Ling 566
Dec 7, 2023

Dependency Parsing, Final Preview

Overview

- Dependency grammar: High level overview
- Variation in dependency annotations
- Redwoods treebank
- Exporting dependencies from HPSG analyses
- MRP shared tasks
- Reading questions
- Final preview

Grammatical Dependencies

- Relate words in the sentence to each other
- A labeled with the type of dependency
- Are typically represented as graphs (sometimes trees)
  - Where each node is a word in the sentence
  - Where word in the sentence is (usually) a node

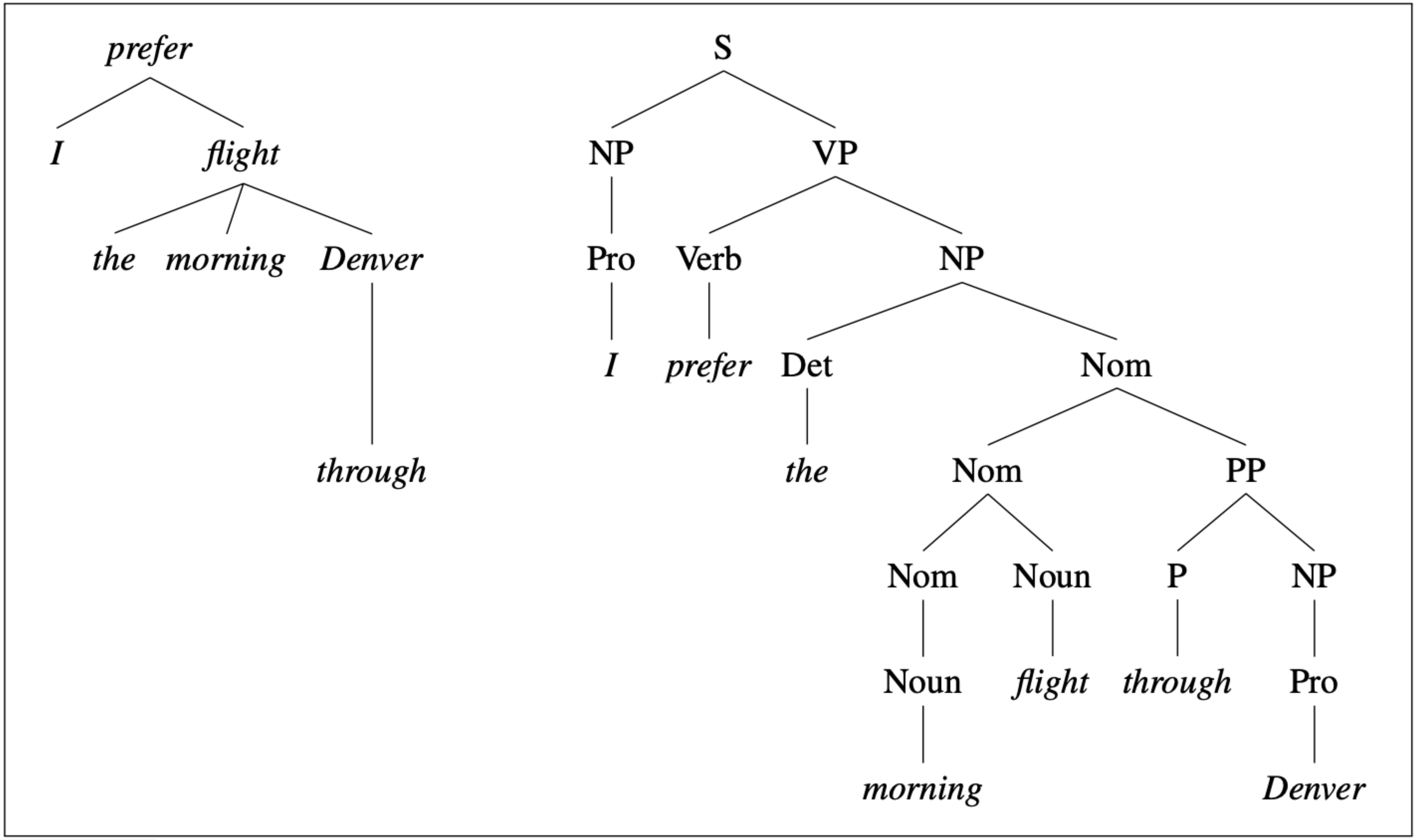
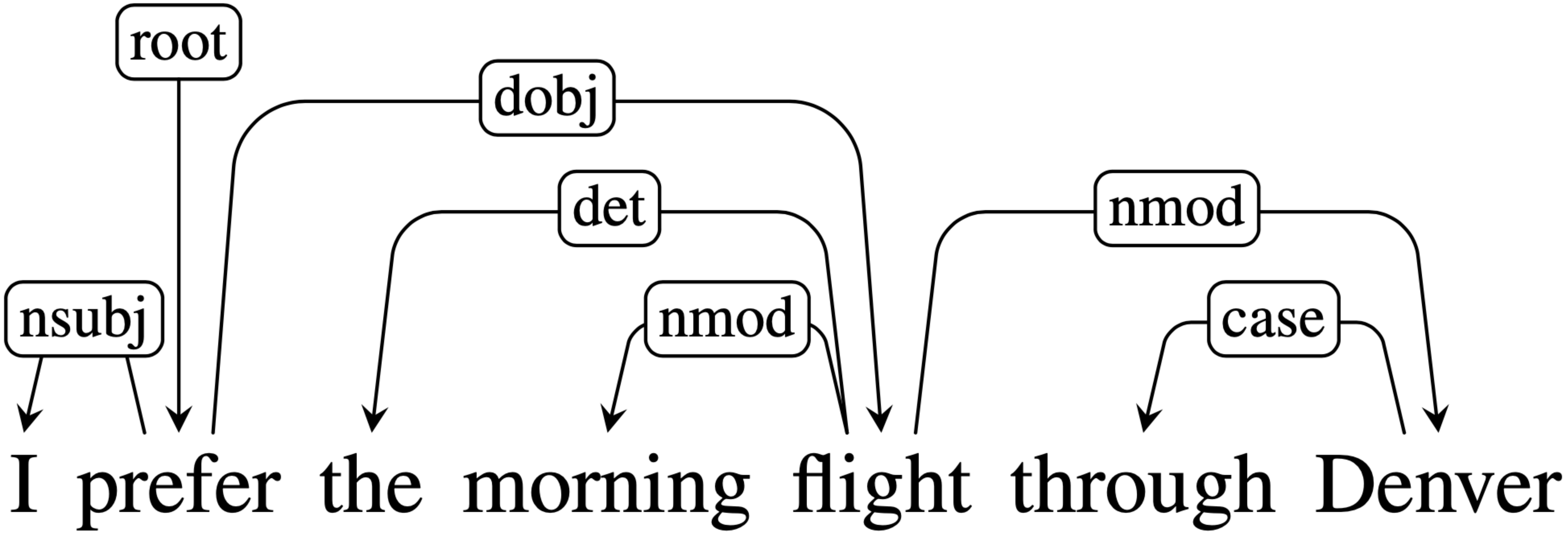


Figure 15.1 A dependency-style parse alongside the corresponding constituent-based analysis for *I prefer the morning flight through Denver.*


root
dobj
det
nmod
nsubj
nmod
case
I prefer the morning flight through Denver

Dependency Grammar

- Theoretical foundations: Tesnière 1959, Mel’čuk 1988, Hudson 1984, Sgall et al 1986

- Focus not on grammaticality (“What’s a possible sentence?”) but on grammatical structure, given a string

Dependency parsing, popularity of

- Very fast algorithms
- Dependency trees tend to be closer to semantics and therefore more useful in applications than phrase structure trees
- UD project produces really appealing datasets

Dependency Treebanks: Universal Dependencies

- https://universaldependencies.org/
- Builds on:
  - Stanford dependencies (LFG-inspired transformation of CFG representations for English from the Stanford parser)
  - Theoretical work on dependency grammar
  - “Universal” POS tagset developed initially for cross-linguistic error analysis (McDonald and Nivre 2007)

What is needed for UD to be successful? (from universaldependencies.org/introduction.html)

- The secret to understanding the design and current success of UD is to realize that the design is a very subtle compromise between approximately 6 things:
    - UD needs to be satisfactory on linguistic analysis grounds for individual languages.
    - UD needs to be good for linguistic typology, i.e., providing a suitable basis for bringing out cross-linguistic parallelism across languages and language families.
    - UD must be suitable for rapid, consistent annotation by a human annotator.
    - UD must be suitable for computer parsing with high accuracy.
    - UD must be easily comprehended and used by a non-linguist, whether a language learner or an engineer with prosaic needs for language processing. We refer to this as seeking a habitable design, and it leads us to favor traditional grammar notions and terminology.
    - UD must support well downstream language understanding tasks (relation extraction, reading comprehension, machine translation, …).
- It's easy to come up with a proposal that improves UD on one of these dimensions. The interesting and difficult part is to improve UD while remaining sensitive to all these dimensions.

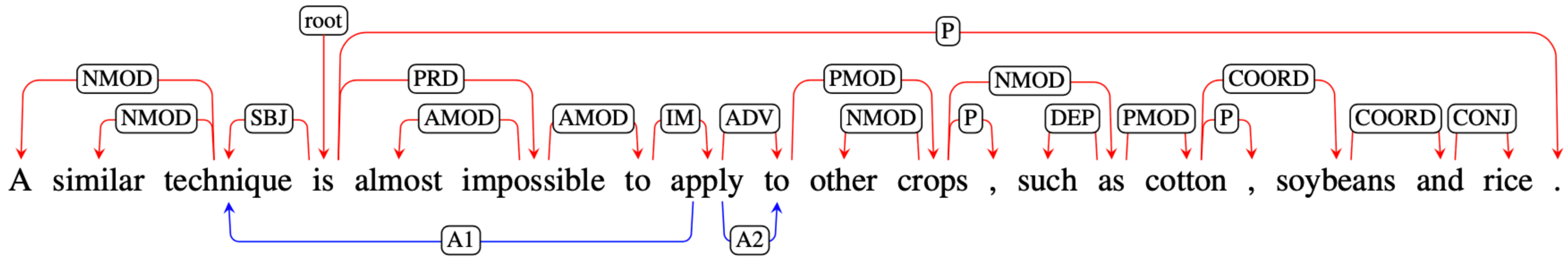
“Typical” dependency structures (quoted from Ivanova et al 2012, p.3)

- Dependency structures are directed trees: labeled, directed graphs, where the word tokens in a sentence constitute the nodes, and
  - (i) every token in the sentence is a node in the graph (combined with a designated root node, conventionally numbered as 0),
  - (ii) the graph is (weakly) connected,
  - (iii) every node in the graph has at most one head, and
  - (iv) the graph is acyclic (Nivre et al., 2007).

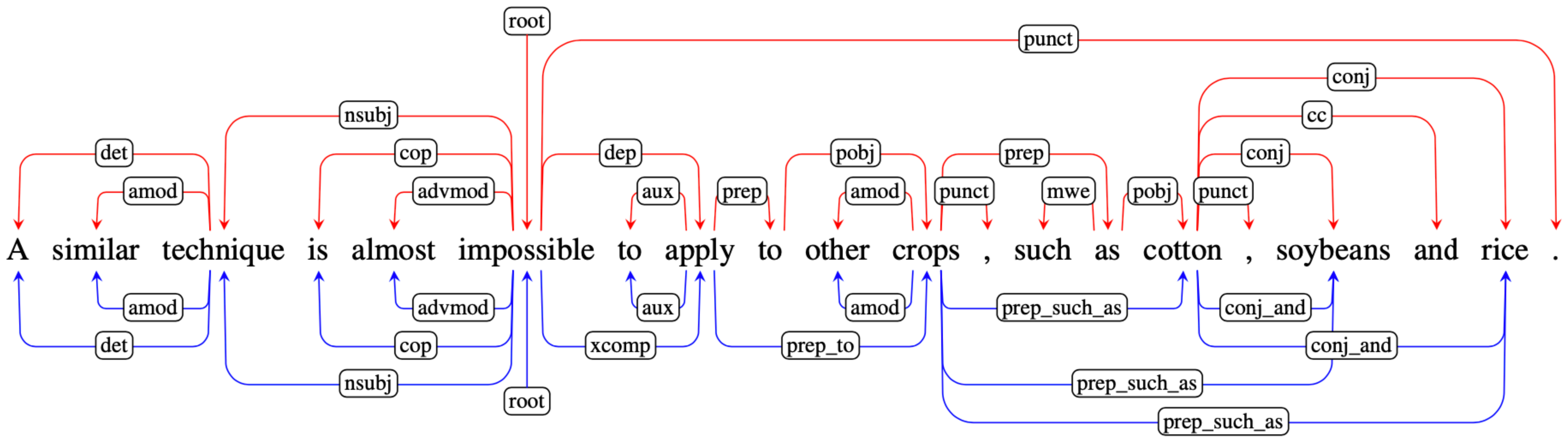
Variation in dependency structure standards

- Head selection (functional v. substantive)
- Dependency inventory
- Formal constraints on admissible graphs
- Bilexical dependencies v. allowing abstract nodes
- Are all words in the string accounted for?

Variation in dependency structure standards (Ivanova et al 2012, Fig 3)

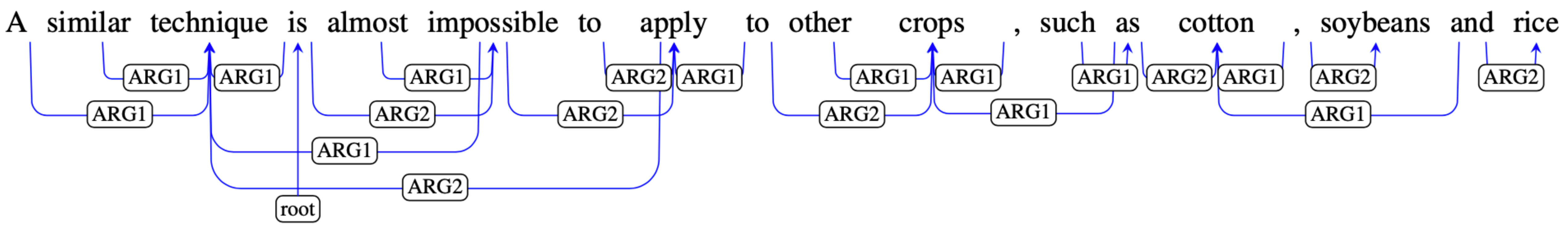

root
P
NMOD
NMOD
SBJ
PRD
AMOD
AMOD
IM
ADV
PMOD
NMOD
NMOD
P
DEP
PMOD
COORD
P
COORD
CONJ
A similar technique is almost impossible to apply to other crops , such as cotton , soybeans and rice .
A1
A2

(a) CoNLL 2008 *syntactic dependencies* (CD; top) and *propositional semantics* (CP; bottom).

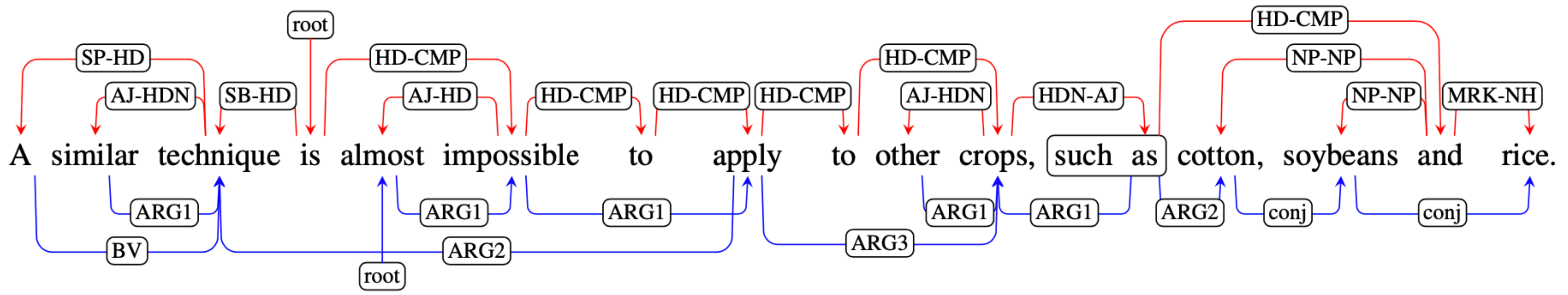


(b) Stanford Dependencies, in the so-called *basic* (SB; top) and *collapsed & propagated* (SD; bottom) variants.

Variation in dependency structure standards (Ivanova et al 2012, Fig 3)


A similar technique is almost impossible to apply to other crops , such as cotton , soybeans and rice
ARG1
ARG2
ARG3
root

(c) Enju *predicate-argument structures* (EP).


A similar technique is almost impossible to apply to other crops, such as cotton, soybeans and rice.
SP-HD
AJ-HDN
SB-HD
root
HD-CMP
AJ-HD
HDN-AJ
NP-NP
MRK-NH
ARG1
ARG2
ARG3
BV
conj

(d) DELPH-IN *syntactic derivation tree* (DT; top) and *Minimal Recursion Semantics* (DM; bottom).

Figure 3: Dependency representations in (a) CoNLL, (b) Stanford, (c) Enju, and (d) DELPH-IN formats.

RQ: Why do so many schemes put the root at *impossible*? Or *almost*?

Overview

- Dependency grammar: High level overview
- Variation in dependency annotations
- Redwoods treebank
- Exporting dependencies from HPSG analyses
- MRP shared tasks
- Reading questions
- Final preview

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

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.
- As illustrated in the following figures from Flickinger et al 2017

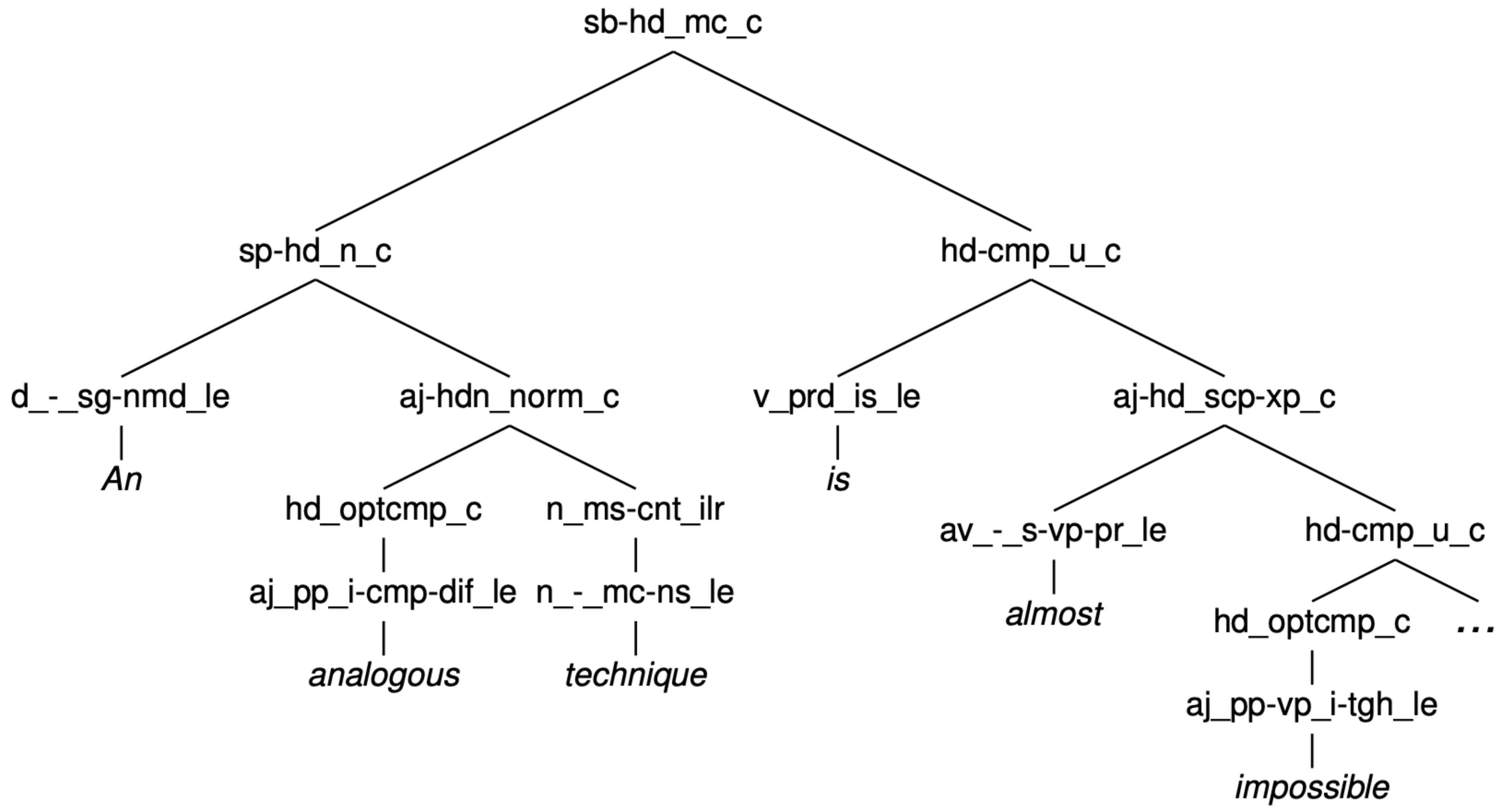


Fig. 1 ERG derivation tree for example (1).

RQ: From Figure 1, we see that a snippet from the ERG has two non-branching nodes. Aren't we trying to avoid this?

$$\begin{bmatrix} \textit{hcomp\_rule} & \\ \text{CAT} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{prep} & \\ \text{MOD} & \left\langle \begin{bmatrix} \text{CAT} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{v\_or\_a\_or\_p} & \\ \text{AUX} & - \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle [\,] \rangle \\ \text{COMPS} & \langle\,\rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \right\rangle \\ \text{PRD} & + \\ \text{CASE} & \textit{obliq} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \left\langle \begin{bmatrix} \text{HEAD} & \textit{n\_or\_adv} \\ \text{HOOK} & \begin{bmatrix} \text{LTOP} & \boxed{6}\textit{handle} \\ \text{XARG} & \boxed{7}\textit{non\_conj\_event} \end{bmatrix} \end{bmatrix} \right\rangle \\ \text{COMPS} & \langle\,\rangle \end{bmatrix} \\ \text{POSTHD} & + \\ \text{NEGPOL} & \text{-} \end{bmatrix} \\ & \begin{bmatrix} \textit{mrs} \end{bmatrix} \end{bmatrix}$$

$$\left|\begin{array}{ll} \text{CONT} & \begin{bmatrix} \textit{mrs} & \\ \text{HOOK} & \begin{bmatrix} \text{LTOP} & \boxed{6} \\ \text{INDEX} & \boxed{7} \\ \text{XARG} & \boxed{1} \end{bmatrix} \\ \text{RELS} & \left\langle \begin{array}{l} \begin{bmatrix} \textit{prep\_relation} & \\ \text{PRED} & \textit{\_to\_p\_rel} \\ \text{LBL} & \boxed{6} \\ \text{ARG0} & \boxed{7} \\ \text{ARG1} & \boxed{1} \\ \text{ARG2} & \boxed{13}\begin{bmatrix} \textit{nonconj\_ref-ind} & \\ \text{PNG.PN} & \textit{3p} \end{bmatrix} \end{bmatrix}, \begin{bmatrix} \textit{adj\_relation} & \\ \text{PRED} & \textit{\_other\_a\_1\_rel} \\ \text{LBL} & \boxed{18}\,\textit{handle} \\ \text{ARG0} & \begin{bmatrix} \textit{non\_conj\_event} & \\ \text{TENSE} & \textit{no\_tense} \end{bmatrix} \\ \text{ARG1} & \boxed{13} \end{bmatrix}, \\ \begin{bmatrix} \textit{quant\_or\_wh\_relation} & \\ \text{PRED} & \textit{udef\_q\_rel} \\ \text{LBL} & \textit{handle} \\ \text{BV} & \boxed{13} \\ \text{RSTR} & \boxed{16}\,\textit{handle} \\ \text{BODY} & \textit{handle} \end{bmatrix}, \begin{bmatrix} \textit{reg\_nom\_relation} & \\ \text{PRED} & \textit{\_crop\_n\_1\_rel} \\ \text{LBL} & \boxed{18} \\ \text{ARG0} & \boxed{13} \end{bmatrix} \end{array} \right\rangle \end{bmatrix} \end{array}\right\|$$

$$\left[\;\left[\begin{array}{ll} & \left[\begin{array}{ll}\text{LBL} & \textit{handle}\\ \text{BV} & \boxed{13}\\ \text{RSTR} & \boxed{16}\,\textit{handle}\\ \text{BODY} & \textit{handle}\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \textit{\_crop\_n\_1\_rel}\\ \text{LBL} & \boxed{18}\\ \text{ARG0} & \boxed{13}\end{array}\right]\\ \text{HCONS} & \left\langle\left[\begin{array}{ll}\textit{qeq} & \\ \text{HARG} & \boxed{16}\\ \text{LARG} & \boxed{18}\end{array}\right]\right\rangle\end{array}\right]\;\right]$$

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

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

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

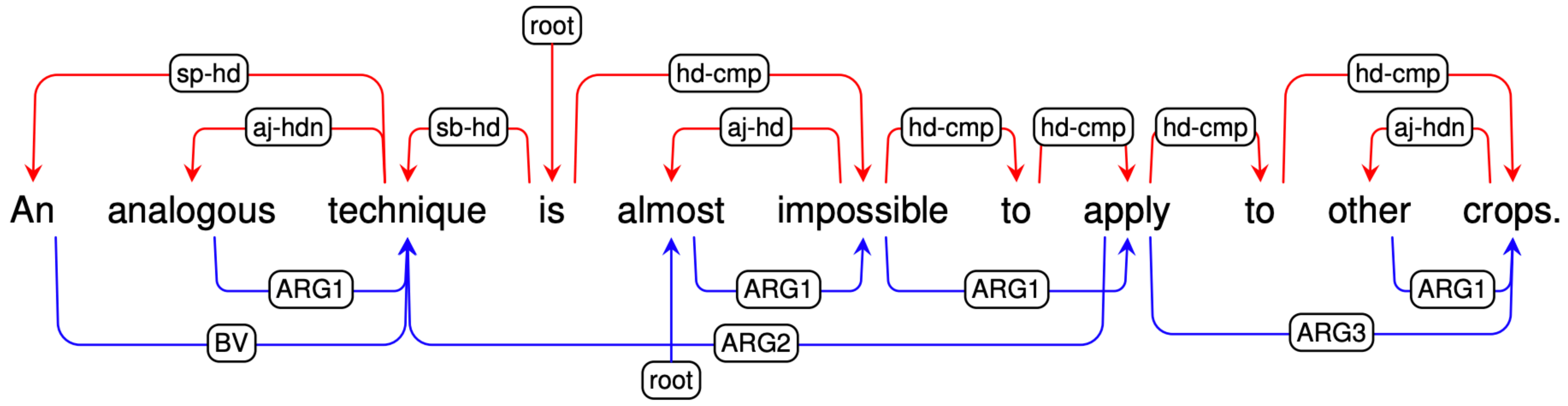


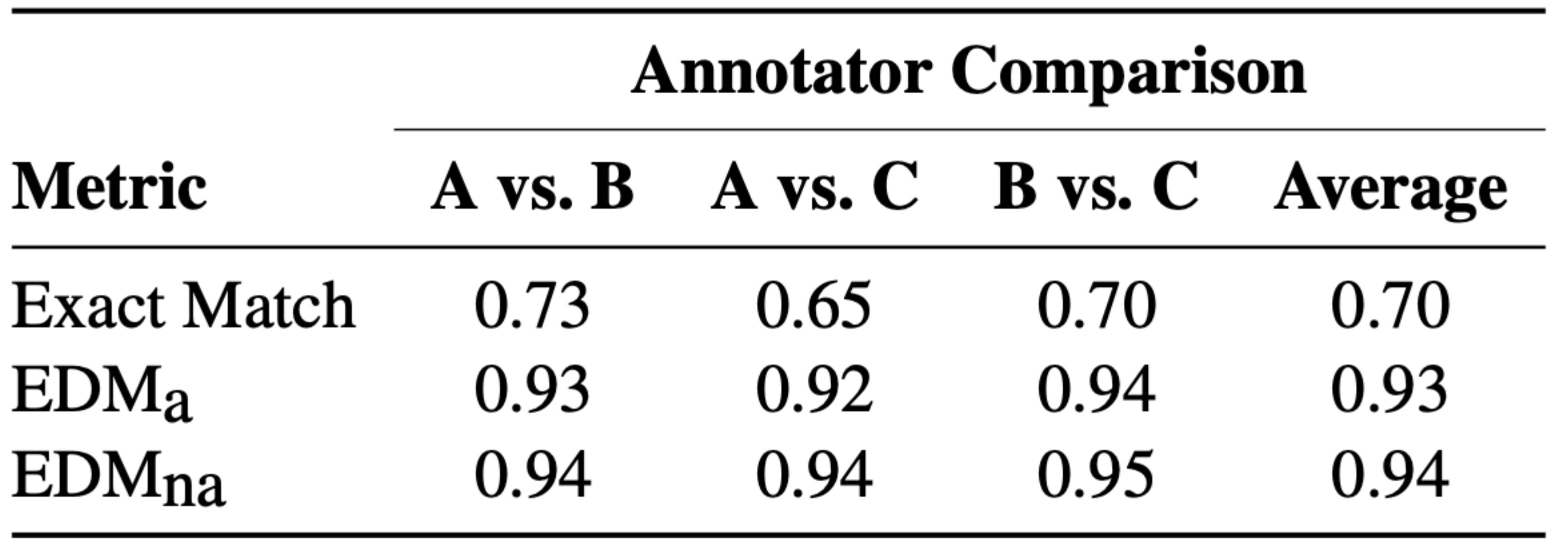
Fig. 4 Bi-lexical syntactic and semantic dependencies for (1).

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

Automatic conversion: Syntactic dependencies (Ivanova et al 2012)

- Remove (collapse) non-branching rules
- Head daughter is marked in (most?) ERG rules
- Dependencies labels come from valence features

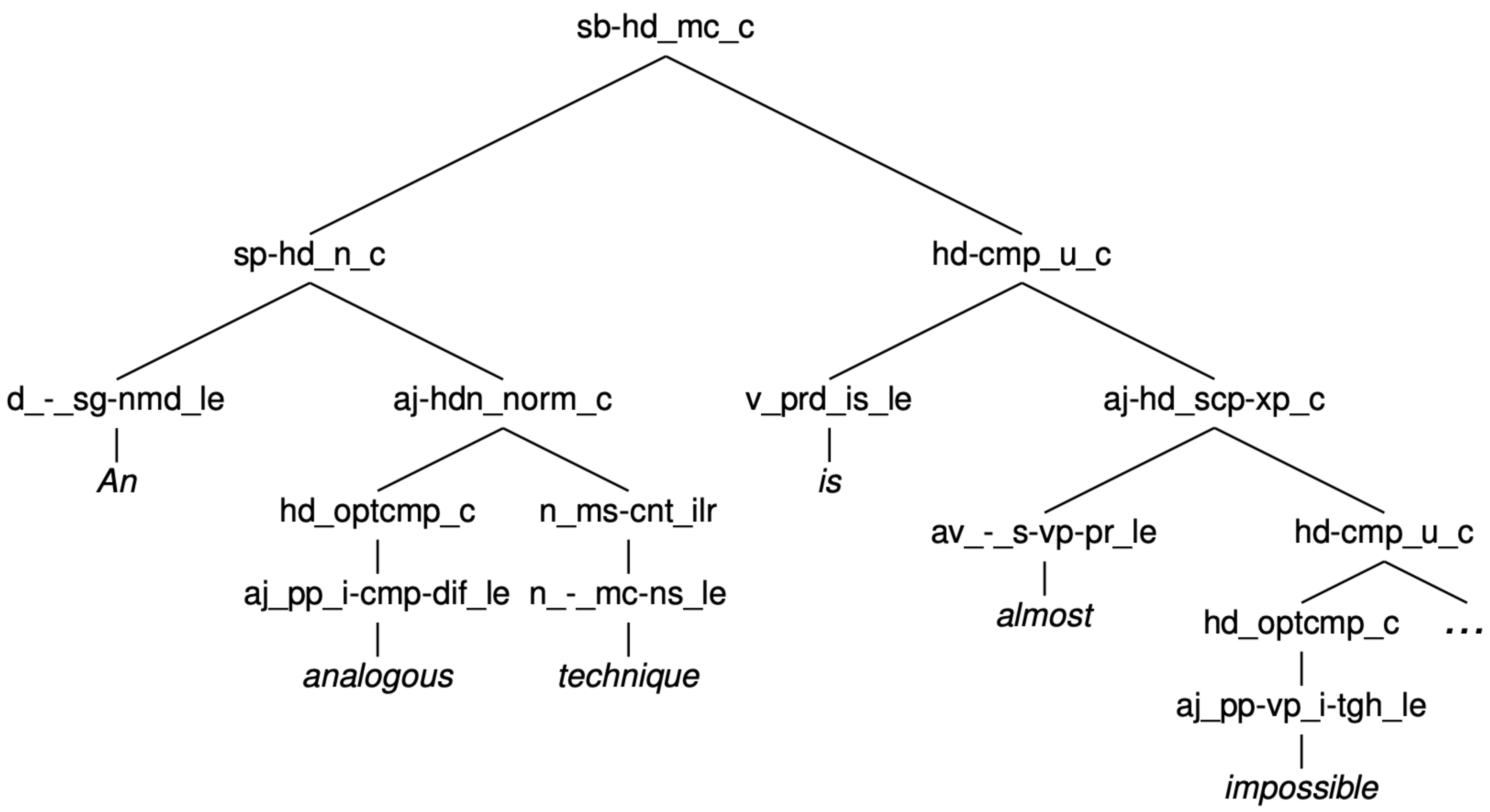


Fig. 1 ERG derivation tree for example (1).

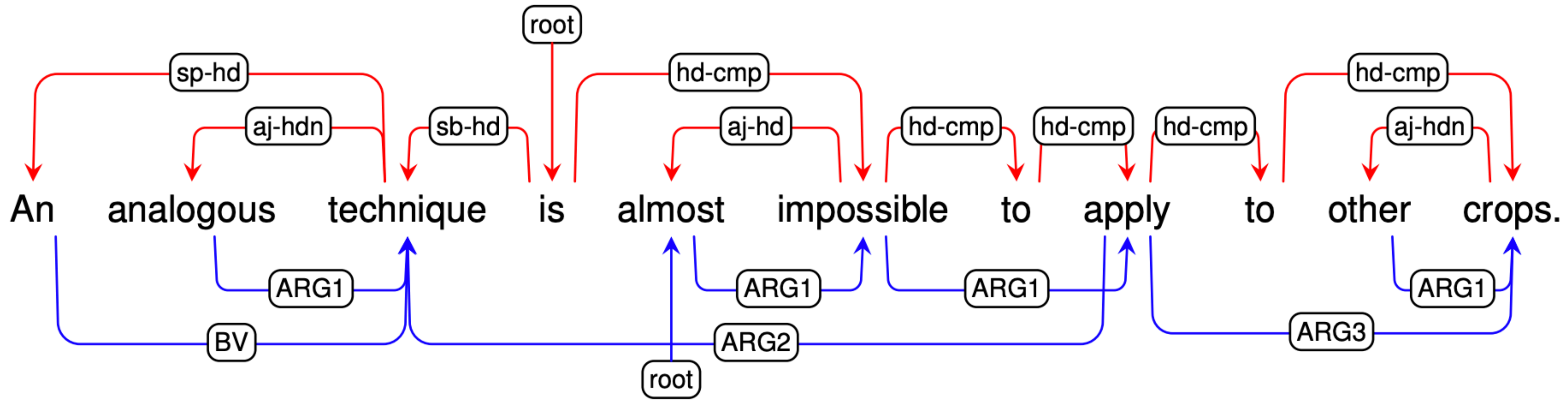
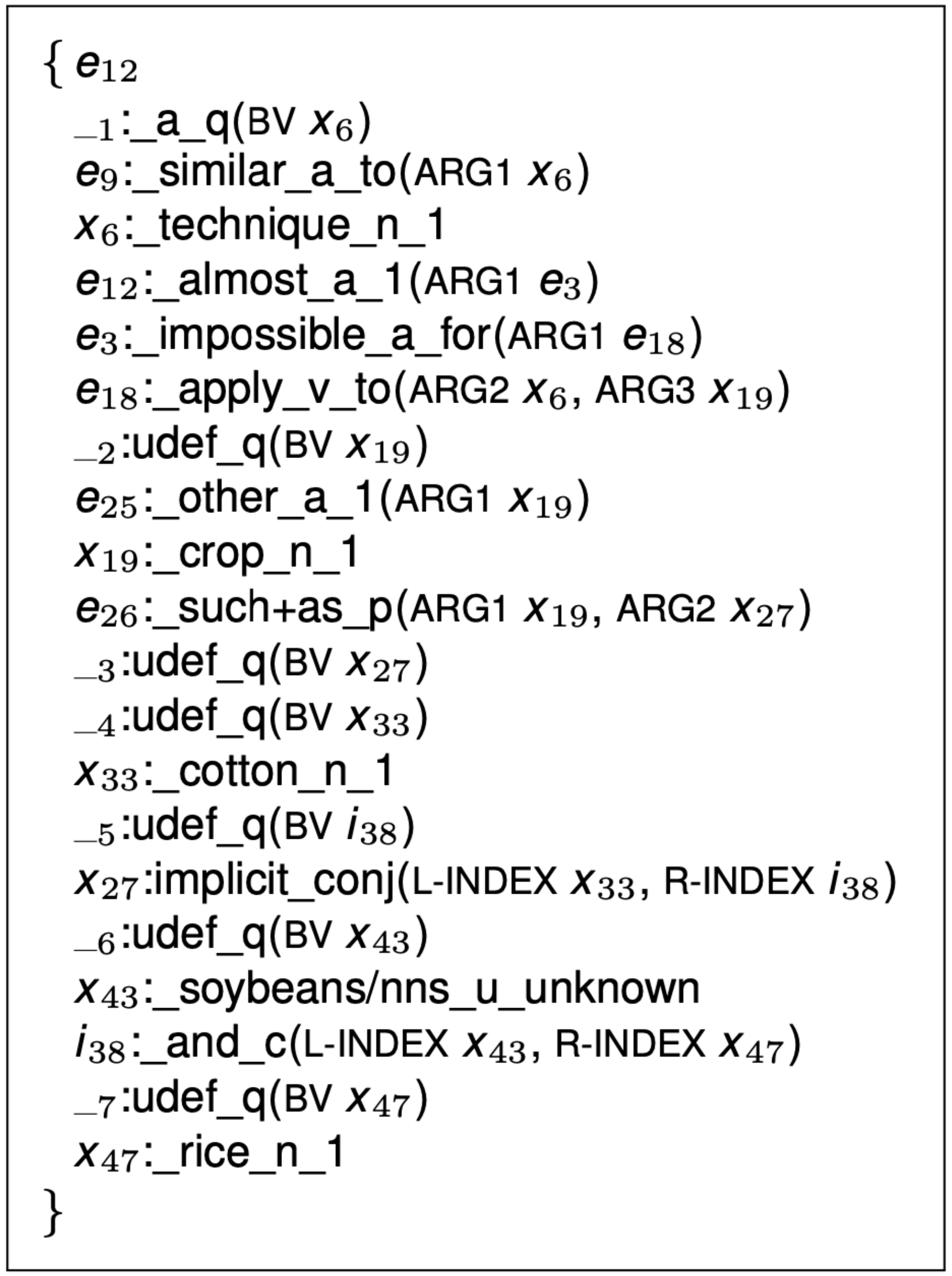


Fig. 4 Bi-lexical syntactic and semantic dependencies for (1).

Automatic conversion: Semantic dependencies (Ivanova et al 2012)

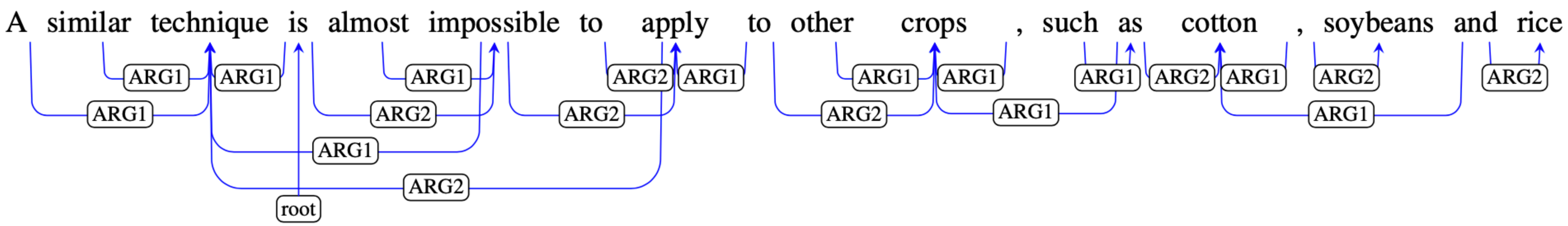
- Lossy conversion of MRS to EDS (elementary dependency structures, Oepen & Lønning 2006)
- Regular predicates: ARGn dependency between tokens
- Transparent and redundant predicates: remove
- Relational predicates: predicate is dependency label



{ e_{12}
$\_{1}$:_a_q(BV x_6)
e_9:_similar_a_to(ARG1 x_6)
x_6:_technique_n_1
e_{12}:_almost_a_1(ARG1 e_3)
e_3:_impossible_a_for(ARG1 e_{18})
e_{18}:_apply_v_to(ARG2 x_6, ARG3 x_{19})
$\_{2}$:udef_q(BV x_{19})
e_{25}:_other_a_1(ARG1 x_{19})
x_{19}:_crop_n_1
e_{26}:_such+as_p(ARG1 x_{19}, ARG2 x_{27})
$\_{3}$:udef_q(BV x_{27})
$\_{4}$:udef_q(BV x_{33})
x_{33}:_cotton_n_1
$\_{5}$:udef_q(BV i_{38})
x_{27}:implicit_conj(L-INDEX x_{33}, R-INDEX i_{38})
$\_{6}$:udef_q(BV x_{43})
x_{43}:_soybeans/nns_u_unknown
i_{38}:_and_c(L-INDEX x_{43}, R-INDEX x_{47})
$\_{7}$:udef_q(BV x_{47})
x_{47}:_rice_n_1
}

Figure 2: ERG Elementary Dependency Structure.

Variation in dependency structure standards (Ivanova et al 2012, Fig 3)


A similar technique is almost impossible to apply to other crops , such as cotton , soybeans and rice
ARG1
ARG2
ARG3
root
(c) Enju *predicate-argument structures* (EP).

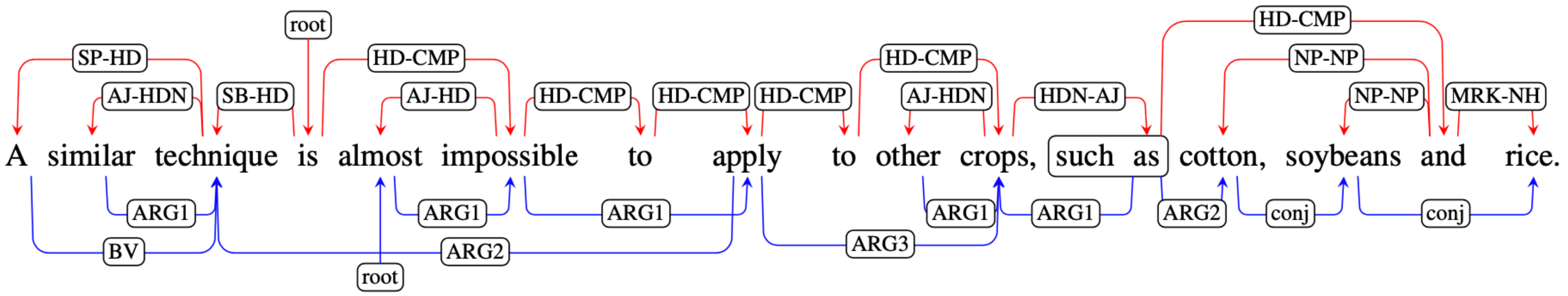

A similar technique is almost impossible to apply to other crops, such as cotton, soybeans and rice.
SP-HD
AJ-HDN
SB-HD
root
HD-CMP
AJ-HD
HDN-AJ
NP-NP
MRK-NH
ARG1
ARG2
ARG3
BV
conj
(d) DELPH-IN *syntactic derivation tree* (DT; top) and *Minimal Recursion Semantics* (DM; bottom).

Figure 3: Dependency representations in (a) CoNLL, (b) Stanford, (c) Enju, and (d) DELPH-IN formats.

MRP (meaning representation parsing) shared tasks

- http://mrp.nlpl.eu/2020/index.php

-

Overview

- Dependency grammar: High level overview
- Variation in dependency annotations
- Redwoods treebank
- Exporting dependencies from HPSG analyses
- MRP shared tasks
- Reading questions
- Final preview

RQs: Crossframework comparison

- Are these frameworks trying to do the same thing? If so, should they ideally produce the same output? Is there a correct output?
- It's interesting to learn about PEST - in order for collections like these to expand, are they always annotated by the same group of organizers or are they passed through various parsers first then manually checked?

RQs: Cyclic structures

- How can there be cycles in a graph that represents a human sentence? How does this logically make sense?

RQs: Dataset representativeness

- The dataset involved in the contrasting analyses seems to be small - only ten PEST sentences. Would it be sufficient to draw conclusions like "The DELPH-IN dependency representations demonstrate comparatively strong interoperability with other schemes, since CD corresponds well with DT syntactically, while EP correlates with DM among the more semantic formats" based on only these ten sentences? Would the conclusions generalize well to analyses on other different sentences?

- It says PEST is a valuable resource due to its careful selection of grammatical phenomena, and I'm curious how linguists decide on which phenomena to examine. Are there any specific guidelines?

RQs: Punctuation

- Slightly tangential question about punctuation. The paper's discussion of punctuation tokenization contrasts (e.g., with punctuation as its own token vs punctuation included with the word token preceding it) between datasets reminded me that I have been wondering about the role of punctuation (if any) in our grammars. We've generally ignored it. It's something we obviously need to consider from a practical/implementation perspective, but does it matter at all from a theoretical one?

RQs: Crosslinguistic applicability

- All the information captured by the features of HPSG lexical entries seem like they would be quite useful for NLP models. Our class (and 567) both stress the importance of creating systems that work across languages and are not English-specific, but I wonder how many of the seven annotation schemes for this paper are implemented in other languages? Also, I can see how syntactic dependencies can be handled though automated computer systems, but are semantic dependencies able to be easily handled? It's harder for me to imagine how to automate semantics.

RQs: Reading technical papers

- Do you have any suggestions for getting better at reading and understanding papers like this? Doing it more seems like an obvious first step, but are there some techniques, tactics, or exercises that can help build this competency?
- Cf: Critical reading questions (writing in grad school Treehouse presentation)

RQs: And so?

- I don't quite understand the end conclusion of this paper. It compared 7 different annotation schemes/dependency formats, and in the end does it have a preference for one that is used? Or is it essentially that no matter which one is used that there is a way to convert to HPSG representation to make it all uniform? I also didn't quite get what happens when formats have differing results for what roots are in sentences and how to treat coordination, is it that HPSG blanket chooses one analysis and then converts to that analysis of the sentence? Or is it that whatever analysis the formats come up with, there will be an HPSG equivalent?