Ling/CSE 472:
Introduction to Computational Linguistics

May 11: Grammar-Based Treebanking

Overview

- Announcements: Assignment 5
- Review: Edges, nodes, constituents
- Grammar-based treebanking
    - History & Motivation
    - Contents & Methodology
    - Outlook
- Reading questions

Review: Nodes, edges, constituents

- What's the difference?
- Node: A part of a parse tree
- Edge: An object manipulated by a parser in the course of finding parse trees
  - Active v. passive edges
- Constituent: A substring of a sentence dominated by a node in a parse tree

=> Demo

HPSG in one slide

- Key references: Pollard & Sag 1987, Pollard & Sag 1994, Sag, Wasow & Bender 2003 (textbook)

- Phrase structure grammar: Like CFG but with elaborate feature structures instead of atomic node labels

- Monostratal/surface oriented: One structure per input item (no movement), with both syntactic and semantic information

- Lexicalist: Rich information in lexical entries (+ type hierarchy to capture generalizations)

- Core & periphery: Construction inventory includes both very general and very idiosyncratic rules

Flickinger et al 2017: Central claims

- Developing complex linguistic annotations calls for an approach which allows for the incremental improvement of existing annotations by encoding all manual effort in such a way that its value is preserved and enhanced even as the resource is improved over time

- Manual effort:
  - Annotation design => Encode in a grammar
  - Disambiguation => Store disambiguation decisions in a treebank

Minimal Recursion Semantics in one slide

- Key references: Copestake et al 2005, Bender et al 2015
- Underspecified description of logical forms
- Captures predicate-argument structure, partial constraints on quantifier scope, morpho-semantic features
- Computationally tractable, grammar-compatible, and linguistically expressive

English Resource Grammar (Flickinger 2000, 2011)

- Under continuous development since 1993.
- 44,000 item lexicon: function words, open-class words with ‘non-standard’ properties
- Feb 2023 trunk: 295 syntactic rules, 101 lexical rules, 1268 leaf lexical types
- Unknown words given default lexical entries based on POS tagging
- 85-95% coverage of open domain, well-edited English text
- Development genres: newspaper text, Wikipedia pages, bio-medical research, literature, customer service emails, meeting scheduling dialogues…

Redwoods Treebank (Oepen et al 2004)

- Under development since 2001
- As of ‘ninth growth’, 1.5 million tokens
- Initial motivation: train parse ranking models
- Also quite useful for grammar maintenance and development

Redwoods: Contents

- Rich syntactico-semantic structures, from which different ‘views’ can be projected.

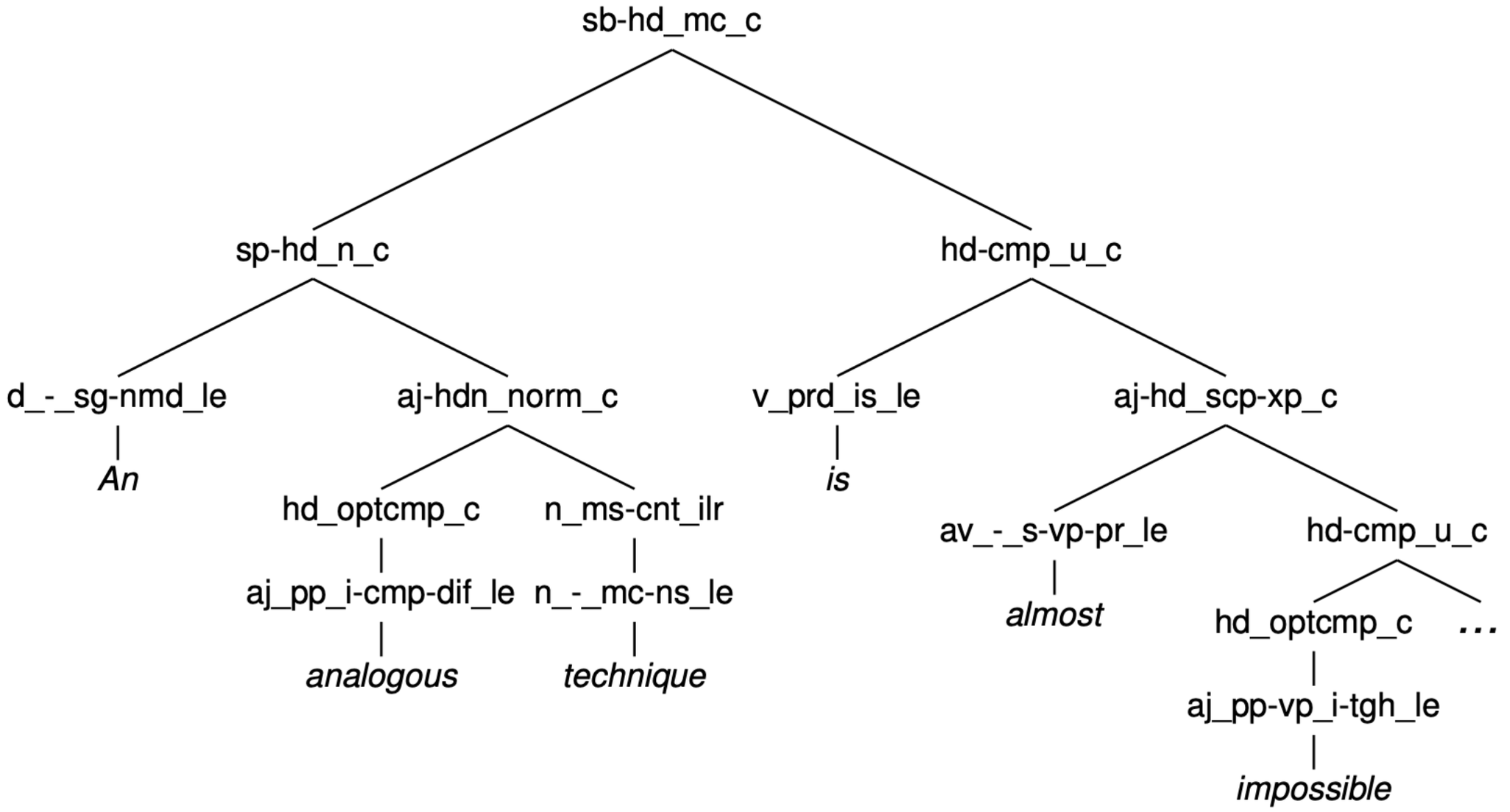
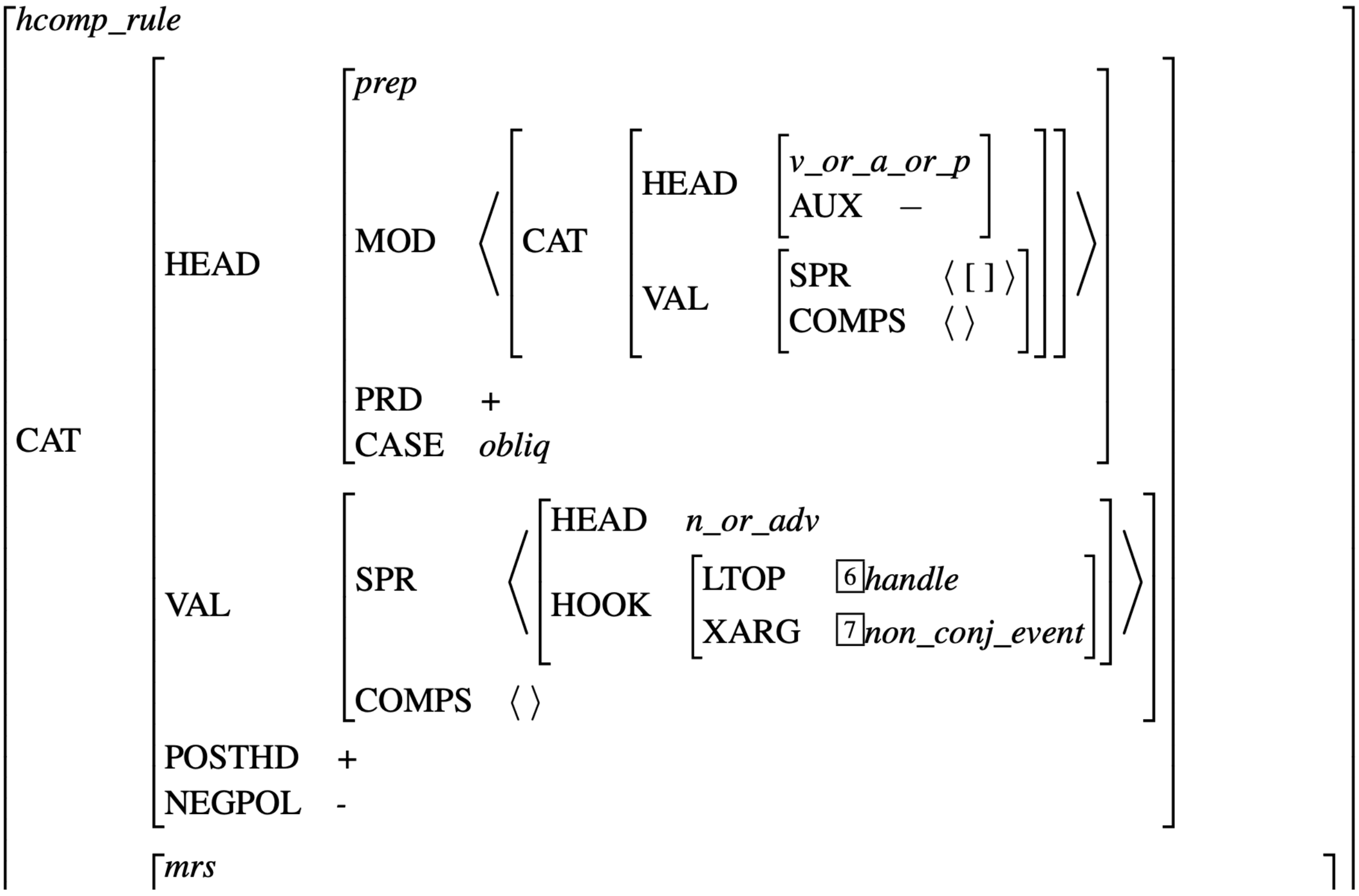


Fig. 1 ERG derivation tree for example (1).


hcomp_rule
CAT
HEAD
prep
MOD
CAT
HEAD
v_or_a_or_p
AUX –
VAL
SPR ⟨ [ ] ⟩
COMPS ⟨ ⟩
PRD +
CASE obliq
VAL
SPR
HEAD n_or_adv
HOOK
LTOP 6 handle
XARG 7 non_conj_event
COMPS ⟨ ⟩
POSTHD +
NEGPOL -
mrs

mrs

HOOK
- LTOP 6
- INDEX 7
- XARG 1

CONT
- RELS ⟨
  - *prep_relation*
    - PRED *_to_p_rel*
    - LBL 6
    - ARG0 7
    - ARG1 1
    - ARG2 13 [*nonconj_ref-ind*, PNG.PN *3p*]
  - *adj_relation*
    - PRED *_other_a_1_rel*
    - LBL 18 *handle*
    - ARG0 [*non_conj_event*, TENSE *no_tense*]
    - ARG1 13
  - *quant_or_wh_relation*
    - PRED *udef_q_rel*
    - LBL *handle*
    - BV 13
    - RSTR 16 *handle*
    - BODY *handle*
  - *reg_nom_relation*
    - PRED *_crop_n_1_rel*
    - LBL 18
    - ARG0 13

⟩

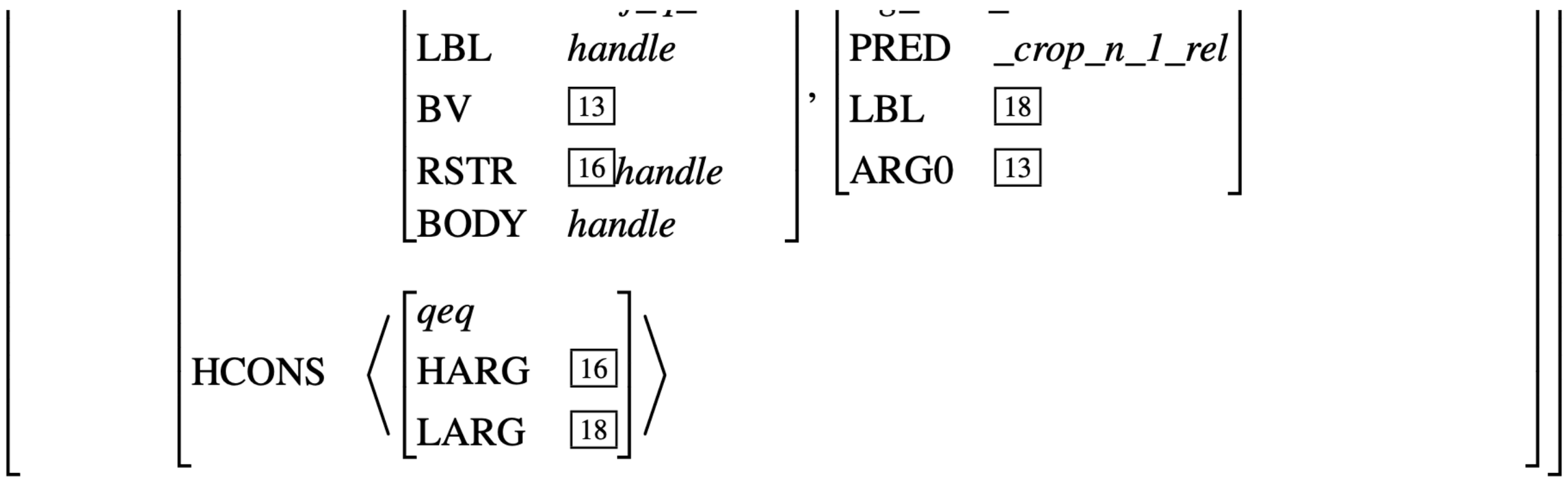


Fig. 2 Partial feature structure for PP *to other crops*

$$
\langle h_1,
\left|\begin{array}{l}
h_4{:}\_a\_q(\text{BV } x_6, \text{RSTR } h_7, \text{BODY } h_5),\\
h_8{:}\_analogous\_a\_to(\text{ARG0 } e_9, \text{ARG1 } x_6), h_8{:}comp(\text{ARG0 } e_{11}, \text{ARG1 } e_9, \text{ARG2 } \_),\\
h_8{:}\_technique\_n\_1(\text{ARG0 } x_6),\\
h_2{:}\_almost\_a\_1(\text{ARG0 } e_{12}, \text{ARG1 } h_{13}), h_{14}{:}\_impossible\_a\_for(\text{ARG0 } e_3, \text{ARG1 } h_{15}, \text{ARG2 } \_),\\
h_{17}{:}\_apply\_v\_to(\text{ARG0 } e_{18}, \text{ARG1 } \_, \text{ARG2 } x_6, \text{ARG3 } x_{20}),\\
h_{21}{:}udef\_q(\text{BV } x_{20}, \text{RSTR } h_{22}, \text{BODY } h_{23}), h_{24}{:}\_other\_a\_1(\text{ARG0 } e_{25}, \text{ARG1 } x_{20}),\\
h_{24}{:}\_crop\_n\_1(\text{ARG0 } x_{20})
\end{array}\right|\\
\{h_1 =_q h_2, h_7 =_q h_8, h_{13} =_q h_{14}, h_{15} =_q h_{17}, h_{22} =_q h_{24}\}\rangle
$$

Fig. 3 Minimal Recursion Semantics for example (1).

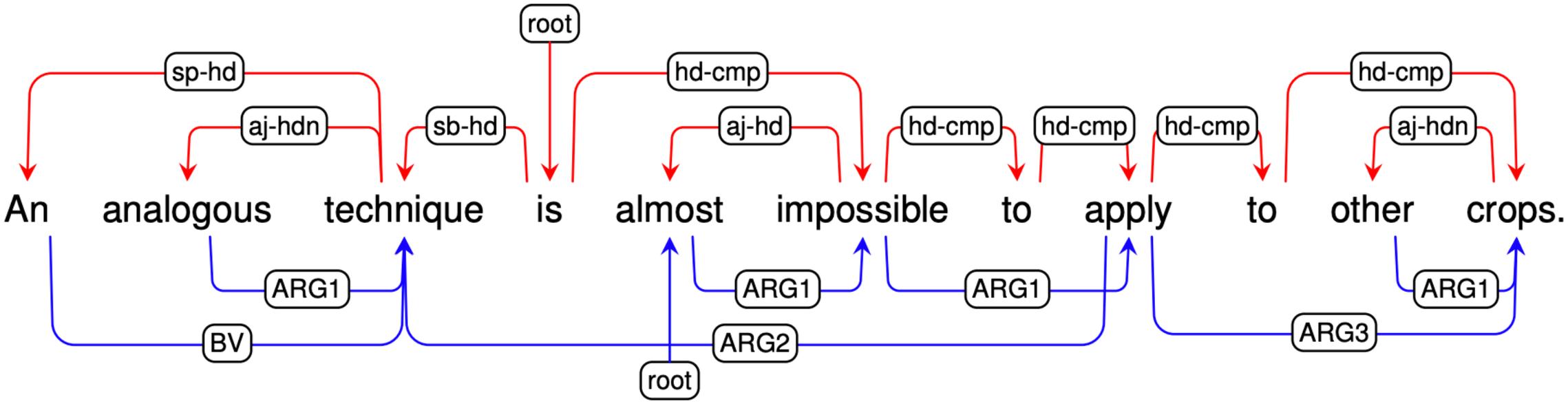


Fig. 4 Bi-lexical syntactic and semantic dependencies for (1).

Redwoods: Methodology

- Parse input corpus
- Calculate ‘discriminants’: properties shared by only a subset of the trees in parse forest (Carter 1997)
    - Picking one tree from among thousands or millions would be infeasible
    - Drawing trees with that level of detail would be infeasible
    - Picking discriminants is quite doable!
- Store both resulting tree & discriminants chosen (and inferred)
    - Maximum value out of all human annotator time

Very high inter-annotator agreement
=> very consistent annotation

- From Bender et al 2015, over 150 sentences from *The Little Prince*

	Annotator Comparison			
Metric	**A vs. B**	**A vs. C**	**B vs. C**	**Average**
Exact Match	0.73	0.65	0.70	0.70
EDM_a	0.93	0.92	0.94	0.93
EDM_{na}	0.94	0.94	0.95	0.94

Table 1: Exact match ERS and Elementary Dependency Match across three annotators.

- Comparable metric for AMR over the same data is 0.71 “SMATCH” (comparable to EDM) (Banarescu et al 2013)

Dynamic treebanking

- Dynamic *refinement* of the treebank
    - Parse corpus with new grammar (better coverage, improved representations)
    - Rerun discriminants chosen in previous annotation rounds
    - Address remaining added ambiguity / newly parsed sentences
- Dynamic *extension* of the treebank
    - Linguistic analysis encoded as a grammar (as opposed to annotation guidelines) can be automatically deployed to new text

Redwoods: Outlook

- Switch from treebanking based on top-500 parses to full-forest (Packard 2015) — done

- Treebanking over robust parsing strategies to capture the remaining 5-15% of sentences

- Integrating further kinds of linguistic annotations (coreference, fine-grained word sense, information structure…)

Reading questions

- In theory, would it be impossible to annotate everything about a language since it is always evolving?

- I know at the beginning of the paper it said that these annotations can be used for things like learning about language structures, but do these annotations have any impact on the use of everyday technologies like translation or speech recognition?

Reading questions

- I don't mean to be rude towards the people who likely have spent a good portion of their lives doing this work, nor be disrespectful towards that work, but this project seems to be a massive undertaking, and seemingly one that may never end. Given that observation, would the total gain this project would realize be worth the hours upon hours of hard work and time put into it?

- Are the methods described in the text unfeasible for lower resource languages?

Reading questions

- The text says that Redwoods can run text from sources like Wikipedia, the Wall Street Journal, and others--usually formal/professional. How might it handle informal texts, like tweets riddled with slang words and acronyms? The fact that it's hand annotated seems like it'd be hard to keep up with how quickly language changes over the internet.

- How old can texts in a corpus/treebank be before they no longer accurately represent modern language? Have there been cases yet of a corpus/treebank falling out of use because it is out of date?

Reading questions

- In section 2, it claims that treebanking supports grammar development rather than being a distraction to it (page 5). Why would it ever be considered a distraction to grammar development? Is it not necessary?

Reading questions

- In the 4th section, the paper discussed the difficulty on treebanking a new corpora and how to make them consistent with each other. I am wondering whether previous corpora had been created using Amazon Mechanical Turk or similar services to distribute tedious, but important, work to less-experienced. I know this might not be true, but I am actually I bit worried after reading this paper and I had seen other people, even researchers, use similar strategies.

Reading questions

- I'm not sure if this was mentioned in the reading, but if there are multiple different annotations for the same string, would they all be maintained in a database simultaneously, or would the best annotation be chosen?

- The paper suggests syntactico-semantic annotations and their ability to be encoded in a machine-readable grammar - the Redwoods project seems to also focus on these annotations. Is this approach generalizable to other forms of annotations that may be important for different applications (named entity recognition, coreference resolution, POS tagging, etc)?

Reading questions

- Are there any large downsides to the redwood approach, and is it still used today or has it been filtered out by new methods?

- Are there any downsides/risks associated with larger treebanks that don't involve ambiguity?

Dagstuhl report-back

- => Lori’s slides

Compling/NLP in the news

- https://blog.google/technology/ai/google-io-2023-keynote-sundar-pichai/