Inferring Minimalist Grammars with an SMT-Solver

Sagar Indurkhya
MIT
77 Mass. Ave.
Cambridge, MA 02139
indurks@mit.edu

Overview. The Minimalist Grammar (MG) formalism (Stabler, 1996) is a well established formal model of syntax inspired by the Minimalist Program (Chomsky, 1995). We introduce (1) a novel parser for MGs[1], encoded as a system of first-order logic formulae that may be evaluated using a solver for Satisfiability Modulo Theories (SMT) (De Moura and Bjørner, 2008; Barrett and Tinelli, 2018), and (2) a novel procedure for inferring MGs using this parser. The input to this procedure is a sequence of sentences that have been annotated with syntactic relations such as semantic role labels (connecting arguments to predicates) and subject-verb agreement. The output of this procedure is a set of MGs, each of which is able to parse the sentences in the input sequence such that the parse for a sentence has the same syntactic relations as those specified in the annotation for that sentence. We applied this procedure to a set of sentences annotated with syntactic relations and evaluated the inferred grammars using cost functions inspired by the Minimum Description Length (MDL) principle (Barron et al., 1998; Grünwald, 2007) and the Subset principle (Berwick, 1985; Wexler, 1993). Inferred grammars that were *optimal* with respect to certain combinations of these cost functions were found to align closely with contemporary theories of Minimalist syntax (Hornstein et al., 2005; Adger, 2003; Radford, 1997), producing the prescribed syntactic structures for a range of constructions that include ditransitive predicates, passivization and Wh-fronting for question formation.

Inference Procedure. Our inference procedure takes the form of a computational model of language acquisition (Chomsky, 1965; Berwick, 1985) consisting of: (1) an initial state, S_0, consisting of a system of first-order logical formulae that serve as axioms for deducing the class of minimalist lexicons; (2) the input, consisting of a sequence of n sentences, denoted $I_1, I_2, \ldots, I_n$, each of which is annotated with syntactic relations between pairs of words in the sentence; (3) a function, Q, that takes as input a state, S_i, and an annotated sentence, I_i, and outputs the successor state, S_{i+1}; (4) a function, R, that maps a state S_i to a set of MG lexicons, G_i, with the property that for each sentence I_j in the input sequence, each lexicon $L \in G_i$ can produce a parse p_j^L such that the syntactic relations in p_j^L parse match those specified in the annotation of s_j. In the case of the initial state, S_0, since there are no constraints yet imposed by the input, $R(S_0)$ will map to the set of all minimalist lexicons. The procedure consumes the input sequence one annotated sentence at a time, using Q to drive the initial state, S_0, to the final state, S_n; the function R is then applied to S_n to produce a set of MG lexicons, G_n, that constitutes the output of the inference procedure.

We implemented this inference procedure by encoding an MG parser as a system of first-order, quantifier-free logical formulas that could be solved with the Z3 SMT-solver (De Moura and Bjørner, 2011; Cadar and Sen, 2013).[2] This system of formulas is composed of formulas for MG parse trees that are connected (by way of shared symbols) to a formula for an MG lexicon (i.e. S_0); by imposing constraints on the formulas for parse trees (via Q), the set of solutions to the lexicon formula is restricted (i.e. R is constrained). When the inference procedure consumes an annotated sentence from the input sequence, the function Q: (1) instantiates a formula for an MG parse;

[1] We used the chain-based formulation of MGs presented in (Stabler and Keenan, 2003).

[2] This approach is inspired by earlier work that modeled grammar with logic (Pereira and Warren, 1983; Rayner et al., 1988; Stabler, 1993; Rogers, 1998; Graf, 2013).

I_i	Sentence	Locality Constraints
I_1	who has eaten/V icecream/N?	$\theta_{\text{eaten}}[s\text{: who}, o\text{: icecream}], Agr_{\text{has}}[s\text{: who}]$
I_2	icecream/N was eaten/V.	$\theta_{\text{eaten}}[o\text{: icecream}], Agr_{\text{was}}[s\text{: icecream}]$
I_3	who was eating/V icecream/N?	$\theta_{\text{eating}}[s\text{: who}, o\text{: icecream}], Agr_{\text{was}}[s\text{: who}]$
I_4	was pizza/N eaten/V?	$\theta_{\text{eaten}}[o\text{: pizza}], Agr_{\text{was}}[s\text{: pizza}]$
I_5	what has john/N eaten/V?	$\theta_{\text{eaten}}[s\text{: john}, o\text{: what}], Agr_{\text{has}}[s\text{: john}]$
I_6	has mary/N eaten/V pizza/N?	$\theta_{\text{eaten}}[s\text{: mary}, o\text{: pizza}], Agr_{\text{has}}[s\text{: mary}]$
I_7	was john/N eating/V pizza/N?	$\theta_{\text{eating}}[s\text{: john}, o\text{: pizza}], Agr_{\text{was}}[s\text{: john}]$
I_8	what was mary/N eating/V?	$\theta_{\text{eating}}[s\text{: mary}, o\text{: what}], Agr_{\text{was}}[s\text{: mary}]$
I_9	what was eaten/V?	$\theta_{\text{eaten}}[o\text{: what}], Agr_{\text{was}}[s\text{: what}]$
I_{10}	was mary/N given/V pizza/N?	$\theta_{\text{given}}[o\text{: pizza}, i\text{: mary}], Agr_{\text{was}}[s\text{: mary}]$
I_{11}	what has mary/N given/V john/N?	$\theta_{\text{given}}[s\text{: mary}, o\text{: what}, i\text{: john}], Agr_{\text{has}}[s\text{: mary}]$
I_{12}	mary/N has given/V john/N money/N.	$\theta_{\text{given}}[s\text{: mary}, o\text{: money}, i\text{: john}], Agr_{\text{has}}[s\text{: mary}]$
I_{13}	who was money/N given/V to/P?	$\theta_{\text{given}}[o\text{: money}, i\text{: to who}], Agr_{\text{was}}[s\text{: money}]$
I_{14}	who has john/N given/V money/N to/P?	$\theta_{\text{given}}[s\text{: john}, o\text{: money}, i\text{: to who}], Agr_{\text{has}}[s\text{: john}]$

Table 1: Model Input — A sequence of sentences annotated with syntactic relations. Some phonetic forms have their category pre-specified, indicated by a suffix of a slash followed by the category. Locality constraints include agreement (Agr) and predicate-argument structure (i.e. a θ grid), with the predicate indicated in the suffix and the subject, object and indirect object components marked by "s:", "o:" and "i:" respectively. The type of the sentence, *declarative* or *interrogative*, is indicated by the end-of-sentence punctuation.

Lexicon-A	Lexicon-B
eaten/$V :: = x_4, \sim x_4$	eaten/$V :: = x_5, \sim x_1$
eating/$V :: = x_4, \sim x_4$	eating/$V :: = x_5, \sim x_1$
given/$V :: = x_4, = x_4, \sim x_4$	given/$V :: = x_5, = x_5, \sim x_1$
given/$V :: = x_2, = x_4, \sim x_4$	has/$T :: = x_0, +l, \sim x_2$
has/$T :: = x_4, +l, \sim x_0$	icecream/$N :: \sim x_5$
has/$T :: = x_4, +l, \sim x_4$	icecream/$N :: \sim x_5, -l$
icecream/$N :: \sim x_4$	john/$N :: \sim x_5$
icecream/$N :: \sim x_4, -l, -r$	john/$N :: \sim x_5, -l$
john/$N :: \sim x_4$	mary/$N :: \sim x_5, -l$
john/$N :: \sim x_4, -l$	money/$N :: \sim x_5$
mary/$N :: \sim x_4, -l$	money/$N :: \sim x_5, -l$
mary/$N :: \sim x_4, -l, -r$	pizza/$N :: \sim x_5$
money/$N :: \sim x_4$	pizza/$N :: \sim x_5, -l$
money/$N :: \sim x_4, -l$	to/$P :: = x_4, \sim x_5$
pizza/$N :: \sim x_4$	was/$T :: = x_0, +l, \sim x_2$
pizza/$N :: \sim x_4, -l$	what/$N :: \sim x_5, -r$
to/$P :: = x_2, \sim x_2$	what/$D :: \sim x_5, -l, -r$
was/$T :: = x_4, +l, \sim x_4$	who/$D :: \sim x_4, -r$
was/$T :: = x_4, +l, \sim x_0$	who/$N :: \sim x_5, -l, -r$
what/$N :: \sim x_4, -r$	$\epsilon/v :: = x_1, \sim x_0$
what/$N :: \sim x_4, -l, -r$	$\epsilon/C_{declarative} :: = x_2, C$
who/$D :: \sim x_2, -r$	$\epsilon/C_{question} :: <= x_2, C$
who/$D :: \sim x_4, -l, -r$	$\epsilon/C_{question} :: <= x_2, +r, C$
$\epsilon/v :: = x_4, \sim x_4$	$\epsilon/v :: <= x_1, = x_5, \sim x_0$
$\epsilon/C_{question} :: <= x_4, C$	
$\epsilon/v :: <= x_4, = x_4, \sim x_4$	
$\epsilon/C_{question} :: <= x_0, +r, C$	
$\epsilon/C_{declarative} :: = x_4, +r, C$	

Table 2: Examples of inferred lexicons that satisfy the conditions imposed by the input sequence in Table-1. Each lexical item has the form, $(PF/CAT :: SFS)$, consisting of a phonetic form (PF), a category (CAT) and a sequence of syntactic features (SFS). The phonetic forms ϵ is covert (unpronounced). The selectional features are $\{x_0, x_1, ..., x_5\}$ and the licensing features are $\{l, r\}$.

(2) translates the annotations for the sentence into (logic) formulas that constrain the parse tree – e.g. predicate-argument relations and morphological agreement are translated into locality constraints[3]; (3) adds these new formulas to the existing system of formulas in S_i to produce S_{i+1}. In order to compute the set of lexicons, $G_i = R(S_i)$, we used the Z3 SMT-solver to solve for the set of lexicons satisfying the formulae in S_i.

Data. The input to the inference procedure is a sequence of fourteen sentences, $I_1 - I_{14}$ in Table-1, each annotated with predicate-argument relations as well as morphological agreement; the sentences listed include passive constructions (I_2, I_4, I_{10}), ditransitive constructions ($I_{11} - I_{14}$), yes/no-questions (I_4, I_6, I_7, I_{10},) and wh-questions (I_1, I_3, I_5, I_8, I_9, I_{11}, I_{13}, I_{14}).

Analysis. We used our procedure to infer a set of minimalist lexicons, denoted here as G^*, from the input sequence described in Table-1. Lexicons sampled from G^* produced parses that do not align with those prescribed by contemporary theories of minimalist syntax. (See Lexicon-A in Table-2 for an example of such a lexicon.)

We filtered out such lexicons by using Z3 to identify lexicons in G^* that were *optimal* with respect to three cost functions that (respectively): (i) *minimized* the number of lexical entries in the lexicon; (ii) *minimized* the total number of selectional and licensing features in the lexicon and the parses (this rewards reduction in the total size of both the lexicon and the parses); (iii) *maximized* the

[3]The *principle of syntactic locality* asserts that syntactic relations are established locally by merge (Sportiche et al., 2013).

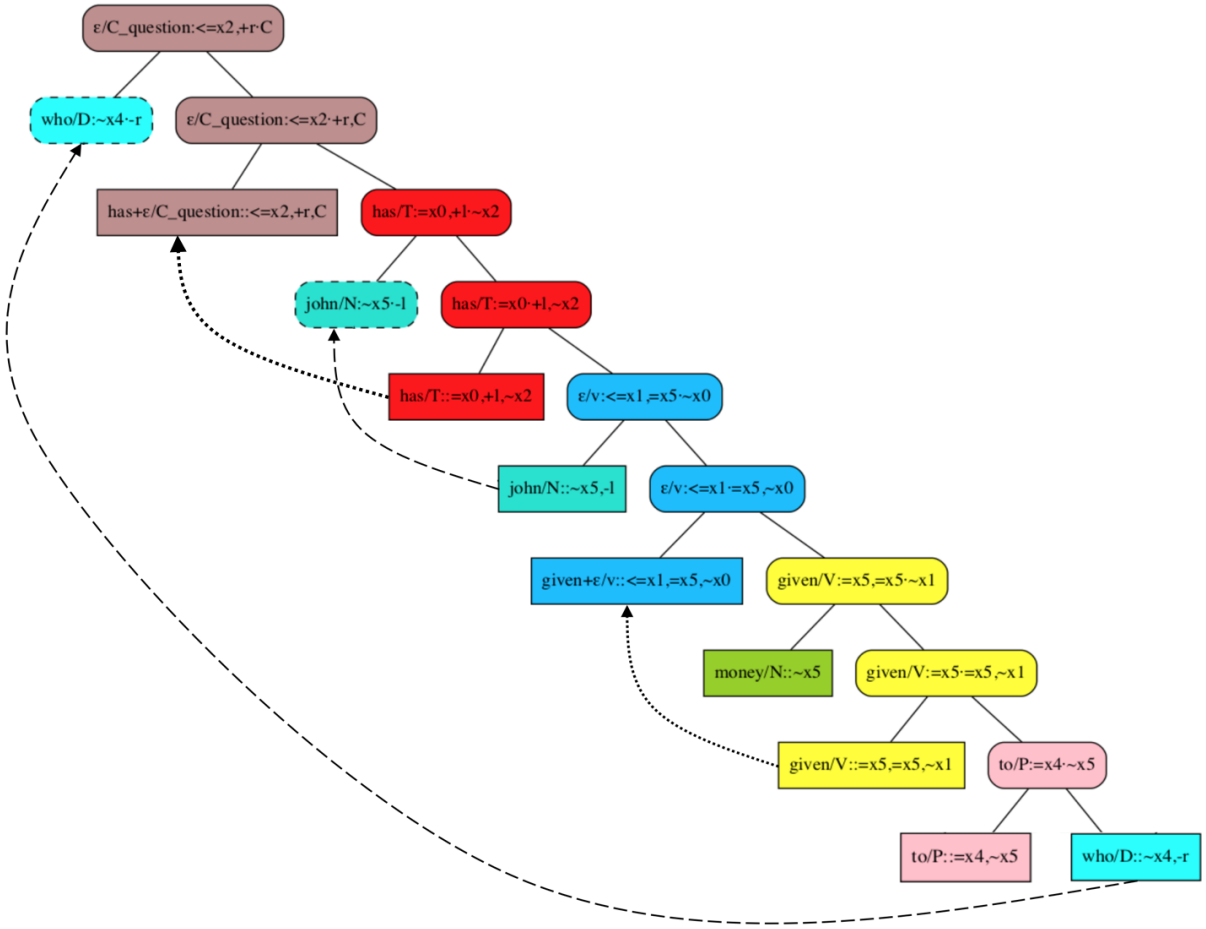

Figure 1: An MG parse for the sentence "Who has John given money to?" (see I_{14} in Table-1 for annotations) derived from Lexicon-B in Table-2. This parse accords with the parse prescribed by contemporary theories of syntax. The feature sequences displayed in non-leaf nodes have a dot, $\cdot$, separating features that have already been consumed (on the left) from those that have not (on the right). The dashed arrows denote phrasal movement. The dotted arrows denote head movement. Nodes with the same *head* have the color. The parse is assembled in a bottom-up manner via merge: "eating" merges with "to who" (formed by first merging "to" and "who") and then with "money", thus establishing (via locality) predicate-argument relations; the resulting structure merges with an empty lexical node with category v, undergoing V-to-v head-movement before merging with the argument "john" in accordance with the VP-Internal Subject Hypothesis (Hale and Keyser, 2002); the resulting structure then merges with the auxiliary verb "has", after which the argument "john" undergoes subject-raising from the VP-shell by (internally) merging with "has", thus establishing morphological agreement between "john" and "has"; next, the head of "has" undergoes T-to-C head-movement to merge with the covert complementizer, $\epsilon/C_{question}$, which indicates that the sentence is an interrogative; finally, "who" undergoes wh-fronting by (internally) merging with $\epsilon/C_{question}$. Wh-fronting (of "who") and Subject-raising (of "john"), instances of A'-movement and A-movement respectively, are triggered by different licensor features, the former by $+r$ and the latter by $+l$.

number of *distinct* selectional features in the lexicon (this rewards lexicons that are more exclusive in which structures they generate).[4] We encoded these cost function as first order logical formulae, adding them to the SMT-solver after running the inference procedure, and then re-solving; the resulting set of (inferred) MGs are optimal with respect to the specified cost functions.

[4]Cost functions (i) and (ii) are based on the MDL principle (see also (Stabler, 1998)), whereas cost function (iii) is based on the Subset principle.

This produced a subset of G^*, denoted F^*, in which each lexicon had exactly: 24 lexical items; 48 features in the lexicon (not including the special feature C); 202 features in the parses; at least five distinct selectional features. Lexicons sampled from F^* produced parses that respect the syntactic relations prescribed in Table-1 and do align with structures prescribed by contemporary theories of minimalist syntax. See Lexicon-B in Table-2 for a representative member of F^* – the

syntactic phenomenon that Lexicon-B correctly models includes: A' movement (Wh-fronting for question formation); a (double) VP shell structure that employs V-to-v head-movement (as part of the predicate-argument structure within the parse tree; see (Hale and Keyser, 2002)); T-to-C head-movement (i.e. subj-auxiliary verb inversion) and A-movement (subject raising for morphological agreement). See Figure-1 for a parse produced by Lexicon-B that demonstrates these syntactic phenomenon.

Conclusion. Our results demonstrate that our procedure for inferring MGs is able to acquire knowledge of syntax from psychologically plausible input and employ movement (i.e. displacement) to establish multiple (crossing and nested) discontinuous relations within a syntactic structure. We observe that by enabling and disabling axioms in our model, it is possible to determine which axioms are redundant, and *thereby gain insight into whether the universal linguistic principles, from which the axioms of the system are largely derived, are justified or can be discarded*, thus aiding in the evaluation of the Strong Minimalist Thesis (Chomsky, 2001, 2008). Going forward, we will focus on examining the over-generations produced by the MGs inferred by our procedure and understanding how these over-generations relate to the cost functions used by our procedure for identifying optimal grammars.

Acknowledgements. The author would like to thank Robert C. Berwick, Sandiway Fong, Beracah Yankama, and Norbert Hornstein for their suggestions, feedback, and inspiration.

References

David Adger. 2003. *Core syntax: A minimalist approach*, volume 33. Oxford University Press Oxford.

Clark Barrett and Cesare Tinelli. 2018. Satisfiability modulo theories. In *Handbook of Model Checking*, pages 305–343. Springer.

Andrew Barron, Jorma Rissanen, and Bin Yu. 1998. The minimum description length principle in coding and modeling. *IEEE Transactions on Information Theory*, 44(6):2743–2760.

Robert C. Berwick. 1985. *The acquisition of syntactic knowledge*. MIT press.

Cristian Cadar and Koushik Sen. 2013. Symbolic execution for software testing: three decades later. *Commun. ACM*, 56(2):82–90.

Noam Chomsky. 1965. *Aspects of the theory of syntax*. MIT Press.

Noam Chomsky. 1995. *The Minimalist Program*. MIT Press.

Noam Chomsky. 2001. Derivation by phase. In Michael Kenstowicz, editor, *Ken Hale: A life in language*, pages 1–52. MIT press.

Noam Chomsky. 2008. On phases. *Current Studies in Linguistics Series*, 45:133.

Leonardo De Moura and Nikolaj Bjørner. 2008. Z3: An efficient smt solver. TACAS'08/ETAPS'08, pages 337–340. Springer-Verlag.

Leonardo De Moura and Nikolaj Bjørner. 2011. Satisfiability modulo theories: introduction and applications. *Communications of the ACM*, 54(9):69–77.

Thomas Graf. 2013. *Local and transderivational constraints in syntax and semantics*. Ph.D. thesis, University of California at Los Angeles.

Peter D Grünwald. 2007. *The minimum description length principle*. MIT press.

Kenneth L. Hale and Samuel J. Keyser. 2002. *Prolegomenon to a theory of argument structure*, volume 39. MIT press.

Norbert Hornstein, Jairo Nunes, and Kleanthes K Grohmann. 2005. *Understanding minimalism*. Cambridge University Press.

Fernando C. N. Pereira and David H. D. Warren. 1983. Parsing as deduction. In *Proceedings of the 21st Annual Meeting on Association for Computational Linguistics*, ACL '83, pages 137–144. Association for Computational Linguistics.

Andrew Radford. 1997. *Syntactic theory and the structure of English: A minimalist approach*. Cambridge University Press.

Manny Rayner, Åsa Hugosson, and Göran Hagert. 1988. Using a logic grammar to learn a lexicon. In *Proceedings of the 12th conference on Computational linguistics-Volume 2*, pages 524–529. Association for Computational Linguistics.

James Rogers. 1998. *A descriptive approach to language-theoretic complexity*. CSLI Publications.

Dominique Sportiche, Hilda Koopman, and Edward Stabler. 2013. *An introduction to syntactic analysis and theory*. John Wiley & Sons.

Edward Stabler. 1996. Derivational minimalism. In *Intl. Conf. on Logical Aspects of Comp. Ling.*, pages 68–95. Springer.

Edward P Stabler. 1993. *The Logical Approach to Syntax*. MIT Press.

Edward P. Stabler. 1998. Acquiring languages with movement. *Syntax*, 1(1):72–97.

Edward P. Stabler and Edward L Keenan. 2003. Structural similarity within and among languages. *Theoretical Computer Science*, 293(2):345–363.

Kenneth Wexler. 1993. The subset principle is an intensional principle. In *Knowledge and language*, pages 217–239. Springer.