THE (PUTATIVE) LIMITS OF INHERITANCE IN CONSTRUCTIONIST GRAMMAR THEORIES

Timm Lichte (Tübingen) & Laura Kallmeyer (Düsseldorf)

ICCG 11 Workshop “Constructional approaches in Formal Grammar” , 20.08.2021
AIM OF THIS TALK

- **Motivation**: Constructionist theories of grammar have been criticized for their exclusive use of *inheritance* when attempting to capture the relationships between constructions.

- In this talk, we argue that inheritance generally suffices within an **EDL framework**, if it provides *sufficient flexibility* to describe and constrain syntactic representations.

- Using **TAG and metagrammars**, we will demonstrate this for
  - active passive alternation in combination with resultative constructions

- Using a new and more flexible EDL formalism, **TUCO**, we will look at
  - coordination of unlikes (involving benefactives and ditransitives)
EDL VERSUS BDL

• Let’s say we have a syntactic tree – be it flat or binary.

• There are two very general, but fundamentally different ways a theory can deal with its structure and meaning.
  1) Bounded Domain of Locality (BDL)
  2) Extended Domain of Locality (EDL)
1) Bounded Domain of Locality (BDL)
   - Grammar rules over **smallest** subtrees
   - Challenge: **Where am I?**
     - Need for a “memory” in the nodes for orientation purposes → valency list / slash list → “potential structure” (Müller 2019b)
     - Need for something that contributes valency → head
     - Need for the distinction between complements and adjuncts (because the VL must be finite)
     - Tendency towards binary structures (also driven by the idea that structures reflect Curried functor-argument combinations)
   - This is sometimes called the **lexicalist way** of doing grammar.
   - Basic formalisms: CFG, Categorical Grammar
EDL VERSUS BDL

2) Extended Domain of Locality (EDL)
   - Grammar rules over *arbitrary* subtrees
   - Challenge: **What am I?**
     - No need for a memory in the nodes for orientation purposes → no valency list → “actual structure” (Müller 2019), aka. "usage-based"
     - Tendency towards flat or non-binary trees
     - Need to capture the nature of and relationship between subtrees
       - by inheritance or rewriting
       - by lexicalization (e.g. as in LTAG)
   - We will call this the **constructionist way** of doing grammar.
   - Basic formalisms: tree rewriting grammars such as TAG
EDL VERSUS BDL

• Which one is better?
• Wrong question! We're at the level of formalisms, not theories!
• Rephrase: Which one enables more correct, comprehensive, "intuitive"/"elegant", and manageable theories?
  – First difficulty: infinitely many possible theories that can be compared
  – Second difficulty: lack of work that outlines the potentials of the EDL approach
• Examples of limited EDL such as TAG have lead to misunderstandings → see, e.g., Müller (2019a, 2019b)
• We think EDL is better than its reputation, in particular unlimited EDL.
• But first take a look at TAG.
TREE-ADJOINING GRAMMAR (TAG)

- A TAG consists of a finite set of **elementary trees (ETs)** that are combined into larger trees with two operations:
  - **Substitution**: rewriting of leaves
  - **Adjunction**: rewriting of nonterminals
- More powerful than CFG → mildly context sensitive
SHAPE OF ELEMENTARY TREES

- **What am I?** What is the shape and function of an ET?
- **XTAG standard:** Lexicalized TAG + well-formedness conditions
  - Every ET has at least one "lexical anchor".
  - The lexical anchor determines the structure of the ET.
  - Verbal ETs correspond to a linearization ("real structure") of the associated valency list.
  - ETs are grouped into tree families that correspond to valency lists.
- But that's just one choice.
- At any rate, the *metagrammar* must be seen as an integral part of any serious theory based on TAG.
TAG AND METAGRAMMARS

- ETs can be arbitrarily large → indeed good for modelling long-distance dependencies and idioms/MWEs.
- But how to express lexical generalizations?
- **Metagrammars** help factorizing elementary trees and representing relations between elementary trees, for example valency alternation (active-passive alternation) or linearization options (base order, extraction).
- The building blocks of metagrammars are **labeled descriptions** of tree fragments, that can be combined and reused within a metagrammar to generate **unlexicalized elementary trees** (tree templates).
TAG AND METAGRAMMARS

- Descriptions refer to (among others)
  - immediate/non-immediate dominance (\(\rightarrow\))
  - immediate/non-immediate precedence (\(\preceq\))
  - identity (=)
  - connected with conjunction (\(\land\)) or disjunction (\(\lor\))

- Tree templates are **minimal models** of tree descriptions (Do not add nodes!).

- The combination of tree descriptions to form bigger tree descriptions can be seen as **inheritance**, because descriptions can only be added, not removed. (monotonicity)
LEXICAL GENERALIZATIONS: ACTIVE-PASSIVE ALTERNATION

• Active and passive are derived independently
  – Commonalities can be factored out using disjunction in the descriptions the trees satisfy.
EXAMPLE: RESULTATIVE CONSTRUCTIONS

- **Kim painted the barn red.**

- **The barn was painted red.**

Burkhardt, Kallmeyer & Lichte (subm.)
EXAMPLE: RESULTATIVE CONSTRUCTIONS

Subj:
ActorSubj V UndergoerSubj

Transitive:
Subj ∨
((ActiveVerb ∧ DirObj) V
(PassiveVerb ∧ (ByObj V None))) ∧
(Resultative V None)
TAG AND METAGRAMMARS

• Drawbacks of TAG & metagrammars
  - Due to the precompilation step, TAG’s EDL is usually limited in order to limit the number of ETs. For example, every verbal ET corresponds to one argument structure construction.
  - For this reason, it is difficult to analyze cases of coordination in which more than one argument structure construction is found:
    - *She offered and made me a wonderful espresso.*  (Müller 2019a)
      - The verbs *offered* and *made* differ with respect to the role they assign to the "dative" pronoun *me*:
        - For *offered, me* is an obligatory argument with a specific role such as GOAL.
        - For *made, me* is an optional argument with a benefactive role.
  
• However, we will show that coordination of unlikes can be treated using a more flexible EDL, namely the one of TUCO, without resorting to lexical rules or ad-hoc constructions.
RADICAL EDL WITH TREE UNIFICATION & CONSTRAINTS (TUCO)

Idea

• Tree descriptions do not describe ETs, but derived trees.
  ➔ No precompilation → Tree descriptions are effective immediately.

• Furthermore, tree descriptions have the shape of constraints:
  – $X \rightarrow Y$: If $X$ is true/exists, then $Y$ is also true/exists.
  – This can be characterized as conditional addition of descriptions → corresponds to inheritance.

• Tree unification is used instead of substitution and adjunction.
RADICAL EDL WITH TUÇO

Example

- Tree unification of spinal ETs (but they could be any shape)
- Tree constraints
  - enforce correct linearization
  - add information about semantic macroroles (following Van Valin):
    - $\text{NP}^A$ is the actor, $\text{NP}^U$ is the undergoer.
  - make sure that at most one $\text{NP}$ has the undergoer role.
BENEFACTIVE & DITRANSITIVE CONSTRUCTION

- Benefactives are semantic roles often expressed as dative NPs or *for*-PPs.
  - The “dative” in English is indicated by the position between the full verb and the accusative/undergoer.
  - *Kim painted Sue the barn.*

- However, the dative NP is ambiguous, and could be also the goal argument of a ditransitive verb such as *give*:
  - *Kim gave Sue the barn.*
COORDINATION OF UNLIKES

S ——— S ——— S ——— S ——— S ——— S ——— S

NP | V | Conj | V | NP | NP | NP ——— NP

N | offered | and | made | N | Det | N

She | me | an | espresso

\[ \begin{array}{c}
\text{S} \\
\text{NP}_1 \leftarrow + \text{V} \\
\text{NP}_1 \leftarrow + \text{NP} \leftarrow + \text{V} \\
\rightarrow \text{NP}_1^A
\end{array} \]

\[ \begin{array}{c}
\text{S} \\
\text{S} \\
\rightarrow \text{NP}_1^A
\end{array} \]

\[ \begin{array}{c}
\text{V} \leftarrow + \text{NP} \\
\rightarrow \text{NP}_1^V
\end{array} \]

\[ \begin{array}{c}
\text{NP}_1^V \leftarrow + \text{NP}_1^V \\
\rightarrow \rightarrow \bot
\end{array} \]

\[ \begin{array}{c}
\text{S} \\
\rightarrow \text{NP}_1^B
\end{array} \]

\[ \begin{array}{c}
\text{V}_m \leftarrow + \text{NP}_1 \leftarrow + \text{NP}_2 \\
\rightarrow \text{NP}_1^G
\end{array} \]
CONCLUSION

• The (putative) limits of inheritance in constructionist grammar theories
  – There are limits, but are they relevant?
  – In this talk, we argued that inheritance generally suffices within an EDL framework that is sufficiently flexible.
  – Using TAG and the new TUCO formalism, we demonstrated this on a selection of phenomena:
    • active passive alternation in combination with the resultative construction
    • coordination of unlikes (involving benefactives and ditransitives)
• Inheritance may be insufficient for specific kinds of analyses (e.g. deriving passive from active), but a rewriting mechanism could be added to achieve this (with all the computational downsides).
• However, in our opinion, this is orthogonal to the distinction between BDL and EDL, or between lexical and phrasal approaches.
LIST OF REFERENCES


LEXICAL GENERALIZATIONS: ACTIVE-PASSIVE ALTERNATION

- Passive is derived from active → using destructive rewriting of trees (known as "metarules" or "lexical rules", basically transformations)
  - Metarules must be powerful.
    - deletion, copying, recursive application, metavariables over trees
    - thereby: order sensitive, non-declarative
    - in the unrestricted case: undecidable
  - Metarules can be restricted: finite closure, bi-closure, explicit ordering, …
  - However, it is unclear why metarules are necessary, i.e., why they are preferable to disjunction.
  - Furthermore, this does not distinguish EDL an BDL approaches.