A Theory of Processing Difficulty Based on Syntactic Prediction

Vera Demberg and Frank Keller

V.Demberg@ed.ac.uk, keller@inf.ed.ac.uk

Abstract: We propose a model of human sentence processing which implements incrementality with full connectedness and explicit prediction of upcoming structure. The model predicts processing difficulty as a combination of the cost of retrieving, verifying, and integrating syntactic expectations.

Motivation

Recent work on sentence processing suggests that comprehenders make **predictions** while they process language: not only do they integrate new words **incrementally** with previous input, put they also anticipate upcoming linguistic material (Kamide, Scheepers, Altmann, & Crocker, 2002; Staub & Clifton, 2006).

Model Architecture

- **Full connectivity** The syntactic structure is fully connected at every word, supporting the incremental construction of semantic interpretations.
- **Prediction** At each word, a set of syntactic expectations is generated; an expectation is an incomplete syntactic structure that specifies the categories need to turn the current input into a grammatical sentence.
- **Parallel Processing** Expectations are held in memory in parallel, and have a probability. We use beam search to only store the most likely expectations and model memory limitations.
- **Decay** Predictions lose activation over time if they are not used to integrate new information. When they are verified against an elementary tree, the activation of the prediction determines how difficult it is to integrate the new information.
- **Verification** Processing difficulty is incurred when predictions are verified: either expectations become incompatible with the current input (Jurafsky-style reranking), or predictions are satisfied (predicted node and new node are unified).

Design Questions

- Prediction strategy: **eager or lazy**?
- Prediction **grain size**
- **Tractability**
  - each lexeme has on average 4.6 different trees
  - 13k unique prediction trees, 220 unique non-predictive trees
  - Precombine predictive tree in lexicon?
    number of prediction trees that were combined for connectivity:
    1: 94.6%, 2: 4.8%, 3: <1%, 4: <.1%, 5: 1, 6+: 0
- Prediction trees for **free word order** languages

Implementation Using TAG

Advantages of using Tree-Adjoining Grammar (TAG) to formalise prediction:

- Makes it possible to model of incremental processing with **full connectivity** (Mazzei, Lombardo, & Sturt, 2007).
- TAG has an **extended domain of locality**, important for modeling long-range predictions (see below).
- Distinction between **modifiers** and **arguments**: modifiers are not predicted unless needed for connectivity; arguments are predicted through subcategorization.

We define Psycholinguistically Motivated TAG (PLTAG), an incremental version of lexicalised TAG that supports full connectivity, as described in (Demberg & Keller, 2008).

Explanatory Power

The proposed theory can account for:

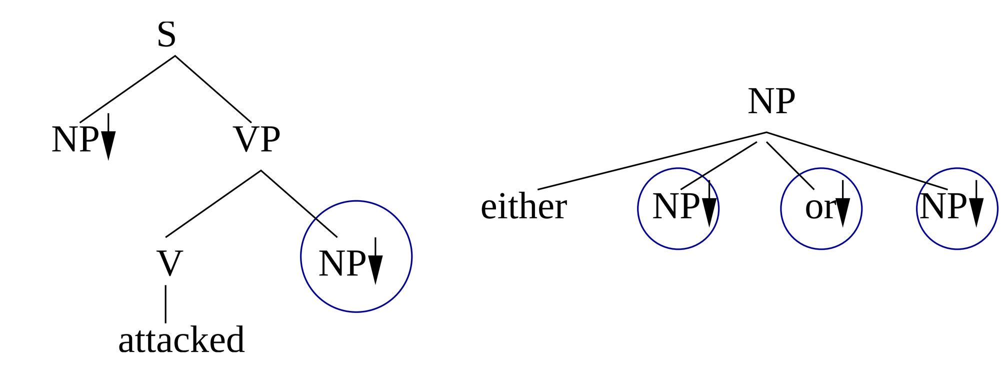
- **Prediction** (Kamide et al., 2002): syntactic categories are predicted explicitly as part of the formalism;
- **Locality effects** (Gibson, 1998): the more dependents are integrated, the more processing cost is incurred, subject to a distance-based decay function;
- **Anti-locality effects** (Konieczny, 2000): the more expectations are discarded, the more processing cost is incurred;
- **Digging-in effects** (Tabor & Hutchins, 2004): discarding expectations that have been maintained for longer is more costly;
- **Ambiguity resolution and garden paths** accounted for by probabilistically ranked parallelism as proposed by Jurafsky (1996).

Prediction and Verification

Predictions are triggered in two cases:

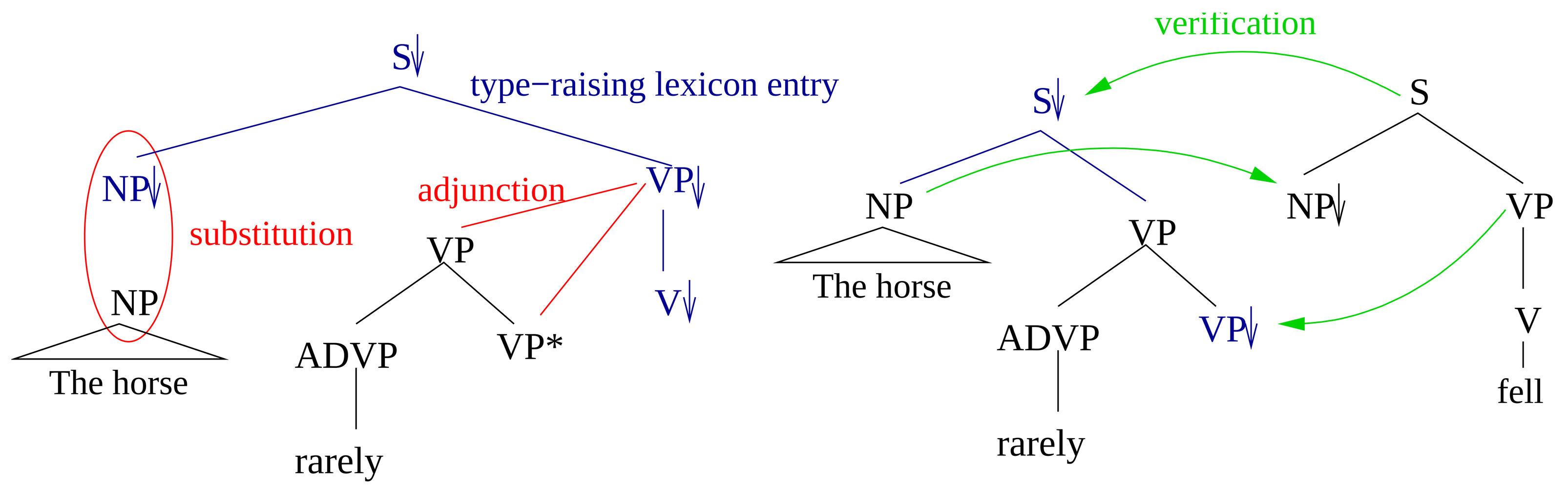
1. through substitution nodes to the right of an anchor in lexicon entries; TAG's extended domain of locality can be used to model prediction in constructions such as *either . . . or* (Staub & Clifton, 2006), see Example 1;
2. under the assumption of incrementality and full connectivity, prediction is technically necessary to interpret structures like *he thinks the . . .* or *the horse rarely . . .*, see Example 2.

Example 1



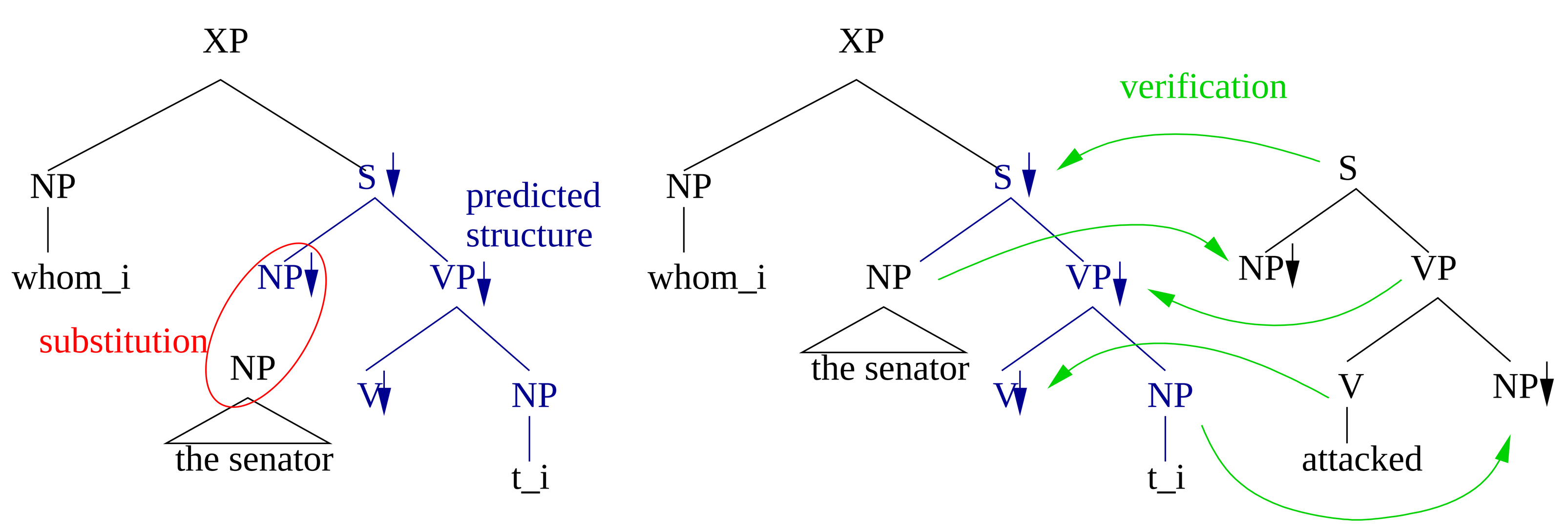
Lexicon entries with substitution nodes to the right of the anchor.

Example 2



PLTAG with type-raising lexicon entry.

Redundancy and Traces



PLTAG processing for an object relative clause.

References

Demberg, V., & Keller, F. (2008). A psycholinguistically motivated version of tag. In *Proc. of the 9th international workshop on tree adjoining grammars and related formalisms (TAG+9 2008).*

Gibson, E. (1998). Linguistic complexity: locality of syntactic dependencies. *Cognition 68*, 1-76.

Jurafsky, D. (1996). A probabilistic model of lexical and syntactic access and disambiguation. *Cognitive Science, 20*, 137-194.

Kamide, Y., Scheepers, C., Altmann, G., & Crocker, M. (2002). Integration of syntactic and semantic information in predictive processing: Anticipatory eye-movements in German sentence processing. In *The annual cuny conference on sentence processing.* New York, NY.

Konieczny, L. (2000). Locality and parsing complexity. *Journal of Psycholinguistic Research, 29-6*, 627-645.

Mazzei, A., Lombardo, V., & Sturt, P. (2007). Dynamic tag and lexical dependencies. *Research on Language and Computation, Foundations of Natural Language Grammar*, 309-332.

Staub, A., & Clifton, C. (2006). Syntactic prediction in language comprehension: Evidence from either...or. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 32*, 425-436.

Tabor, W., & Hutchins, S. (2004). Evidence for self-organized sentence processing: Digging-in effects. *Journal of Experimental Psychology, 30*(2), 431–450.