Using Templates to Account for English Resultatives

Elizabeth MacGregor

February 14, 2010

The English resultative construction is a secondary predication relationship in which the main verbal predicate is affected by the existence of a resultant state. This resultant state describes the outcome of the action which was performed on the patient of the main predicate. In this study I will attempt to describe a manner in which the resultative can be formalized through the use of LFG templates and Glue semantics.

For the sake of expedience during this treatment of the resultative, we will focus on two types of resultatives, (1) transitive resultatives, and (2) unaccusative resultatives (demonstrated below), although several other types can be identified.

(1)   I dyed my skirt red.

(2)   The river froze solid.

LFG templates, as presented in [ADT08], act as a bridge between syntactic form and semantic meaning. This is because the the lexical entries for verbs are associated with a given argument relationship, as represented by a statement in Glue Semantics. The lexical entry calls a structural form by default through a template call. This structural and semantic arrangement can be overwritten by a template which only calls to the original argument name, and then changes its syntactic form and Glue equation. So, given the structural rules for transitive (3) and intransitive (4) below, one can utilize the verbal predicates given in a lexical entry to generate any transitive or intransitive sentence, without the lexical entry having to specify the argument structure, as demonstrated in 5. A template for a more complex structure would then call for the function name which can be paired with a certain structure (TRANSITIVE or INTRANSITIVE). This template call will specify new values for both the argument structure and semantic relationships.

(3)   TRANSITIVE(FN)   = $(\uparrow PRED) = `FN < (\uparrow SUBJ)(\uparrow OBJ) >'$

(4)   INTRANSITIVE(FN)   = $(\uparrow PRED) = `FN < (\uparrow SUBJ) >'$

(5)
$$\left(\begin{array}{ll} \text{hammer} \quad \text{V} & \lambda e.hammer(e) : (\uparrow_\sigma REL) \\ & \text{@TRANSITIVE(hammer)} \\ & \lambda P \lambda x \lambda y \lambda e.P(e) \wedge agent(e) = x \wedge patient(e) = y : \\ & (\uparrow_\sigma REL) \multimap (\uparrow SUBJ)_\sigma \multimap (\uparrow OBJ)_\sigma \multimap \uparrow_\sigma \end{array} \right)$$

With these basic rules and templates in place, we can look at the more complex structure of the resultative. Both of the kinds of resultative we are looking at take the meaning of the function, and apply it to a new verbal frame and Glue equation. Using a template hierarchy extended from [ADT08] (shown as 6), the English Resultative becomes an extension of the use of the RESULT-MEANS template, and inherits its properties and meanings. The two kinds of resultative that we are concerned with then inherit and further specify the ENG-RESULT template (7), forming the final templates of RESULT-T (transitive) and RESULT-U (unaccusative), shown below as 8 and 9, respectively.

(6)   Extended Template Hierarchy

```
           MEANS                              TRANSITION        MANNER
                      RESULT-MEANS                 TRAVERSAL
        ENG-RESULT           DUTCH-TLC      TRAVERSAL-MEANS   TRAVERSAL-MANNER
   RESULT-T  RESULT-F  RESULT-U      SWEDISH-DMC  COEXTENSIVE-MEANS  ENGLISH-WAY
                                              DUTCH-WEG
```
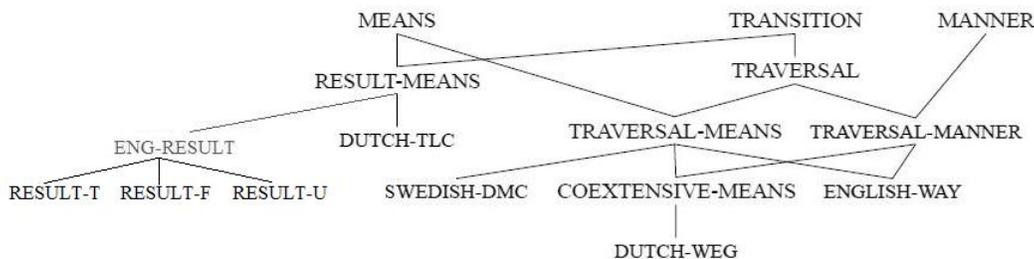
$$
\begin{array}{lll}
\text{ENG-RESULT} & = & \text{@RESULT-MEANS} \\
& & \lambda R \lambda s. R(s): \\
& & [(\uparrow_\sigma \text{EVENT2}) \multimap \uparrow_\sigma] \multimap [(\uparrow_\sigma \text{EVENT2}) \multimap \uparrow_\sigma]
\end{array}
$$
(7)

$$
\begin{array}{lll}
\text{RESULT-T(FN)} & = & \text{@ENG-RESULT} \\
& & \text{@TRANSITIVE-X(FN)} \\
& & \lambda R \lambda P \lambda e \lambda s \lambda y. P(e) \wedge R(s) \wedge agent(e) = \neg y \wedge patient(e) = y \\
& & \quad \wedge\, experiencer(s) = y \text{:} \\
& & (\uparrow \text{SUBJ})_\sigma \multimap (\uparrow \text{OBJ})_\sigma \multimap (\uparrow \text{XCOMP})_\sigma \multimap (\uparrow_\sigma \text{EVENT1}) \multimap (\uparrow_\sigma \text{EVENT2}) \multimap \uparrow_\sigma
\end{array}
$$
(8)

$$
\begin{array}{lll}
\text{RESULT-U(FN)} & = & \text{@ENG-RESULT} \\
& & \text{@INTRANSITIVE-X(FN)} \\
& & \lambda R \lambda P \lambda e \lambda s \lambda x. P(e) \wedge R(s) \wedge agent(e) = x \wedge patient(e) = x \\
& & \quad \wedge\, experiencer(s) = x \text{:} \\
& & (\uparrow \text{SUBJ})_\sigma \multimap (\uparrow \text{XCOMP})_\sigma \multimap (\uparrow_\sigma \text{EVENT1}) \multimap (\uparrow_\sigma \text{EVENT2}) \multimap \uparrow_\sigma
\end{array}
$$
(9)

Finally, the resultative is combined into the syntax by the c-structure rules in 10 and 11. These rules can only be used if the resultative interpretation is available to the verb, and create the extra XCOMP argument which takes the secondary predication.

(10)
$$
\begin{array}{llll}
\text{V'} & \rightarrow & \text{V} & \text{NP} & \{NP|AP|PP\} \\
& & \uparrow = \downarrow & (\uparrow OBJ) = \downarrow & (\uparrow XCOMP) = \downarrow \\
& & & & (\downarrow SUBJ) = (\uparrow OBJ) \\
& & & & @RESULT - T((\uparrow PRED\ FN))
\end{array}
$$

(11)
$$
\begin{array}{lll}
\text{V'} & \rightarrow & \text{V} & \{NP|AP|PP\} \\
& & \uparrow = \downarrow & (\uparrow XCOMP) = \downarrow \\
& & & (\downarrow SUBJ) = (\uparrow SUBJ) \\
& & & @RESULT - U((\uparrow PRED\ FN))
\end{array}
$$

When everything is combined, the resultative is created because the lexical entry permits the optional use of the resultative template, the resultative template calls in the extra verbal argument, and the c-structure rule provides the extra constituent. From the semantic side, the Glue equation combines the meaning of the subject with the meaning of the XCOMP and the meaning of the two events to create the overall meaning of the sentence.

References

[ADT08] Ash Asudeh, Mary Dalrymple, and Ida Toivonen. Constructions with lexical integrity: Templates as the lexicon-syntax interface. In Miriam Butt and Tracy Holloway-King, editors, *Proceedings of the LFG08 Conference*. CSLI Publications, 2008.