A structural theory of derivations:
adding structural change data to a formal grammar

Zach Stone
University of Maryland, College Park, Linguistics
zstone@umd.edu

INTRODUCTION
- We propose a model of derivations which keeps track of the derived syntactic objects (DSOs) occurring in a derivation, but also the structural changes (SCs) applied in it.
- We develop the theory in a very general setting which can accommodate complex dependency structures used in morphosyntactic theories.
- This improves over models which do not include this data as part of their formal structure.

SYNTACTIC RELATIONS
- Formal minimalist grammars (MGs) in the tradition of [4] view a language as a set of expressions, together with a set of grammatical rules.
- Expressions of trees or dependency structures with assignments of features to parts of the derived objects ([1], [4], [5]) and sequences of strings of pronounced symbols with assignments of features ([4], [5]) have been described.
- Rules take in tuples of expressions, and return an expression, as in the figure below.

Though there is an “obvious” pair of functions from each set of nodes of the input trees to the set of nodes of the output tree (which also preserves certain feature and projection data), this information is NOT part of the formal structure of the model.
- This can be contrasted with Algebraic Graph Grammars (AGGs) [3], where the operations are associated to morphisms: explicit mappings which compare nodes of the input objects to nodes in the output object. We refer to a tuple of such morphisms associated to an instance of an operation as the SC data.
- Morphosyntactic theories like nanosyntax, such as the model sketched in [2], indicate how features can be represented in the DSOs themselves, where the DSOs are dependency structures.

REFERENCES

RULES
- We enrich the rules with SC information.
- This allows us to develop a theory of structural contexts that a rule applies in.
- Let R be the rule which assigns to each pair of rooted DSOs (X,Y) the DSO with a dependency added from the root of X to the root of Y.
- A pair of A-morphisms \( (f_s \cdot f_r) : (X,Y) \rightarrow (X',Y') \) preserves conditions for G” if the pushout of every G-SC on (X,Y) along \( f_s \cdot f_r \) is a G-SC on (X,Y). The condition category for R is the collection of root-preserving maps.
- If a rule is generated by some finite set of basic operations applied in some admissible context, the rule reduces to the SPO in AGGs. R is generated by the basic operation above.

EQUIVALENCES
- The categorical structure of ADer induces “good” definitions of isomorphisms and substructure inclusions.
- Isomorphic derivations have the same syntactic relations like projection, agreement, etc.
- Isomorphisms “trickle up” to “strong extensional equivalence of languages”. Two languages L and M are equivalent if they have isomorphic DSOs, and for every SC applicable to some tuple of derivations in L, an isomorphic DSO in M is possible with an isomorphic tuple of derivations.

BACKGROUND
- We develop the theory in a very general setting which can accommodate complex dependency structures used in morphosyntactic theories.
- Pullbacks of constituents give information about structural contexts used in morphosyntactic theories.

PROPOSAL
- For any A concrete over FPos, U:A→FPos, we give a category of derivations ADer
- An A-derivation is essentially a finite partial order, together with an assignment of A-object (DSO) to each point x and A-morphism (SC) to each order relation \( x \leq y \).
- Morphisms between A-derivations are given by order-preserving maps between underlying partial orders together with A-morphisms for each DSO in correspondence.
- Assumptions are made about (A,U) to ensure that certain recursive constructions are possible. Given a base set of lexical items and rules (enriched with SC data), we iteratively form derivations by ‘extending’ a tuple of derivations to a new DSO along the specified SCs.

MORPHISM: explicit mappings which compare nodes of the input objects to nodes in the output object. We refer to a tuple of such morphisms associated to an instance of an operation as the SC data.

Morphosyntactic theories like nanosyntax, such as the model sketched in [2], indicate how features can be represented in the DSOs themselves, where the DSOs are dependency structures.

SYNTACTIC RELATIONS
- The SCs below indicate AGREEMENT relations. Iterative applications of the same rule produce this structure.
- These kinds of relations can be described using...

PULLBACKS, generalizations of intersections. Pullbacks of constituents give information about dependencies introduced “over” SCs. Most core grammatical relations can be expressed in terms of this data.