A Comparison of HPSG and LFG

Frank Richter and Valia Kordoni
Universität Tübingen

Frank Richter and Valia Kordoni
Seminar für Sprachwissenschaft
Kleine Wilhelmstraße 113
D-72074 Tübingen
Email: {fr,korder}@sfs.nphil.uni-tuebingen.de
WWW: http://www.sfs.nphil.uni-tuebingen.de/~{fr,korder}/

LFG: Nonlocal Dependencies

- early work within LFG on the characterization of long-distance dependencies was based on the then-standard assumption that the relationship between a “filler” and a “gap” was properly characterized in the c-structure;
- on analogy to $\uparrow$, which relates a c-structure node to its corresponding f-structure, Kaplan and Bresnan 1982 introduced the double arrows to relate a local f-structure to the f-structure of a more distant, displaced constituent;
- this was done for 2 reasons:
  1. first, the c-structure was thought to provide the proper vocabulary for the statement of constraints on the relation of a filler to a ap, and
  2. second, there was thought to be no way to finitely encode a potentially infinite long-distance dependency in f-structure terms

LFG: Nonlocal Dependencies (2)

- LFG, though, has evolved away from this initial c-structure encoding of long-distance dependencies;
- long-distance dependencies interact strongly with subcategorization restrictions, and a c-structure approach therefore meshes well with theories in which subcategorization is expressed in terms of constituent structure frames;
- but subcategorization in LFG is defined on the functional structure, by means of the Completeness and Coherence conditions on the appearance of governable grammatical functions;
- **thus**, the interactions of subcategorization and long-distance dependencies would be easier to account for if long-distance dependencies were also characterized in f-structure terms

LFG: Nonlocal Dependencies (3)

- *Definitions of Completeness and Coherence*
  1. An f-structure is *locally complete* if and only if it contains all the governable grammatical functions that its predicate governs. An f-structure is *complete* if and only if it and all its subsidiary f-structures are locally complete.
  2. An f-structure is *locally coherent* if and only if all the governable grammatical functions that it contains are governed by a local predicate. An f-structure is *coherent* if and only if it and all its subsidiary f-structures are locally coherent.
- *Grammaticality Condition*:
  A string is grammatical only if it is assigned a complete and coherent f-structure.

LFG: Nonlocal Dependencies (4)

- moving long-distance dependencies to the f-structure component led to a deeper consideration of the formal difficulty presented by the need to encode potentially infinitely many possibilities in constraints that must be embedded in a finite grammar;
- Kaplan, Maxwell, and Zaenen 1987 provided a finite encoding of an infinite set of possible f-structural relations by permitting expressions denoting regular languages to appear in place of simple attributes in f-structure constraints;
- they called this new descriptive mechanism *functional uncertainty*, since it can be used to indicate that one of a set of alternative grammatical function assignments is acceptable even though a particular function cannot be uniquely determined from locally available information;
- given this new formal device, it became possible to characterize relations like the relation between a "filler" and a "gap", or between an anaphor and its antecedent

"Outside-In" Functional Uncertainty: Filler-Gap Dependencies

- LFG does not require the presence of a "gap" at the c-structure level. The "gap" in a filler-gap dependency is an f-structure position that is filled by an f-structure which appears in a dislocated position in the c-structure;
- filler-gap dependencies are relations that are stated in f-structure terms, by the use of equations involving *functional uncertainty*: an infinite disjunction of possibilities is encoded in finite terms, by the use of a regular expressions
- the kind of *functional uncertainty* used to describe filler-gap dependencies is the so-called **outside-in functional uncertainty**: this involves the statement of a relation between two f-structures with respect to the *less* deeply embedded f-structure of the two

"Outside-In" Functional Uncertainty: Filler-Gap Dependencies (2)

- the regular expression COMP* OBJ, which involves the Kleene closure operator "*", stands for a an infinite disjunction of paths within f-structures: paths involving zero or more COMPS followed by the attribute OBJ. This equation, then, identifies the TOPIC of the sentence with the value of any one of these paths;
- in general, the position occupied by an atomic attribute in an expression such as (f OBJ) can also be occupied by a path expression such as (f COMP OBJ). The f-structure referred to is the one found at the end of that path in f. The same position can also be occupied by a regular expression indicating a disjunction over a (possibly infinite) number of paths consisting of zero or more occurences of COMP followed by one occurence of OBJ. Such expressions are said to involve *functional uncertainty*, since they encode an uncertainty over possible positions in the functional structure

"Outside-In" Functional Uncertainty: Filler-Gap Dependencies (3)

- outside-in functional uncertainty can be schematically represented as in the following:
  (f_1 GF1) = (f_1 GF2 GF3 GF4 GF5)
- this equation states that the f-structure f_1 contains a singleton path GF1 whose value is also the value of the path GF2 GF3 GF4 GF5. The result is that the less embedded GF1 is functionally identified with the more embedded GF5

“Inside-Out” Functional Uncertainty: Anaphoric Binding

- an alternative way of achieving the same result involves *inside-out* functional uncertainty: this involves the statement of a relation between two f-structures with respect to the *more* deeply embedded f-structure of the two
- we might associate an expression involving functional uncertainty not with the f-structure f_1, but with the one labeled f_5:
  ((Gf2 GF3 GF4 GF5 f_5) GF1) $= f_5$
- essentially, the idea is that GF5 looks “outward” for a higher GF1 to identify itself with
- the path expression Gf2 GF3 GF4 GF5 appears here *in front of* the f-structure identifier f_5, rather than after it. Prefixing a path expression to an f-structure identifier such as f_5 picks out a structure through which that path leads to f_5: in this case, the outermost f-structure;
- any f-structure expression can be followed by the name of an attribute: for example, (f SUBJ) denotes the f-structure that is the value of the attribute SUBJ in the f-structure f. Since the expression
  (Gf2 GF3 GF4 GF5 f_5)
  also denotes an f-structure, it is possible to specify an attribute within it, just as in the simple case. ((Gf2 GF3 GF4 GF5 f_5) GF1) denotes the value of GF1 within (Gf2 GF3 GF4 GF5 f_5), the outermost f-structure;

- in our example, the path prefixed to the f-structure identifier does not contain a regular expression. In fact, just as in the case of outside-in functional uncertainty, it is possible to prefix any regular expression to a f-structure identifier, picking out any f-structure through which there is a path in the language of the regular expression leading to the f-structure;
- in this way, inside-out functional uncertainty provides a way of relating two f-structures. In particular

“Inside-Out” Functional Uncertainty: Anaphoric Binding (2)

- it is possible to relate the f-structure of an anaphor to that of its antecedent by means of inside-out functional uncertainty, using an expression lexically associated with the anaphor that picks out a set of less embedded f-structures: those that must be the antecedent of the anaphoric element, or those with which the anaphor may not corefer;
- in characterizing anaphoric binding constraints, Dalrymple 1993 refers to the path prefixed to an f-structure identifier as the DomainPath, and the path through the containg f-structure as the AntecedentPath:
  ((DomainPath $\uparrow$) AntecedentPath)
- such expressions appear as components of *binding equations* enforcing syntactic constraints on anaphoric binding;
- these equations are lexically associated with individual anaphoric elements: $\uparrow$ is the f-structure for the anaphoric element;
- intuitively, the DomainPath is a path through some set of f-structures that ends at $\uparrow$;
- the f-structure (DomainPath $\uparrow$) is, then, the set of all f-structures that contain the anaphor at the end of some DomainPath;
- the AntecedentPath is a path through one of those f-structures leading to a potential antecent for the anaphor

“Inside-Out” Functional Uncertainty: Anaphoric Binding (3)

- since (DomainPath $\uparrow$) is the f-structure that contains the anaphor and its antecedent, constraints on the DomainPath determine the domain of binding or disjointness of the anaphoric element;
- if, for instance, an anaphor obeys the Minimal Complete Nucleus Binding Condition (cf., Dalrymple 1993), its antecedent must appear in the minimal f-structure containing the anaphor and a subject. This constraint is enforced by placing conditions on the DomainPath: the DomainPath is required **not** to pass through an f-structure with a SUBJ

"Inside-Out" Functional Uncertainty: Anaphoric Binding (4)

- AntecedentPath determines the grammatical function borne by the antecedent of the anaphor;
- constraints on the form of the AntecedentPath determine, then, the permissible or impermissible grammatical function of its antecedent: for example, AntecedentPath might be required to be the singleton SUBJ, enforcing the requirement for SUBJ antecedents

"Inside-Out" Functional Uncertainty: Anaphoric Binding (5)

- the f-structure of the anaphor and its antecedent are not the same: the anaphor is syntactically autonomous, having its own structure;
- in this way, anaphoric binding is unlike the phenomenon of filler-gap dependencies, in that syntactic identity between the anaphor and its antecedent is not enforced;
- rather, the relation between an anaphor and its antecedent is semantic in nature. In the simple cases, the two are coreferent:
  $((\text{DomainPath } \uparrow) \text{ AntecedentPath})_\sigma = \uparrow_\sigma$

References

Dalrymple, M. (1993). *The Syntax of Anaphoric Binding*. CSLI Publications.

Kaplan, R. M. and J. Bresnan (1982). Lexical Functional Grammar: a Formal System for Grammmatical Representation. In J. Bresnan (Ed.), *The Mental Representation of Grammatical Relations*, pp. 173–281. MIT Press:Cambridge, MA.

Kaplan, R. M., J. T. Maxwell, and A. Zaenen (1987). Functional uncertainty. CSLI Monthly Newsletter.