the completion conditions.[2]

The fourth map component, the translation rule, converts labeled-argument predicate applications into ordinary logic expressions based on the roles they instantiate. Its separation from the rest of the map avoids duplicate specification of the details of logical form construction.

This last point is important because a map can have multiple units on the same semantic role to represent multiple syntactic realizations of it. For example, the utterance "fly DENVER to Boston" (meaning "fly FROM DENVER to Boston") can be accomodated simply by adding the following unit to the map above:

```
OTHER-NP
(NP :trans)
(CITY :trans)
(= ORIG-CITY :trans)
```

and the function of the translation rule is exactly the same.

Any semantic role can be filled only once, so that overgeneration from multiple mapping units assigning the same role ("fly Boston from Denver", "fly Denver from Denver") is prevented.[3]

Certain grammatical relations can also be assigned only once in the derivation of any clause. These are the "major" grammatical relations —SUBJECT, DIRECT-OBJECT, INDIRECT-OBJECT. The OTHER-⟨cat⟩ relations can be assigned arbitrarily many times, subject only to the constraint that semantic roles be filled only once. Multiple mapping units that assign the same major grammatical relation are also allowed, subject only to the above constraint and of course to the constraint on the unique assignment of semantic roles. This is useful for handling certain types of polysemy—specifically, the semantic overloading of syntactic argument positions.

For modifiers which are normally treated as adjuncts, such as temporal or locative modifiers, our framework provides a notion of "free" mapping units associated with distinguished roles (such as TIME-OF-DAY, etc.). Such units do not have to be included in the map for individual lexical items whose labeled-argument predicate translation includes the role. In the example map above, TIME-OF-DAY is such a "free" mapping unit.

Finally, we should point out that while the map information structures can handle a considerable degree of variation, it is not necessary for any one map to handle all the possible variations associated with a verb. A verb can have multiple maps in the case of conventional lexical ambiguity, just as it can have multiple subcategorization frames in other approaches.

[2]Note that this assumption of optionality means that, as far as any given lexical item is concerned, any grammatical realization of a semantic role is optional, including SUBJECT. However, as is well known, English, unlike Italian or Greek, requires an overt subject in finite clauses. This requirement is imposed by the grammar, rather than by individual lexical items.

[3]This requirement, in effect, implements the Functional Consistency Condition of LFG [4]. This requirement and the requirement that completion predicates be satisfied, in effect, implements the Theta Criterion of GB theory [5].

## USE OF MAPPING UNITS IN GRAMMAR AND SEMANTIC INTERPRETATION

Standard phrase structure rules augmented with features [11] are in our approach further augmented with the non-constituent predicates AVAILABLE, VP-BIND, and COMPLETE-WFF, along with the selector CONSTIT. The following is an example of the grammar rule that assigns DIRECT-OBJECT, reduced to include only features of interest (non-constituent predicates appear in curly brackets):

```
(VP :MAP :BINDINGS2)
->
(VP :MAP :BINDINGS1)
{AVAILABLE DIRECT-OBJECT :MAP :BINDINGS1}
(NP :TRANS)
{VP-BIND DIRECT-OBJECT :MAP
         {CONSTIT (NP) (1)}
         :BINDINGS1 :BINDINGS2}
```

The predicate AVAILABLE takes a grammatical relation, a map, and a bindings list, which is a list pairing mapping units with role fillers. It is satisfied if there is a unit in the map with that grammatical relation such that the semantic role of that unit is not set in the bindings list, and the grammatical relation is not assigned in the bindings list (if it is a "major" grammatical relation).

The predicate VP-BIND takes a grammatical relation, a map, an entire constituent (retrieved by the function CONSTIT as seen above) an input bindings list and an output bindings list. If it succeeds, it produces a new bindings list containing an additional pair of unit and filler. It will succeed if it finds a free unit in the map that can be matched with the passed-in constituent both syntactically and semantically. VP-BIND will have more than one solution if it finds multiple units with these properties, in which case there are multiple parses. (Note that the AVAILABLE predicate is really only necessary to prevent the parser from looking for a constituent that would only wind up not being attachable to the VP.)

The pair of map and bindings effectively constitutes the meaning of the VP, and can be likened to a an application of lambda-expression (the map) to arguments (the bindings). The difference is that while the arguments to a regular lambda-expression can either be bound all at once or in some fixed order (e.g. through currying) the arguments to a map are referred to by label, and can be applied in any order we please.

Currently, the maps only provide optionality information, while the relative order of complements is enforced by the grammar via grammatical relations. This has the advantage that certain ordering constraints need only be stated once, as opposed to over and over again in map entries. An example of a rule imposing ordering constraints is the ditransitive VP rule, which handles "Show me the flights":

```
(VP :MAP :BINDINGS2)
->
(VP :MAP :BINDINGS)
{AVAILABLE INDIRECT-OBJECT :MAP :BINDINGS}
{AVAILABLE DIRECT-OBJECT :MAP :BINDINGS}
(NP :TRANS1)
{VP-BIND INDIRECT-OBJECT :MAP {CONSTIT (NP)(1)}
         :BINDINGS :BINDINGS1}
(NP :TRANS2)
{VP-BIND DIRECT-OBJECT :MAP {CONSTIT (NP)(2)}
         :BINDINGS1 :BINDINGS2}
```

The constraint that the subject precedes post-verbal complements is expressed by the clause-level S rule, which assigns the relation SUBJECT:

```
(ROOT-S (QUESTION) :MAP :BINDINGS2)
->
(NP :TRANS)
(VP :MAP :BINDINGS1)
{VP-BIND SUBJECT :MAP {CONSTIT (NP)(1)}
         :BINDINGS1 :BINDINGS2}
```

The completion conditions for the clause are enforced by the rule for the top-most node, START. This rule contains a predicate COMPLETE-WFF that takes a map, bindings list, and delivers an output formula:

```
(START (QUERY :WFF))
->
(ROOT-S (QUESTION) :MAP :BINDINGS)
{COMPLETE-WFF :MAP :BINDINGS :WFF}
```

COMPLETE-WFF enforces the completion conditions of the map and reduces the map and bindings combination to a formula if these conditions are satisfied.

The formula to be generated is specified by the translation rule component of the map. This translation rule can really be regarded as a kind of meaning postulate for the predicate that is associated with it directly. It consists of an ordinary logic expression containing references to the argument labels of the predicate. Repeated below is the translation for the predicate 'FLY1':

```
(P-AND (flight-dest FLIGHT-OF DEST-CITY)
       (flight-orig FLIGHT-OF ORIG-CITY)
       (flight-departure-time FLIGHT-OF
                              TIME-OF-DAY)
```

To generate the formula, the fillers of the argument roles are substituted for these references. The P-AND is a meta-conjunction operator with the property that if any of the role references of one of its conjuncts are unfilled, that conjunct is left out of the final formula. In this way we are not required to generate an existential quantification for a missing argument place (as for example the departure time of the flight).

There are certain instances in which an existential quantification is generated, however. If a semantic role has merely been restricted instead of filled, a narrow-scope existential quantification is generated and the variable of this quantification substituted for the role reference in the translation rule. Thus for "Flight 1 flies before 3 pm" we would have:

```
(exists t (precede t (time 3 0 pm))
          (flight-departure-time (flight-no 1)
                                 t))
```

## PREDICATIVE METONYMY

Another case in which narrow-scope existentials are generated is the case of predicative metonomy, in which the semantic role in question has been type-coerced to accept an argument of a type different from its restriction. In this type of metonymy, the referent of this argument does not change. Instead a relation is established between it and an indefinite, existentially quantified object of the proper type. Thus for "Delta Airlines flies from Boston to Baltimore" we would have:

```
(exists x flight (airline-of x Delta)
                 (and (orig-city x boston)
                      (dest-city x baltimore)))
```

The distinction between referential and predicative metonymy only becomes visible when the actual referents of NPs are sought, as in:

What airlines fly to Boston?
What wide-body jets serve dinner?

In the first, it is implausible that "airlines" is being used to refer to some set of flights, since every flight is on some airline and there is no constraint. In the second, "wide-body jets" is far more likely to refer to some set of flights, since not every flight is on a wide-body jet.

Predicative metonymy is an essentially local phenomenon, while referential metonymy is an essentially global one. Our present implementation assumes predicative metonymy only and allows only a limited set of binary relations. Processing is such as to prefer attachments that do not require metonymy, by assigning a lower probability at parse-time to parses which do require it [2]. This is necessary to exclude an unreasonable parse of:

Show flights to Denver on wide-body jets serving dinner.

## OTHER BENEFITS

The combination of labeled-argument predicate and translation rule offers several benefits not yet mentioned. One is that a given predicate can be shared between different lexical entries which provide different syntactic realizations of it. For example, in the ATIS domain the verbs "depart" and "originate" have very similar core meanings, yet have semantic roles realized by different prepositions:

The flight departs from Boston.
The flight originates in Boston.
*The flight departs in Boston.

The words are not synonyms in the normal sense that one can be substituted for the other in such a way as to preserve grammaticality. But their common semantic content can be represented.

Another advantage is that a denotational semantics with optionality is implemented without requiring Davidsonian-style event quantifications [7]. While event objects make sense in some contexts, having an existential quantification over events for every verb is frequently inconvienient in further processing. Certainly it is so in the ATIS domain, where the chief semantic outcome of clauses seems to be a set of predications on attributes of flight-individuals and there really are no "events" as such at all. Event quantifications are not precluded, however—they could be produced with a different translation rule schema.

## COMPARISON WITH OTHER WORK

Our treatment of the syntactic aspects of subcategorization is most like that of PATR-II [13]. Both PATR-II and the mapping units approach use recursive VPs, with each level of VP structure, in effect, "peeling off" a single constituent of the head verb's complement list.[4] A major difference between the two approaches is that the PATR-II system of subcategorization is essentially limited to popping constituents off the subcategorization list, in fixed order, requiring a separate subcategorization list for each variation in order. Our approach allows complements to be found in whatever order the grammar will allow them. In this respect, it is more like the UD system [8], which has an operator for "non-deterministic extraction from arbitrary list positions".[5] However, our system does not literally remove units from the map, but rather simply marks them as no longer available for binding. Moreover, the mechanism of allowing multiple syntactic realizations for a given semantic relation in a single representation of complement structure is, to our knowledge, unique.

Related work on the semantic aspects of argument optionality has been reported by Palmer [10], [6]. Our work differs from this mainly in the tighter coupling of syntax and semantics during processing and the use of recursive VP structures, which potentially allows for an elegant solution of cases of non-consituent conjunction, not addressed in the other work. Our system also has a finer grained treatment of optionality; whereas [10] and [6] divide argument roles into OBLIGATORY, ESSENTIAL, and NON-ESSENTIAL roles, we provide a richer set of possibilities. The decoupling of syntactic realizations of a role in a mapping unit from the semantic typing constraints on that role, to allow for different types of metonymic extension is also a distinction. On the other hand, the use of named thematic or semantic roles which cut across particular predicates in [10] and [6] might provide a more compact form for capturing linguistic generalizations, to the extent that such a theory of thematic relations is well-motivated.

## CONCLUSIONS AND FUTURE WORK

We have presented a framework for handling three unruly phenomena: argument-order variation, optionality, and metonymy. Yet this framework is simple and does not sacrifice formal goals such as declarativeness.

We are currently working on extending the mapping unit to other parts of speech such as nouns and adjectives. We are also working on more lexically-based ways to encode order, perhaps by a system of precedence among mapping units.

Other future work will include referential metonymy, the interaction of metonymy with anaphora, and non-constituent conjunction phenomena.

[4]Our approach allows more than a single complement constituent to appear at a given level, however, as the ditransitive VP rule above shows.

[5]Other systems which have a similar mechanism include the Lilog system of IBM Germany and the MiMo machine translation system.

## ACKNOWLEDGEMENTS

The work reported here was supported by the Advanced Research Projects Agency and was monitored by the Office of Naval Research under Contract No. N00014-89-C-0008. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the United States Government.

## REFERENCES

[1] Ades, A. E. and Steedman, M. J. "On the Order of Words", *Linguistics and Philosophy* 44.3, 1982, pp. 517–558.

[2] Bobrow, R. "Statistical Agenda Parsing" *Proceedings Speech and Natural Language Workshop February 1991* (this volume), Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1991.

[3] Bobrow, R., Ingria, R. and Stallard, D. "Syntactic and Semantic Knowledge in the DELPHI Unification Grammar". *Proceedings Speech and Natural Language Workshop June 1990*, Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1990, pp. 230–236.

[4] Bresnan, J. *The Mental Representation of Grammatical Relations*, The MIT Press, Cambridge, Massachusetts, 1982.

[5] Chomsky, N. (1981) *Lectures on Government and Binding*, Foris Publications, Dordrecht, Holland.

[6] Dahl, D. A., Palmer, M. S., and Passonneau, R. J. "Nominalizations in PUNDIT". *25th Annual Meeting of the Association for Computational Linguistics: Proceedings of the Conference*, Association for Computational Linguistics, Morristown, NJ, 1987, pp. 131–137.

[7] Davidson, D. "The Logical Form of Action Sentences". In N. Rescher, ed., *The Logic of Decision and Action*, University of Pittsburgh Press, Pittsburgh, 1967, pp. 81–95.

[8] Johnson, R. and Rosner, M. "A rich environment for experimentation with unification grammars". *Fourth Conference of the European Chapter of the Association for Computational Linguistics*, Association for Computational Linguistics, Morristown, NJ, 1989, pp. 182–189.

[9] Moore, R. C. "Unification-Based Semantic Interpretation". *27th Annual Meeting of the Association for Computational Linguistics: Proceedings of the Conference*, Association for Computational Linguistics, Morristown, NJ, 1989, pp. 33–41.

[10] Palmer, M. S. *Driving Semantics for a Limited Domain*. Doctoral Dissertation, University of Edinburgh, 1983.

[11] Pereira, F. C. N. and Warren, D. H. D. "Definite Clause Grammars for Language Analysis—A Survey of the Formalism and a Comparison with Augmented Transition Networks". *Artificial Intelligence* 13, 1980, pp. 231–278.

[12] Schabes, Y., Abeille, A., and Joshi, A. K. "Parsing Strategies with 'Lexicalized' Grammars': Application to Tree Adjoining Grammars". *COLING Budapest: PROCEEDINGS of the 12th International Conference on Computational Linguistics*, Association for Computational Linguistics, Morristown, NJ, 1988, pp. 578–583.

[13] Shieber, S. M. *An Introduction to Unification-Based Approaches to Grammar*. Center for the Study of Language and Information, Stanford, CA, 1986.