Refining the Semantics of Lexical Rules

David Lahm, University of Frankfurt (lahm@uni-frankfurt.de)

1 Introduction

We argue that the semantics for lexical rule specifications (LRSs) proposed by Meurers (2001) does not adequately capture certain intuitions about what these specifications should mean. A lexical rule may constrain the range of its possible inputs far more severely than one would expect on the basis of its specification. In extreme cases, a rule with a satisfiable LRS may even become unsatisfiable (i.e. describe nothing and thus licence no new words). We argue the problem to be that properties of attribute paths are transferred in isolation, disregarding the effects such transferrals may have on the properties of other paths, and propose an alternative semantics that is based on transferring the properties of maximal jointly transferrable sets of paths and sensitivity to the possible inputs to the rule and which can be shown not to suffer from the problems mentioned.[1]

The lexical rules in the sequel will presuppose the toy sort-hierarchy in figure 1, derived from that given in Meurers (2001), whose sole purpose is to provide a simple basis for this investigation.

2 Specifying Lexical Rules

Commonly, lexical rule specifications (LRSs) are given as two AVMs separated by the symbol "$\mapsto$": $A \mapsto B$. A describes the licit input objects while B describes what the output is supposed to look like. Following Meurers (2001), $A \mapsto B$ can be understood as a notational convention abbreviating

$$\begin{bmatrix} \textit{lex\_rule} & \\ \text{IN} & A \\ \text{OUT} & B \end{bmatrix}$$

So lexical rules, like lexical entries, licence objects in the models of grammars. But the output of a rule is expected to be related to the input in ways not explicitly specified in B. The output objects should have as much in common with the input objects as possible while still conforming to B. A semantics for LRSs must hence explicate this transferral of unmentioned properties.

In what follows, we shall distinguish between those objects which a full lexical rule (with all transfers in place) is to licence and those which would be licenced by its LRS before any transferrals have been put in place. We call the former *instances*, the latter *proto-instances (PIs)* of the rule.

We assume that the following criterion should be met by any semantics for lexical rules.

Criterion (Universality (of possible input)). *For every PI of the rule, there is an instance such that the values of the* IN *attributes on the PI and the instance are congruent.*[2] *Hence the LRS alone determines whether a given object is a possible input to a given rule.*

Intuitively, this requires a rule to have something to say about each object that can serve as an input object according to its specification. It seems that this is what linguists have in mind when putting down a lexical rule, but it will become clear that Meurers' semantics does not fulfill it.

3 The Semantics of (Meurers, 2001)

Meurers (2001) considers two types of transfers: sort- and value transfers. Sort transfers are to keep the sorts of corresponding paths[3] the same wherever possible. They are performed by letting no PI become an instance if it does not transfer the species of some path while some other does.

[1] Like Meurers' semantics, the one put forth here will ultimately be formulated in an indirect manner by specifying a translation from LRSs to the actual descriptions of the rules, but in order to convey the main idea, it seems more profitable to leave out the intermediate layer and talk about the denoted objects directly. We shall further not deal with the special symbols $\sharp$ and $\flat$ introduced by Meurers. These can easily be incorporated into our semantics but would only complicate the discussion here. The same holds for specifications of path inequalities, which are absent from the rules discussed here for the sake of exposition but which can also be incorporated into the approach.

[2] Two objects are congruent if they are look-alikes. They have values of the same maximal sort on all paths and the values of two paths are identical in one of them iff they are in the other.

[3] Two paths correspond if they are of the shapes INπ and OUTπ.

More precisely, the existence of a PI x may prevent a PI y from becoming an instance if for some path π mentioned in the LRS, $\textsc{in}\pi$ has the same species on x and y and $\textsc{out}\pi$ also has this species on x but not on y. In x, the species of π is said to be *transferred* while in y it is not. Ruling out every PI that does not transfer it clearly guarantees that it will be transferred in all instances.

Transferral of values is realized by requiring that only such PIs may become instances in which $\textsc{in}\pi\alpha$ and $\textsc{out}\pi\alpha$ are token-identical (for any path π and attribute α), given that $\textsc{out}\pi$ is mentioned in the rule specification but $\textsc{out}\pi\alpha$ is not and provided that the species of of $\textsc{out}\pi$ and $\textsc{in}\pi$ allow for identical values of α. So for every path mentioned in the LRS, its possible unmentioned extensions by one attribute are required to be identical in the in- and output.

4 Problems with Meurers' Semantics

Consider the seemingly rather trivial rule in (1).

(1) $$\begin{bmatrix}\textsc{u} & g\end{bmatrix} \mapsto \begin{bmatrix}\textsc{u} & \begin{bmatrix}h & \\ \textsc{k} & bool \\ \textsc{l} & bool\end{bmatrix}\end{bmatrix}$$

In each PI, the values of U K and U L must both be (of species) + in the input. Regarding the output, every PI is such that U K is + and U L is − or *vice versa*. As a consequence, no PI can become an instance: PIs in which U K is + in the input and − in the output are ruled out due to PIs in which U K is + in both input and output. But these in turn rule out those in order to transfer U L. Since the rule clearly has PIs and hence possible input values, the requirement of Universality is violated, and it even becomes altogether unsatisfiable.[4]

The way in which Meurers transfers path values leads to similar problems in a variety of configurations. Quite generally, there might be PIs on which $\textsc{in}\pi$ has a value whose species allows a wider range of species for the value of an attribute α than does that of $\textsc{out}\pi$. But if there also is, say, a single species s that both of them accept, the value of $\pi\alpha$ must be transferred, narrowing the possible species of input objects down to s.

More concretely, another problematic case is exemplified by the following rule.

(2) $$word \mapsto \begin{bmatrix}\textsc{x} & \boxed{1} \\ \textsc{y} & \boxed{1}\end{bmatrix}$$

The values of all attributes that are defined on the values of X and Y will be transferred to the output. Since X and Y must be token-identical in the output, so must be all paths defined on them. It follows that transferring e.g. X K and Y K, 'backwards' implies the token-identity of these paths in the input value. Differently from what one would expect, thus, this rule only accepts input values in which the values of each common attribute on X and Y are identical. This is again a violation of Universality.

5 The Refined Semantics

The basic idea that we employ to solve these problems is the same for the transferral of species as for that of path equalities: while Meurers treats each path in isolation and transfers its properties at any cost, we shall concern ourselves with maximal sets of paths with jointly transferrable properties. In the cases considered above, this will result in the possibility of 'choosing' which properties to transfer. If transferring them all is not possible, any maximal consistent subset of them is allowed.

To make this precise we determine, for each PI of a given rule, the set of all relevant corresponding paths whose species or values are identical. Call the former $SFrame(x)$ and the latter $VFrame(x)$ for any PI x:

(3) a. $SFrame(x) = \{\pi \in Mentioned(LRS) \mid \textsc{in}\pi \text{ and } \textsc{out}\pi \text{ have the same species on } x\}$

[4] A violation of universality without unsatisfiability is obtained by replacing g with l in the rule. The rule would then have instances, but reject any input with a U-value of species g, for the reasons given above. Also, as an anonymous reviewer points out, the translation algorithm Meurers gives as a specification of his semantics would not yield an unsatisfiable result but instead one that is not uniquely determined by the rule. The semantics I am presenting is what I believe the algorithm was supposed to derive.

b. $VFrame(x) = \{\pi \in Edge(LRS) \mid \text{IN}\pi, \text{OUT}\pi$ are def. and have the same value on $x\}$

The paths in (3a) are drawn from the set $Mentioned(LRS)$ of all paths explicitly mentioned in the LRS. Those in (3b) are drawn from $Edge(LRS) = \{\pi\alpha \mid \text{OUT}\pi \in Mentioned(LRS)$ & $\text{OUT}\pi\alpha \notin Mentioned(LRS)\}$, the set of unmentioned extensions of mentioned paths by one attribute.[5]

Like Meurers, we will need to keep certain PIs from becoming instances of the rule considered in order to realize the desired transfer of properties. In order to guarantee Universality for our approach, we will allow ruling out any PI only if there exists one that transfers more properties and has an IN-value congruent with that of the one ruled out. We define:

Notational Convention (Input Value Congruence). $IVC(x, y)$ *iff the values of* IN *on* x *and* y *are congruent.*

Definition 1 (Species Transfers).
$STrans(PI) = \{x \in PI \mid For\ no\ y \in PI : IVC(x, y)\ and\ SFrame(y) \supset SFrame(x)\}$

Definition 2 (Value Transfers).
$VTrans(PI) = \{x \in PI \mid For\ no\ y \in PI : IVC(x, y)\ and\ VFrame(y) \supset VFrame(x)\}$

Let $PI(\lambda)$ denote the set of proto-instances of a lexical rule specification λ.[6] $STrans(PI(\lambda))$ is the set of all proto-instances x of λ such that no proto-instance with an IN-value congruent with that of x exists that transfers the species of a proper superset of the paths whose species are transfered in x. $VTrans(PI(\lambda))$ is the analogous notion for value transfers. Since there will clearly exist maximal elements wrt $\supset$ (note that both $SFrame$ and $VFrame$ are finite), these sets are guaranteed to be non-empty and to contain, for every proto-instance of λ, some element with a congruent IN-value. So Universality is clearly respected by $STrans$ and $VTrans$.

The set of instances of the rule is given by

(4) $Transfers(PI(\lambda)) = VTrans(PI(\lambda)) \cap STrans(PI(\lambda))$

Does $Transfers$ still respect Universality? One can show that

$$STrans(VTrans(PI(\lambda))) = VTrans(STrans(PI(\lambda))) = VTrans(PI(\lambda)) \cap STrans(PI(\lambda))$$

So the order in which the transfers are performed is immaterial and, since each of the transfer operations respects Universality, so does $Transfers$ itself.

6 Application

Consider again rules (1) and (2). Regarding the former, some of its PIs will have the $SFrame$ $\{$U K$\}$ and some will have the $SFrame$ $\{$U L$\}$. These do not stand in the subset relation and so no proto-instance can be dropped to frame one of the two paths.

If the sort-hierarchy allowed both $+$ and $-$ on the species subsumed by h, PIs would exists in which both U K and U L are $+$. Their $SFrame$ would be $\{$U K, U L$\}$. Hence they would rule out any PI which has $-$ on any of these paths, and the pertinent properties would be fully transferred.

Regarding rule (2), consider a proto-instance on which, e.g., X K and Y K have different values. This will preclude transferring the values of both paths together. It is still possible to transfer the value of one of them, so there will be instances on which the input value of X K is the output value of both X K and Y K and others on which this holds for the input value of Y K.

7 Conclusion

We have argued that the semantics of lexical rules proposed in Meurers (2001) has counterintuitive properties. The possible input values to lexical rules can come out as far more constrained than the specification of the rules would lead one to expect. We have identified the problem in the semantics' strictly isolating transferral of path properties and its insensitivity to the shapes of possible inputs to the rule and proposed an alternative which can be shown to overcome it.

[5] We follow Meurers' decision only to transfer values of paths in $Edge(LRS)$. This is is not a necessary move under our approach, but discussing the merits of possible alternatives is not our present concern.

[6] Strictly speaking, this 'set' is of course a proper class. It is possible however to find sets which contain, for any PI, a congruent object. Since the actual formalization of the ideas laid out here proceeds indirectly (cf. footnote 1), operating on descriptions instead of the objects themselves, there is no reason to worry here.

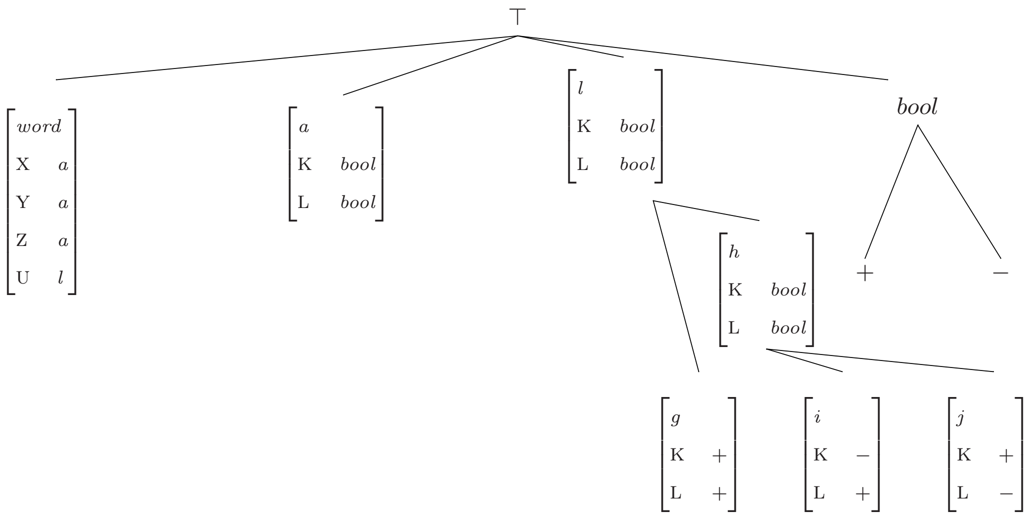

Figure 1: The Sorthierarchy.
The edges signal subsumtion. Where a sort does not narrow down the acceptable sorts for an attribute appropriate to a subsuming sort, the attribute is not repeated. The sorts lowest in the graph are assumed to be maximally (or minimally, depending on the favoured parlance) specific, i.e. species.

References

Meurers, Walt Detmar. 2001. On Expressing Lexical Generalizations in HPSG. *Nordic Journal of Linguistics* 24:161–217. URL `http://www.sfs.uni-tuebingen.de/~dm/papers/lexical-generalizations.pdf`.