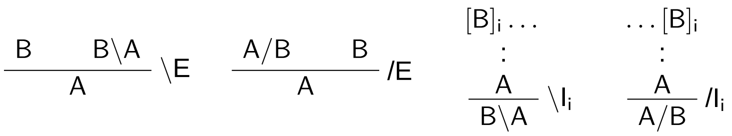
Grammar as logic, processing as deduction, actions as theorems, states as propositions

Shinnosuke Isono[1], Kohei Kajikawa[2]
[1]National Institute for Japanese Language and Linguistics (NINJAL), [2]Georgetown University
[1]s-isono@ninjal.ac.jp

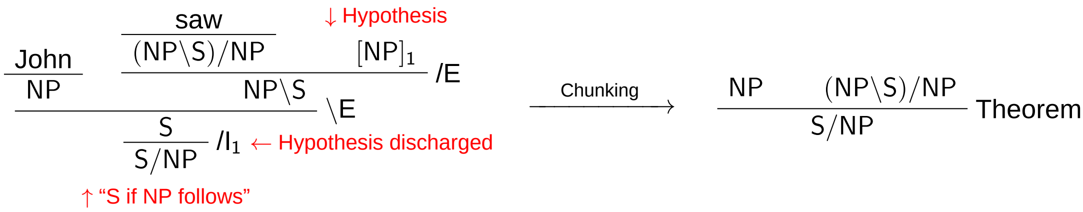
We argue for the **categorial** view on natural language grammar as a unifying, high-level framework to analyze sentence processing. While combinatory categorial grammar (CCG) is recently shown to be effective in modeling neural and behavioral data [1, 2], the principles behind the categorial approach and its full potential are more visible in the **type-logical** variant [3], which we adopt here. In type-logical grammars (TLGs), grammar is understood as a formal logic. Thus, categories like S and NP are primitive propositions, and complex categories $\mathsf{B}\backslash\mathsf{A}$ and A/B are directionally-sensitive implications. Rules listed in (1) follow from this analogy: E(limination) rules correspond to Modus Ponens and I(ntroduction) to Conditional Proof. This simple framework provides an insightful perspective on sentence processing, accounting for its key characteristics such as:

- **Incremental, predictive,** and **grammar-guided**: TLG enables a tight integration of competence grammar and incremental, predictive parsing since it can parse any left fragment of a sentence into a single proof tree by hypothesizing the continuation, as in (2). TLG even offers incremental parses for long-distance dependencies (LDDs) (see (3)). This contrasts with CFGs and Minimalist Grammars, which need extra-grammatical mechanisms to parse LDDs incrementally. CCG is similar to TLG in these respects, but TLG is more flexible: while CCG imposes a fixed limit on incrementality by its rule set [4], TLG offers a scale of incrementality or parsing depth, based on the complexity of inference, that comprehenders under different conditions (cognitive resources, pressure, attention, etc.) may adopt.
- **Resource-efficient**: TLG-based processing accounts for the remarkable efficiency of sentence processing by information compression at two levels (cf. [5]). First, frequently appearing deductive steps can be chunked into a single action (see (2)). This amounts to memoization of a theorem. CCG can be seen as hard-coding a specific set of theorems. TLGs suggest that the parser may be more **adaptive**, with theorems can be formed dynamically from experience. Second, categories like S/NP can be seen as an austere summary of the past input extracting information needed for future processing ("S if NP follows"). In other words, categories represent parser states (see (4)). See [2] for behavioral evidence.
- **Resource-constrained**: When chunking fails, unconnected chunks use up the memory space. The difficulty of double center embedding can be attributed such a chunking failure, followed by interference (see (5)). Chunking fails when the parser cannot find a single category for it in a reasonable time using theorems available. And available theorems depend on past experience (see (2)). Thus, TLG provides a basis to analyze the interplay of frequency and rules and its effect on memory-constrained parsing.
- **Both resilient and vulnerable to ambiguity**: TLG also offers Boolean connectives ("and"/"or") to build categories, which allow the parser to deal with some ambiguities without invoking multiple parses. The NP/S ambiguity, for example, can be dealt with by specifying the complement of the verb as "NP or S". Even under the single-path model, then, the disambiguation only requires local reanalysis of the most recent parsing steps at the NP (see (6a)). The NP/Z ambiguity cannot be dealt with the same way if we disallow Zero (silent) categories, and needs reanalysis of the verb. The backtracking needed for this reanalysis accounts for the relative difficulty of NP/Z compared to NP/S (see (6b)).

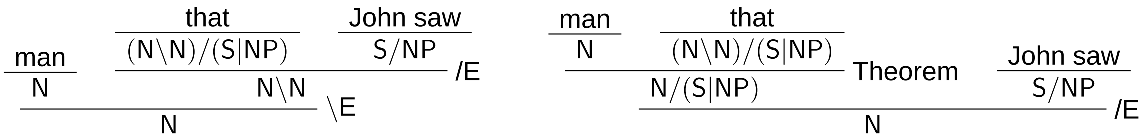
(1) **Rules of a simple TLG** (Lambek calculus, natural deduction style; n.b. B\A in TLG = A\B in CCG).



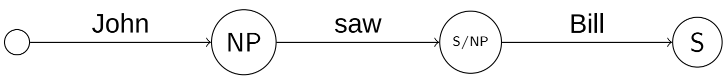
(2) **Incremental parsing and chunking.**



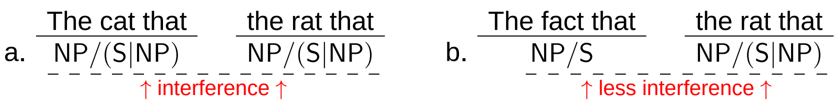
(3) **Long distance dependency** (Left: non-incremental, right: incremental; S|NP = NP\S or S/NP).



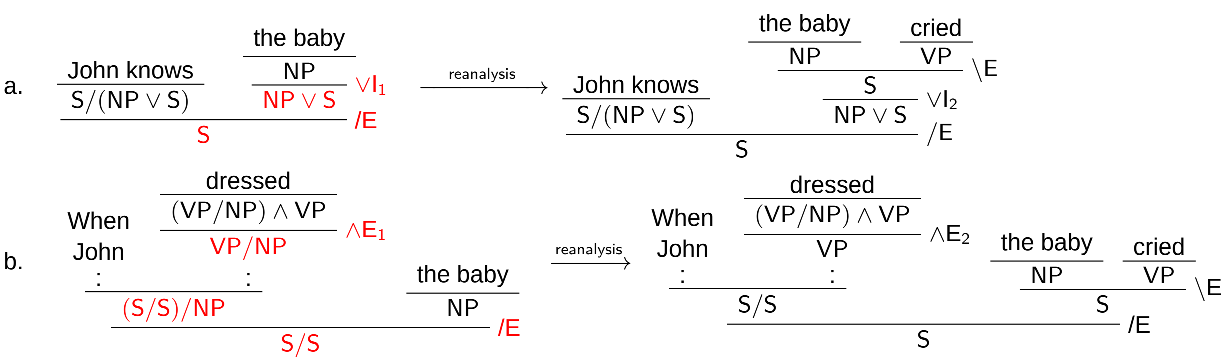
(4) **Categories as parser states.**



(5) **Double center embedding.**



(6) **NP/S vs. NP/Z ambiguity** (VP abbreviates NP\S; the rules for ∧ ("and") and ∨ ("or") are the same as in propositional logic; red indicates targets of reanalysis)



References

[1] Stanojević, M., Brennan, J. R., Dunagan, D., Steedman, M., & Hale, J. T. (2023). Modeling structure-building in the brain with CCG parsing and large language models. *Cognitive Science*, *47*(7), e13312.

[2] Isono, S. (2024). Category Locality Theory: A unified account of locality effects in sentence comprehension. *Cognition*, *247*, 105766.

[3] Moortgat, M. (1997). Categorial Type Logics. In J. van Benthem & A. ter Meulen (Eds.), *Handbook of logic and language* (pp. 93–177). MIT Press.

[4] Demberg, V. (2012). Incremental derivations in CCG. *Proceedings of the 11th International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+11)*, 198–206.

[5] Christiansen, M. H., & Chater, N. (2016). The Now-or-Never bottleneck: A fundamental constraint on language. *Behavioral and Brain Sciences*, *39*, e62.