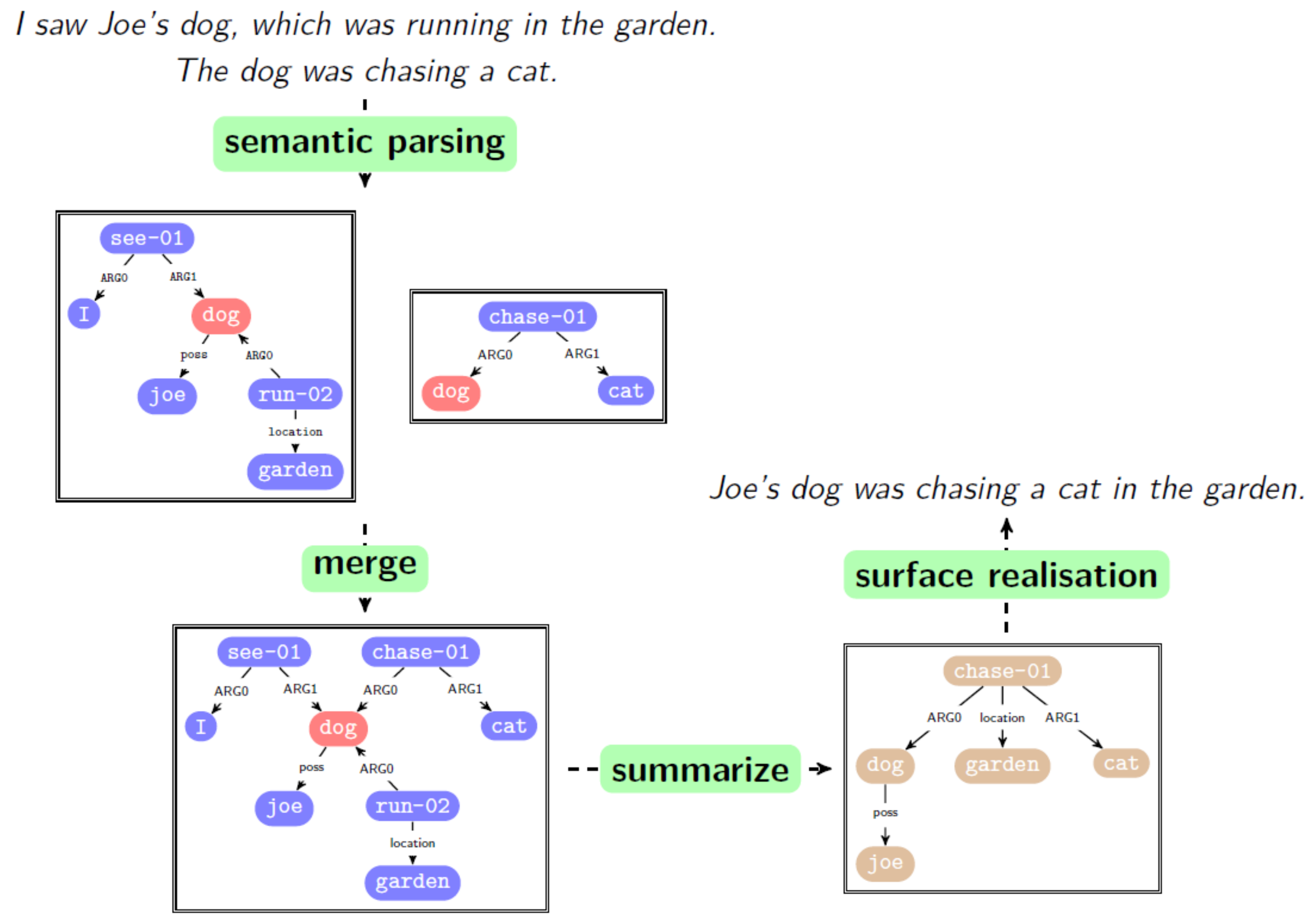
ACL 2019

Graph-Based Meaning Representations: Design and Processing

Semantic Parsing

- The task of mapping natural language sentences into complete formal meaning representations which a computer can execute for some domain-specific application.
- This view brings along a tacit expectation to map (more or less) directly from a linguistic surface form to an actionable encoding of its intended meaning, e.g. in a database query or even programming language.
- Who did What to Whom

Preview: Semantic Structure in Applications



Hardy & Vlachos (2018): 2$^+$ ROUGE points over strong encoder–decoder.

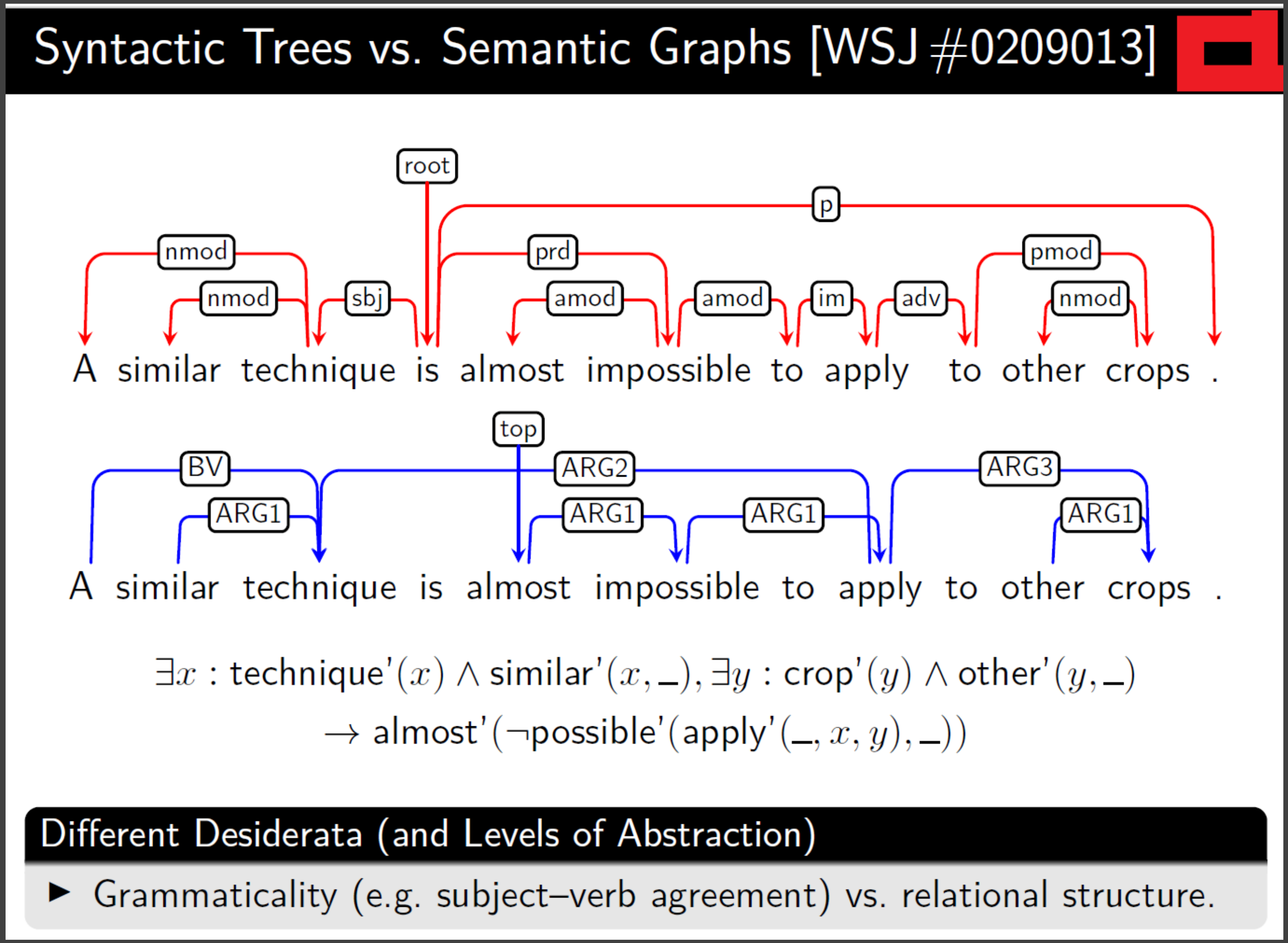
Outline

- Foundations: Linguistic & Formal
- Graph-Based Meaning Banks
- Parsing into Semantic Graphs
- Outlook: Using Semantic Graphs

Foundations : Semantics

- Linguistic semantics does not furnish a characterization of the interpretation of utterances in use, which is what one finally needs for natural language understanding applications—rather, it (mostly) provides a characterization of conventional content, that part of meaning determined by linguistic form. [1]

Foundations : Semantics



Foundations : Semantics

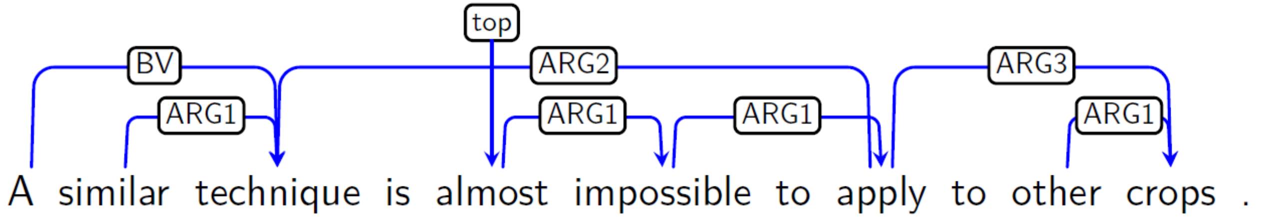
Trees: every node in the dependency graph is reachable from a distinguished root node by exactly one directed path. Trees are ill-suited for producing meaning representations

Semi-Formally: Trees vs. Graphs

Structural Wellformedness Conditions on Trees

- Unique root, connected, single parent, free of cycles; maybe: projective;

$\rightarrow$ all nodes (but the root) reachable by unique directed path from root.



Beyond Trees: General Graphs

- Argument sharing: nodes with multiple incoming edges (***in***-degree > 1);
- some surface tokens do not contribute meaning (many function words);
- (structurally) multi-rooted: more than one node with zero ***in***-degree;

$\rightarrow$ massive growth in modeling and algorithmic complexity (NP-complete).

Foundations : Semantics

there is now growing interest in general graphs as more expressive and arguably more adequate target structures for sentence-level grammatical analysis beyond surface syntax and in particular for the representation of semantic structure.

Terminology: Syntactic vs. Semantic 'Predicates'

Syntax: Head–Dependent Relations

- Heads license and govern, e.g. verbs, relational nouns, prepositions;
- complements include subjects, objects, obliques, clausal arguments;
- adjuncts add information, e.g. adjectives, adverbials, relative clauses:

Abrams bet $10 on Monday that it rained in Florence.

Semantics: Predicate–Argument Relations

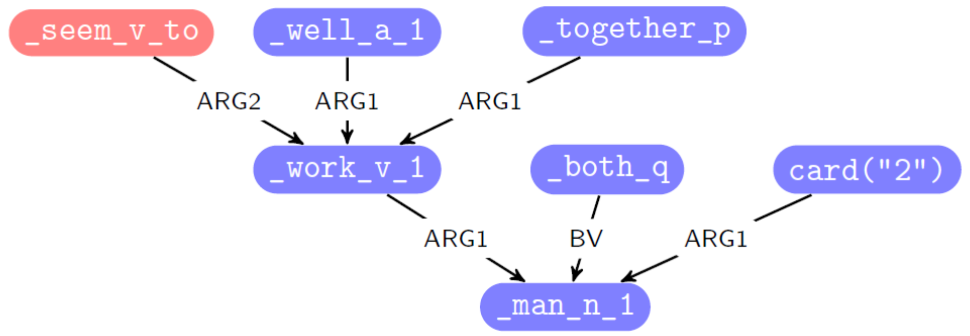
- Predicates evoke relations of variable arity; from all major word classes;
- arguments fill semantic roles, defined relative to each specific predicate;

Abrams bet $10 on Monday that it rained in Florence.

- Prepositions *on* or *in* as two-place relations, e.g. temporal or locative.

Foundations : Semantics

Mismatches between Syntax and Semantics



Both men seem to work well together. [WSJ #0109043]

- Discord between head–dependent and predicate–argument relations;
- syntactic subject of *seem* is a semantic argument of its complement;
- consider close paraphrase: *It seems that both men work well together.*
- expletive *it* subject is not referential, hence no semantic contribution;
- about two dozen subject raising verbs in broad-coverage English lexicon.

Facets of Linguistic Meaning

- Predicate–argument structure
- Quantification and Scope
- Presupposition and focus
- Word sense differentiation
- Lexical decomposition
- Anaphoric coreference

- Grounding (in world; in picture; in Wikipedia; ...)
- Tense and aspect
- Information structure
- Discourse structure
- ... and many others ...

Foundations : Basic Graph Theory

$$\mathbb{G} = \langle N, E, T \rangle$$

- G is a directed graph: *N* is set of nodes; $E \subseteq N \times N$ is set of edges;
- $T \subseteq N$ is possibly empty set of top node(s): the 'main' predicate(s);
- ***in***- and ***out***-degree of $n \in N$ count edges to and from *n*; ***in*** = 0: root;
- top in *Abrams arrived quickly.* is *arrive*, but can be argument of *quick*;
- semantic graphs often multi-rooted: rootness just a structural property;
- a node *n* is reentrant if ***in***(*n*) > 1 (shared argument across predicates);
- cycles can occur: directed path from *m* to *n* and ('back') from *n* to *m*;
- G is connected if there is an undirected path between all pairs of nodes;
- G is a tree if $|T| = 1$ and there is a unique path to all other nodes.

Semantic GraphBanks

Flavor	Name	Example	Type of Anchoring
0	bilexical	SDP	nodes are sub-set of surface tokens
1	anchored	EDS	free node–sub-string correspondences
2	unanchored	AMR	no sub-string correspondences annotated

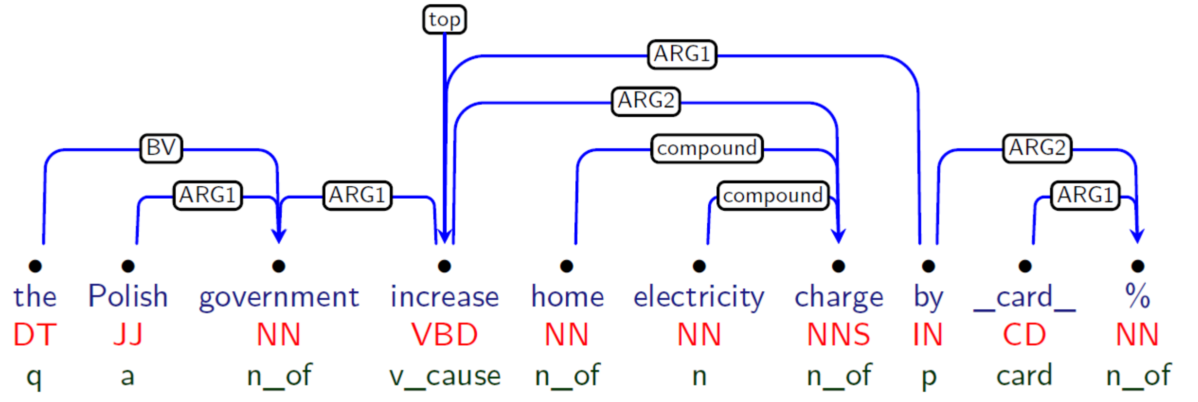
- Bilexical semantic dependencies: DM, PAS, PSD, CCD;
- Variants of English Resource Semantics: EDS, (DMRS, DM);
- Abstract Meaning Representations: AMR;
- Universal Conceptual Cognitive Annotation: UCCA.

Bi-lexical Dependency Graphs

- graph nodes injectively correspond to surface lexical units (tokens).

- Two decades of great advances in syntactic dependencies and parsing;
- recently, renewed interest in meaning; algorithmic interest in graphs;
- nodes limited to surface lexical units (words): lemmas, PoS, and frames;
- edges encode argument roles and maybe some construction semantics;
- limited expressivity, e.g. no lexical decomposition, no covert meaning.

The Polish government increased home electricity charges by 150%. [WSJ #0037059]



Bi-lexical Dependency Graphs

- This flavor of semantic graphs was popularized in part through a series of Semantic Dependency Parsing (SDP) [2,3]

Background

- Parallel work on bilexical semantic dependencies over the same corpus;
- different linguistic traditions, syntactico-semantic frameworks, schools.

SemEval 2014 (Oepen et al., 2014)

- Sentence- and token-align annotations; simplify into bilexical digraphs;
- 34,004 + 1348 sentences, 745,543 + 29,808 tokens from WSJ corpus;
- three frameworks: DM, PAS, PSD; nine participating teams: 78–92 F_1.

SemEval 2015 (Oepen et al., 2015)

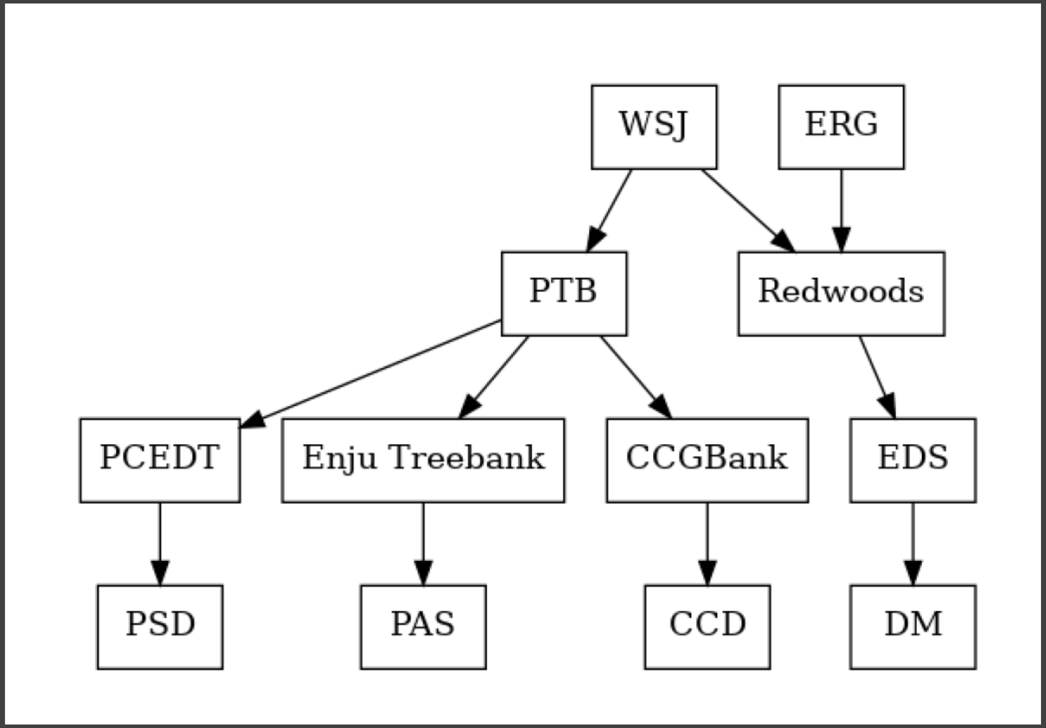
+ Out-of-domain English test data from Brown Corpus: 1,849; 31,583;
+ Chinese PAS (32,783; 687,433) and Czech PSD (48,972; 1,111,626);
+ frame (or sense) prediction; evaluation beyond ‘atomic’ dependency F_1.

More Recently (`http://sdp.delph-in.net`)

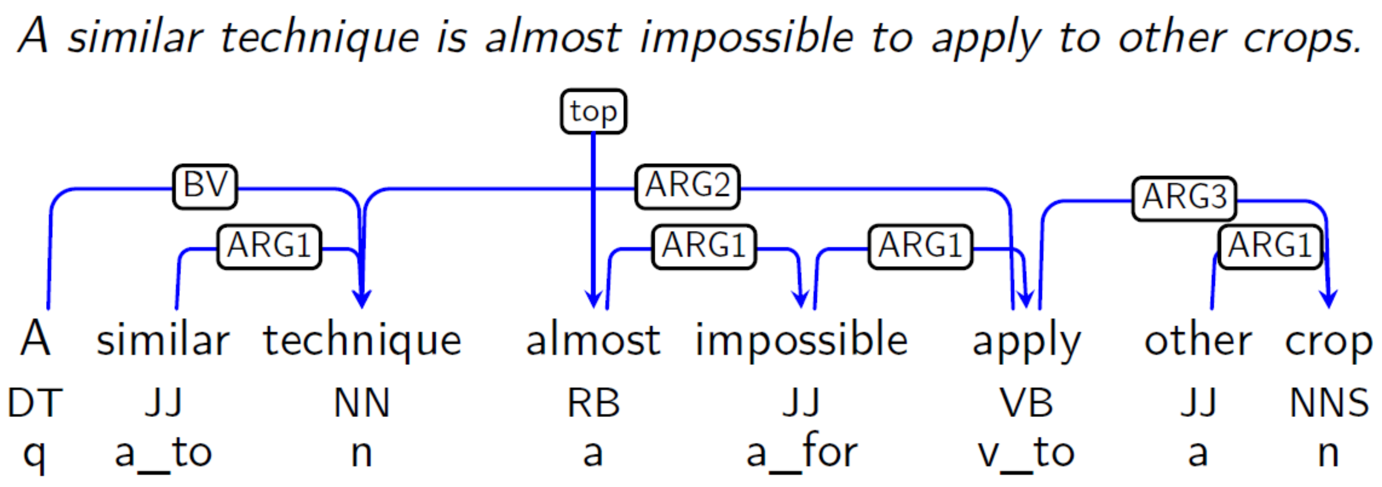
- Release as LDC2016T10 with fourth framework; ‘standard’ benchmark.

Bilexical Semantic Dependencies: English Genealogy

- PSD : Prague Semantic Dependencies[4]
- PAS : Enju Predicate–Argument Structures[5]
- CCD : CCG word–word dependencies[6]
- DM : DELPH-IN MRS Bi-Lexical Dependencies[7]



DM: DELPH-IN MRS Bi-Lexical Dependencies

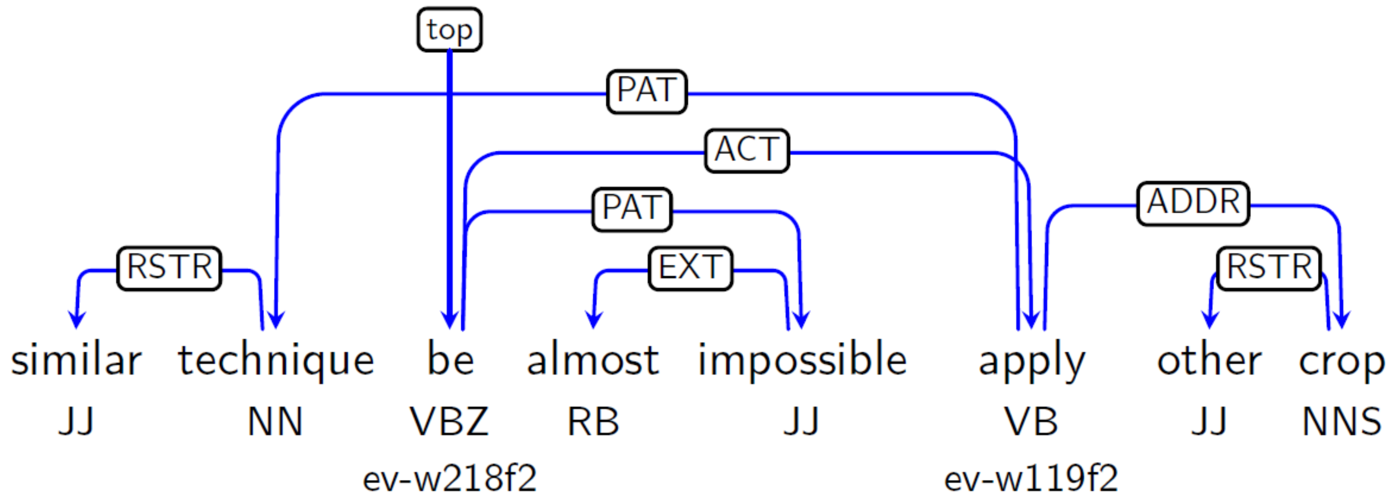


Ivanova et al. (2012)

- Simplification from underspecified logical forms (ERS; coming right up);
- edge labels are mostly semantic argument positions, e.g. *apply'*(_, _, _);
- PoS and frames for coarse sense differentiation (*apply* … *to* vs. … *for*);
- typically, unique top node; quantifiers have special 'bound variable' role.

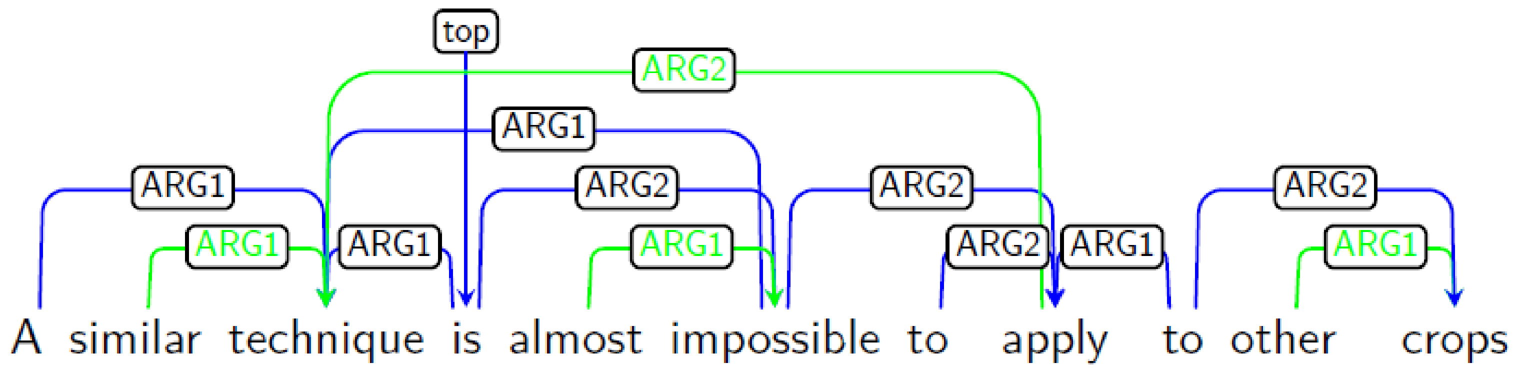
PSD:Prague Semantic Dependencies

A similar technique is almost impossible to apply to other crops.



- Simplification of multi-layer dependency annotation (coming right up);
- ACT(or), PAT(ient), ADDR(essee), ORIG(in), EFF(ect): complements;
- first verbal argument is ACT, also in causative–inchoative alternation;
- unlike DM, adjuncts are dependents, e.g. EXT(end) or RSTR(iction);
- simple determiners (*a*, *the*) treated as not content-bearning (in PSD).

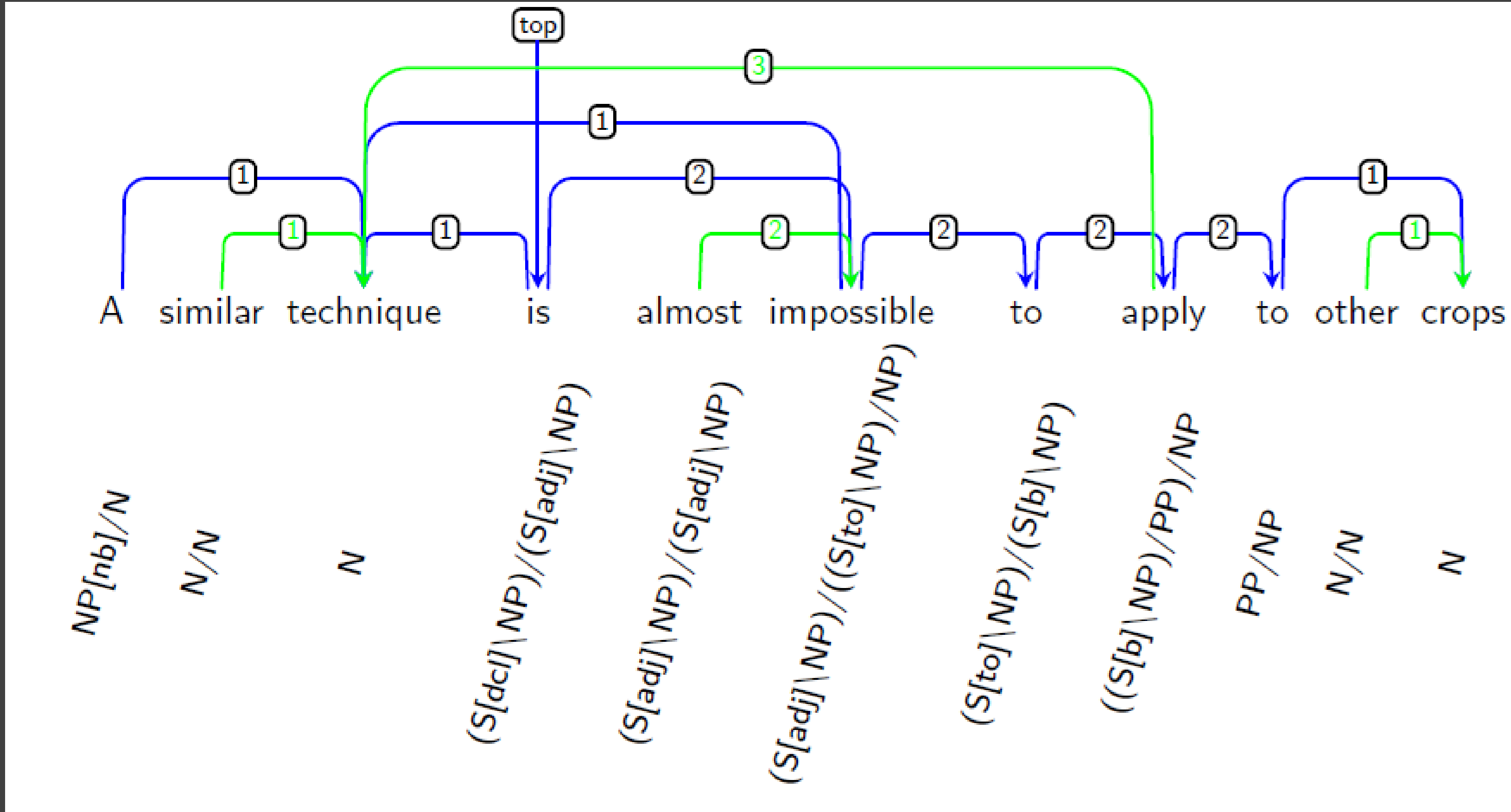
PAS: Enju Predicate–Argument Structures



PAS: Enju Predicate–Argument Structures (Miyao, 2006)

- Similar in pedigree to DM: derived compositionally by lage-scale HPSG;
- Enju Treebank: Mostly automatic conversion from PTB; limited syntax;
- missing lexical knowledge, e.g. ARG3 (oblique complement) of *apply*;
- like in CCD, several syntactic dependencies, e.g. *technique* as 'subject';
- Enju Parser was early broad-coverage engine for semantic dependencies.

CCD: CCG word–word dependencies(*)



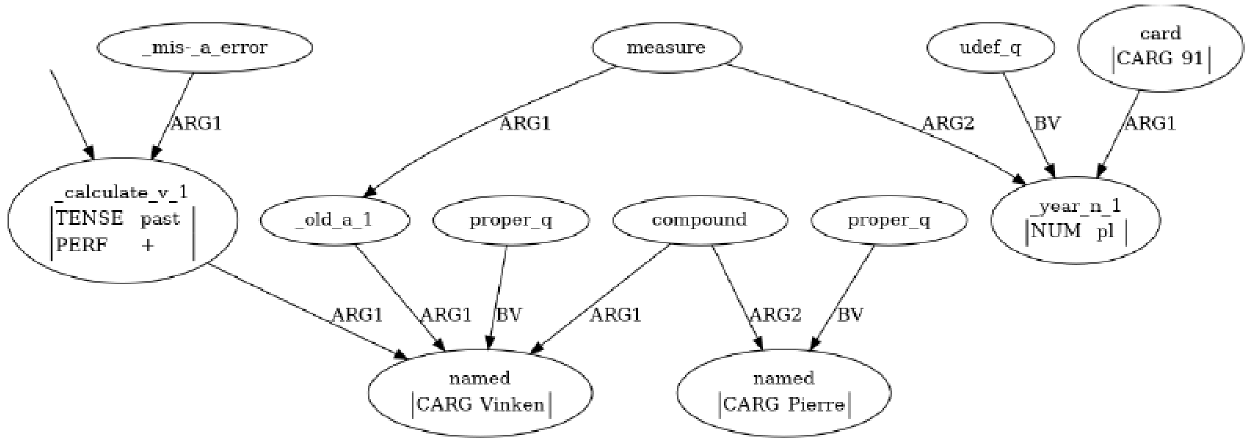
CCD: CCG Word–Word Dependencies (Hockenmaier & Steedman, 2007)

- CCG categories as 'frame' identifiers; edge labels for argument position;
- more 'deep syntax' than semantics, but functor–argument directionality.

Limitations of Bi-Lexical Semantic Dependencies

- Challenges :
  - lexical decomposition
  - sub-lexical
  - construction semantics.

- A more general form of **anchored semantic graphs** is characterized by relaxing the correspondence relations between nodes and tokens, while still explicitly annotating the correspondence between nodes and parts of the sentence.



Pierre Vinken, 91 years old, had mis-calculated.

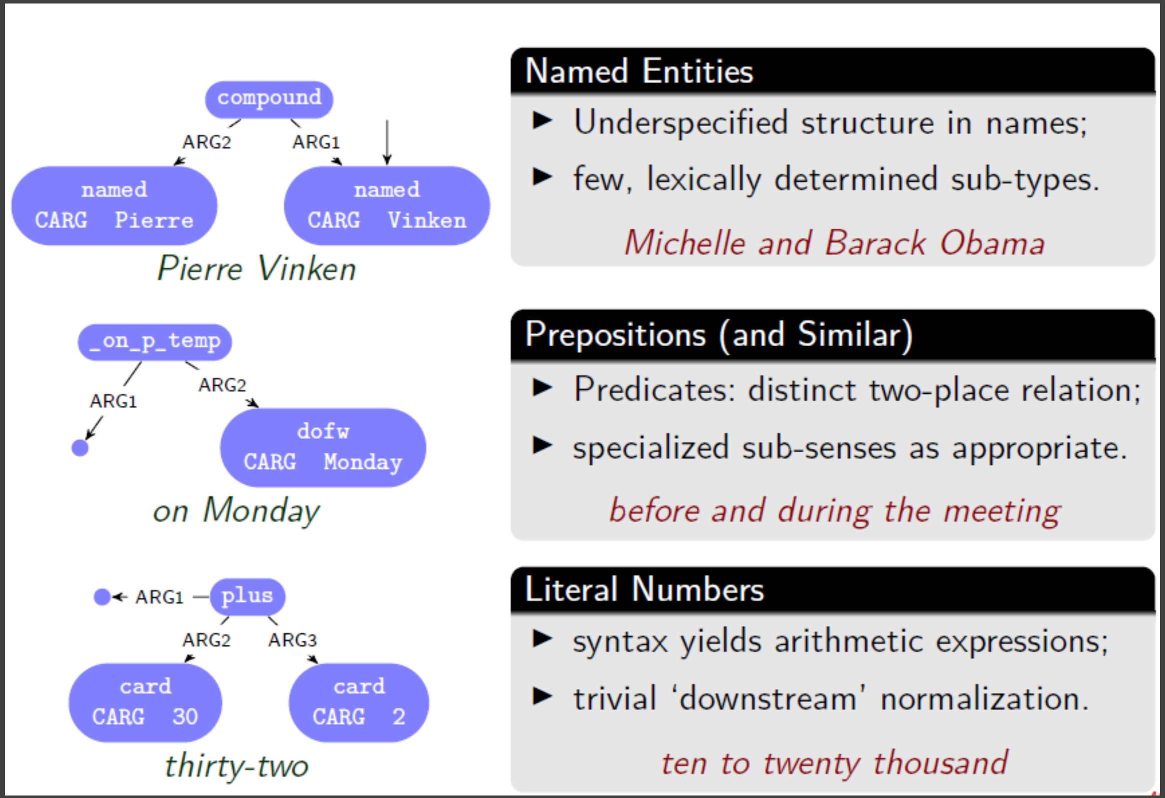
- ERS without variables (Oepen et al., 2002; Oepen & Lønning, 2006);
- like DM, nodes encode n-ary predications, edges (expressed) arguments;
- Flavor (1): arbitrary (overlapping) sub-strings carry pieces of meaning.

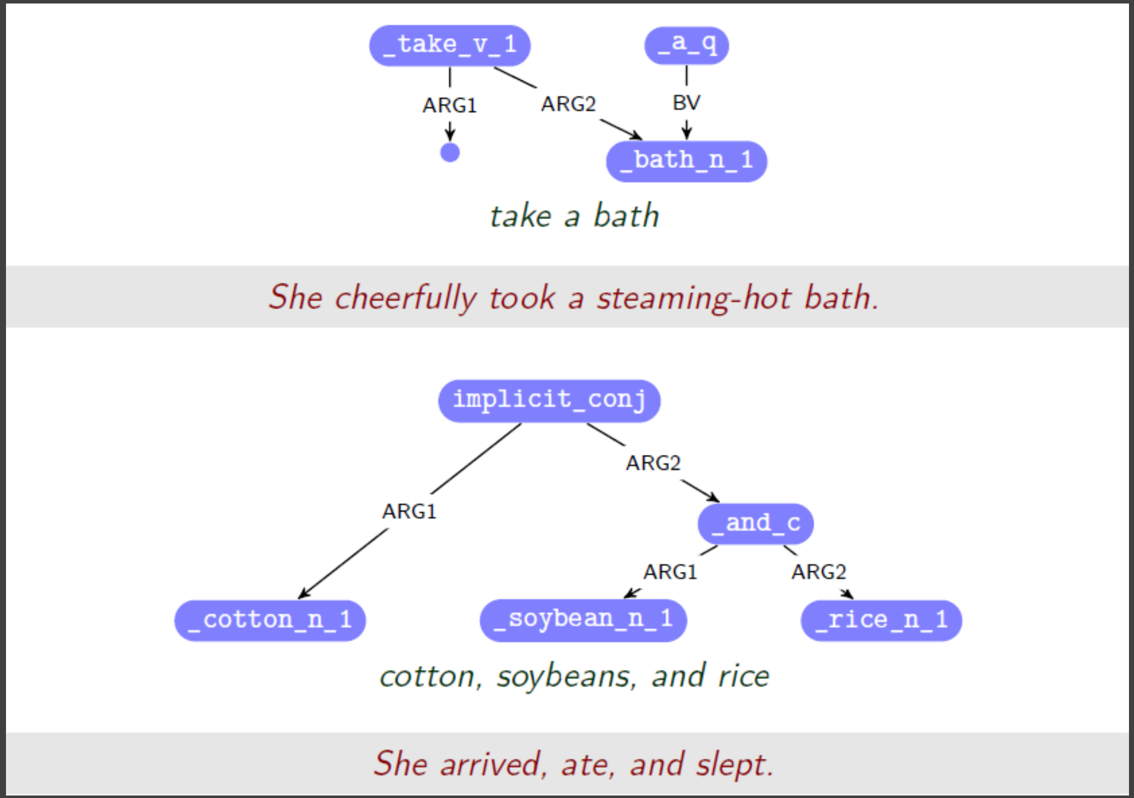
Anchored Semantic Graphs

- Some graph banks of this flavor align nodes with arbitrary parts of the sentence, including sub-token or multi-token sequences, which affords more flexibility in the representation of meaning contributed by, for example, (derivational) affixes or phrasal constructions. Some further allow multiple nodes to correspond to overlapping spans, enabling lexical decomposition (e.g. of causatives or comparatives).
- Examples:
  - UCCA : Universal Conceptual Cognitive Annotation [8]
  - EDS : Elementary Dependency Structures [9]
  - DMRS : Dependency Minimal Recursion Semantics [10]

EDS: Elementary Dependency Structures

- Compositionality in EDS





EDS: Elementary Dependency Structures

Sense Differentiation

- Focus on sentence meaning: dinstinguish grammaticalized contrasts:
- *look up the answer* (`_look_v_up`) vs. *look up the hill* (`_look_v_1`);
- *he broke the vase* (`_break_v_cause`) vs. *the vase broke* (`_break_v_1`);
- no grammatical contrast in *draw a house* vs. *draw a cart* (`_draw_v_1`).

Limitations

- Grammar-based annotation: no 'correct' analysis in 5–15 % of inputs;
- limited resources for other languages (German, Japanese, Spanish, . . .);
- tense, aspect, mood, number, etc. as 'morphological' node properties;
- partial information about scope discarded in conversion to EDS graphs.

Dependency Minimal Recursion Semantics (DMRS)

- Recall: Original ERSs contain partial, underspecified scope information;
- Copestake (2009) monotonically extends EDS with scopal 'overlays'.

AMR : Abstract Meaning Representation

- Goals:
  - Capture predicate–argument structure of a sentence.
  - Nodes annotated with lexically decomposed predicates, using PropBank senses.
  - Different sentences with same meaning should have the same AMR.
  - Use for NLU, NLG, machine translation.
- First large-scale hand-annotated sembank:
  - "Little Prince" pilot annotation, ~1500 sentences
  - AMRBank v1 (LDC2014T12), ~13k sentences
  - AMRBank v2 (LDC2017T10), ~40k sentences, includes v1
  - ISI, since 2013 (Banarescu et al., 2013)
- Inter-annotator agreement: Typical Smatch scores of 70–80.

AMR : Abstract Meaning Representation

- Limitations of AMR

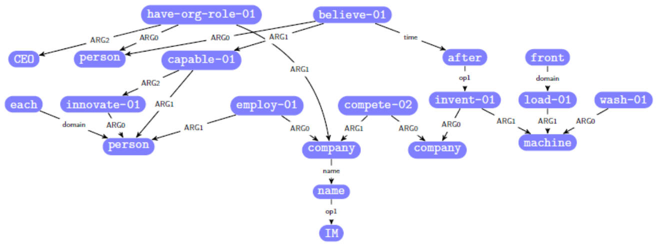
Coreference

- Coreference-based edges indistinguishable from others.
- Linguistically, coreference is very different than e.g. control, namely noncompositional.
- Challenge for composition-based semantic parsers.

Expressive Capacity

- Operators such as quantifiers and modal verbs have scope. This is hard to represent when the MR is not a tree.
- People are stil trying.
- AMR has no model theory. "Man" and "every" are the same type of node label. If "man" refers to a set of men in the world, what's an "every"?

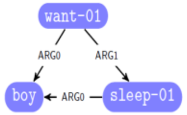
(Even) More Complex AMRs



After its competitor invented the front loading washing machine, the CEO of the IM company believed that each of its employees had the ability for innovation. (AMR2015 #1, simplified)

(a) lexical decomposition (c) coreference
(b) named entities (d) quantification (?)

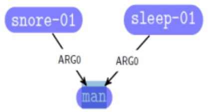
- Standard string format for AMRs is Penman-style:



The boy wants to sleep.

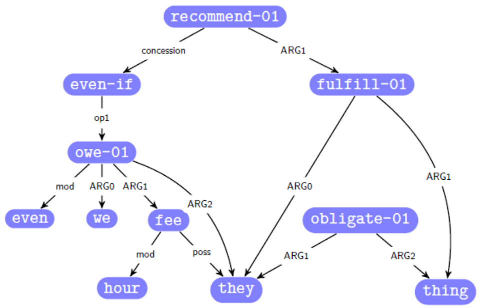
```
(w / want-01
   :ARG0 (b / boy)
   :ARG1 (s / sleep-01 :ARG0 b))
```

- String representation is based on DFS traversal of AMR, which sometimes traverses edges backwards. Represent with "label-of" edges:



The man who sleeps snores.

```
(sn / snore-01
   :ARG0 (m / man
      :ARG0-of (sl / sleep-01)))
```



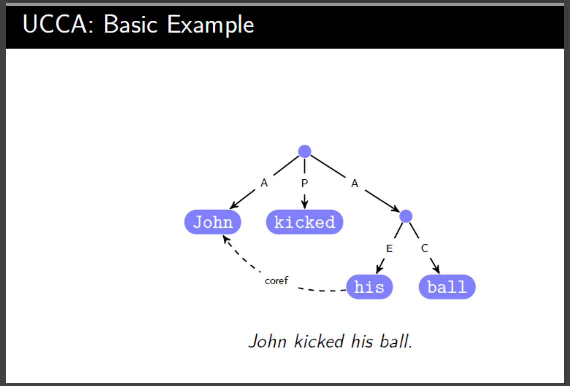
Even if we owed their hourly fees, they still should fulfill their obligations. (AMR2015 #34, simplified)

(a) implicit arguments (c) lexical decomposition
(b) raising-style argument passing (d) coreference

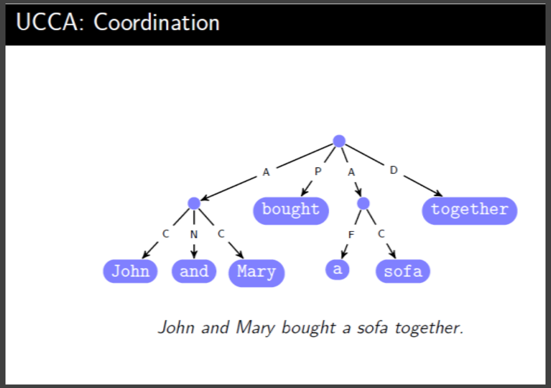
UCCA:Universal Conceptual Cognitive Annotation

- Goals:
  - Capture predicate–argument structure of a sentence, in a way that abstracts over syntactic details.
  - Inspired by typological principles (Basic Linguistic Theory).
  - Make annotation as intuitive as possible, also cross-linguistically.
- Basic ideas:
  - Backbone of UCCA graph is a tree with the tokens as leaves and additional internal nodes, connected by a small set of semantic relations.
  - Additional remote edges represent argument sharing.
  - Multiple annotation layers, e.g. pred-arg structure vs. coreference.
- Annotations of "20000 Miles under the Sea" available in English, French, German; also web texts annotated.
- Hebrew University, since 2013 (Abend & Rappoport, 2013)
- Inter-annotator agreement: around 80 f-score.

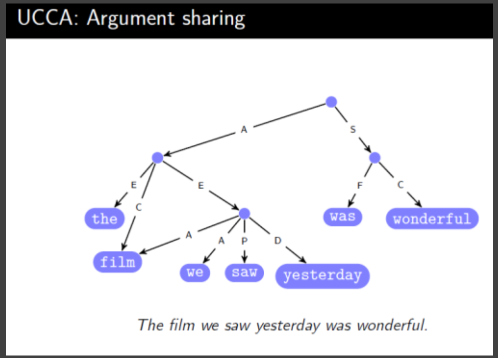
UCCA:Universal Conceptual Cognitive Annotation



- A scene is a description of a single action or state. Sentences can contain multiple scenes. UCCA annotations distinguish between "processes" and "states".
- Scenes can have participants (≈ arguments) and adverbials/times (≈ modifiers).
- Below the clause level, distinguish centers from their elaborators and combine them with connectors.



- Limitations:
  - Less well-studied approach than the others; may change with MRP Shared Task at CoNLL 2019.
  - Does not distinguish different argument roles.
  - Modification of a head causes major changes to graph structure; may be challenging for accurate parsing.



GraphBank Statistics [11]

			DM	PSD	EDS	UCCA	AMR^{-1}
counts	(01)	number of graphs	35,656	35,656	35,656	6,572	56,240
	(01)	number of tokens	802,717	802,717	802,717	138,268	1,000,217
	(02)	average number of tokens	22.51	22.51	22.51	21.03	17,78
	(03)	average nodes per token	0.77	0.64	1.29	1.37	0.65
	(04)	number of edge labels	59	90	10	15	101
treeness	(05)	$\%_g$ trees	2.31	42.26	0.09	34.83	22.24
	(06)	$\%_g$ treewidth one	69.82	43.08	68.99	41.57	50.00
	(07)	average treewidth	1.30	1.61	1.31	1.61	1.56
	(08)	maximal treewidth	3	7	3	4	5
	(09)	average edge density	1.019	1.073	1.015	1.053	1.092
	(10)	$\%_n$ reentrant	27.43	11.41	32.78	4.98	19.89
	(11)	$\%_g$ cyclic	0.00	0.00	0.12	0.00	0.38
	(12)	$\%_g$ not connected	6.57	0.70	1.74	0.00	0.00
	(13)	$\%_g$ multi-rooted	97.47	40.60	99.93	0.00	71.37
	(14)	percentage non-top roots	44.94	4.34	54.85	0.00	20.09
order	(15)	average edge length	2.684	3.320	–	–	–
	(16)	$\%_g$ noncrossing	69.21	64.61	–	–	–
	(17)	$\%_g$ pagenumber two	99.59	98.08	–	–	–

Semantic Parsing Approaches

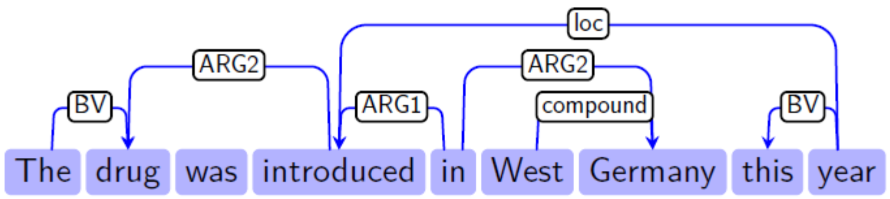
Parsing to Flavor (0) Graphs

Parsing to Flavor (0) Graphs

Flavor	Name	Example	Type of Anchoring
0	bilexical	SDP	nodes are sub-set of surface tokens
1	anchored	EDS	free node–sub-string correspondences
2	unanchored	AMR	no sub-string correspondences annotated

Parsing to Flavor (0) graphs

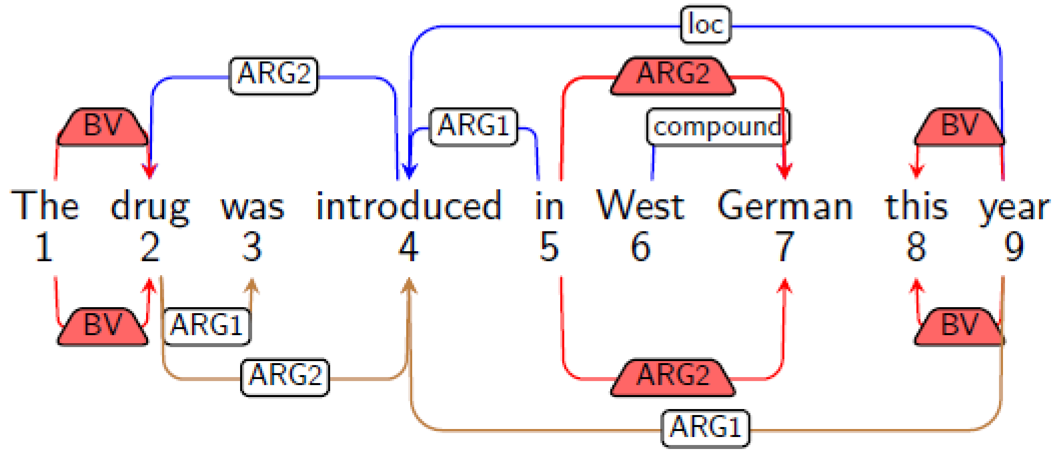
- Nodes = tokens
- The goal is to predict labeled edges



Semantic Parsing is Making Rapid Progress

	DM		PAS		PSD	
	id	ood	id	ood	id	ood
Du et al. (2015) (close)	89.1	81.8	91.3	87.2	75.7	73.3
H. Peng et al. (2017)	89.4	84.5	92.2	88.3	77.6	75.3
+Multitask learning	90.4	85.3	92.7	89.0	78.5	76.4
Dozat & Manning (2018)	93.7	88.9	94.0	90.8	81.0	79.4
Lindemann et al. (2019)	**93.9**	**90.3**	**94.5**	**92.5**	**82.0**	**81.5**
+Multitask learning	**94.1**	**90.5**	**94.7**	**92.8**	**82.1**	**81.6**

Evaluation for Parsing to Flavor (0) Graphs



$$E_{\text{gold}} = \{(1, 2, \text{BV}), (2, 4, \text{ARG1}), \cdots\} \qquad |E_{\text{gold}}| = 7$$

$$E_{\text{system}} = \{(1, 2, \text{BV}), (2, 3, \text{ARG1}), \cdots\} \qquad |E_{\text{system}}| = 6$$

$$E_{\text{match}} = E_{\text{gold}} \cap E_{\text{system}} = \{(1, 2, \text{BV}), (5, 7, \text{ARG1}), \cdots\} \qquad |E_{\text{match}}| = 3$$

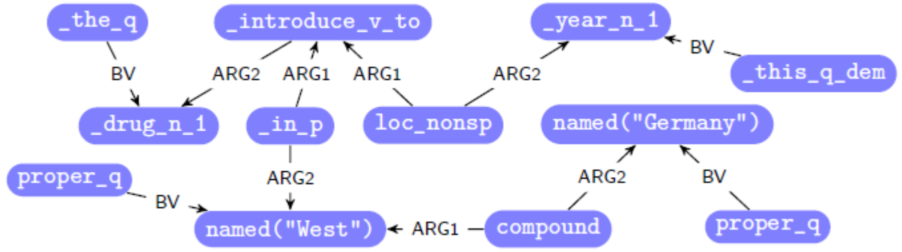
Precision	Recall	F-score
$\frac{\|E_{\text{match}}\|}{\|E_{\text{system}}\|} = 0.43$	$\frac{\|E_{\text{match}}\|}{\|E_{\text{gold}}\|} = 0.5$	$\frac{2 * \|E_{\text{match}}\|}{\|E_{\text{gold}}\| + \|E_{\text{system}}\|} = 0.46$

Parsing to Flavor (1)(2) Graphs

Flavor	Name	Example	Type of Anchoring
0	bilexical	SDP	nodes are sub-set of surface tokens
1	anchored	EDS	free node–sub-string correspondences
2	unanchored	AMR	no sub-string correspondences annotated

Parsing to `Flavor (1)` and `Flavor (2)` graphs

- We need to predict labeled nodes and labeled edges

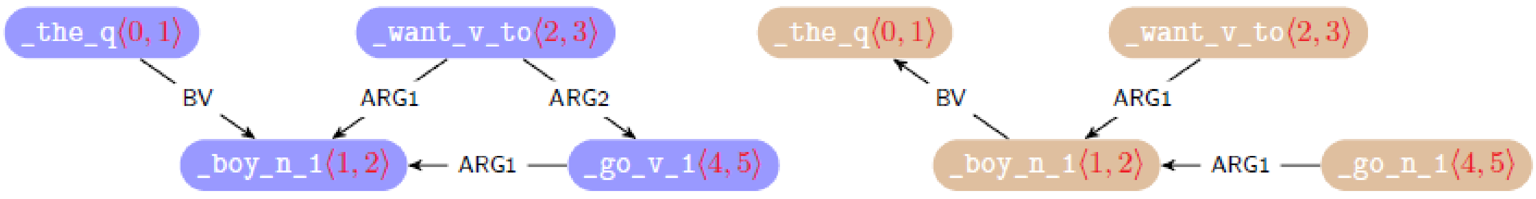


Semantic Parsing is Making Rapid Progress

	EDS		AMR 2015	AMR 2017
	Smatch F	EDM_{na}	Smatch F	Smatch F
Groschwitz et al. (2018)	-	-	70.2	71.0
Lyu & Titov (2018)	-	-	73.7	74.4
S. Zhang et al. (2019)	-	-	-	**76.3**
Buys & Blunsom (2017)	85.5	85.9	60.1	-
Chen, Sun, & Wan (2018)	**90.9**	**90.4**	-	-
Lindemann et al. (2019)	90.1	84.9	**74.3**	75.3
+ Multitask learning	90.4	85.2	74.5	75.3

Evaluation for Parsing to Flavor (1) Graphs[13]

$_0$ *The* $_1$ *boy* $_2$ *wants* $_3$ *to* $_4$ *go* $_5$. $_6$



$V_{\text{gold}} = \{(\langle 0,1\rangle, \texttt{\_the\_q}), \cdots\}$, $|V_{\text{gold}}| = 4$ $\quad V_{\text{sys}} = \{(\langle 0,1\rangle, \texttt{\_the\_q}), \cdots\}$, $|V_{\text{sys}}| = 4$

$E_{\text{gold}} = \{(\langle 0,1\rangle, \text{BV}, \langle 1,2\rangle), \cdots\}$, $|E_{\text{gold}}| = 4$ $\quad E_{\text{sys}} = \{(\langle 1,2\rangle, \text{BV}, \langle 2,1\rangle), \cdots\}$, $|E_{\text{sys}}| = 3$

$$V_{\text{match}} = V_{\text{gold}} \cap V_{\text{sys}} = \{(\langle 1,2\rangle, \texttt{\_boy\_n\_1}), \cdots\} \quad |V_{\text{match}}| = 3$$

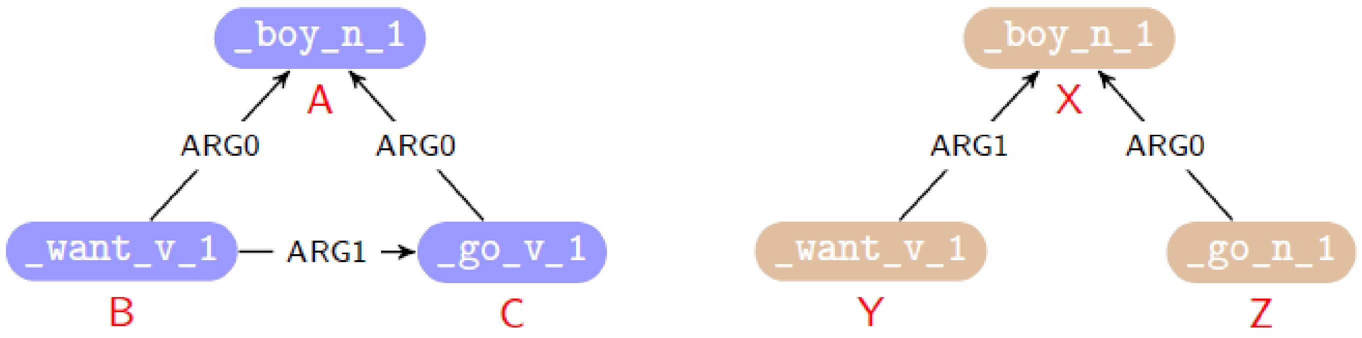
$$E_{\text{match}} = E_{\text{gold}} \cap E_{\text{sys}} = \{(\langle 2,3\rangle, \text{ARG1}, \langle 1,2\rangle), \cdots\} \quad |E_{\text{match}}| = 2$$

EDM_n	EDM_a	EDM_{na}
$\frac{2*\lvert V_{\text{match}}\rvert}{\lvert V_{\text{gold}}\rvert+\lvert V_{\text{sys}}\rvert} = 0.86$	$\frac{2*\lvert E_{\text{match}}\rvert}{\lvert E_{\text{gold}}\rvert+\lvert E_{\text{sys}}\rvert} = 0.57$	$\frac{2*(\lvert V_{\text{match}}\rvert+\lvert E_{\text{match}}\rvert)}{\lvert V_{\text{gold}}\rvert+\lvert V_{\text{sys}}\rvert+\lvert E_{\text{gold}}\rvert+\lvert E_{\text{sys}}\rvert} = 0.67$

Dridan & Oepen (2011)

Evaluation for Parsing to Flavor (2) Graphs[14]

Flavor	Name	Example	Type of Anchoring
1	anchored	EDS	free node–sub-string correspondences
2	unanchored	AMR	no sub-string correspondences annotated



$$\textsc{Smatch}(G_g, G_s) = \max_{a \in \mathcal{A}(G_g, G_s)} \mathrm{EDM}_{\mathsf{na}}(a)$$

$\mathcal{A}(G_g, G_s)$ denotes the set of all plausible alignments between G_g and G_s

Cai & Knight (2013)

Sub-Tasks

- Concept Identification (CI): predicting nodes
- Relation Detection (RD): linking nodes
- Concept-to-word alignment: finding concept-word correspondences

	Alignment	Concept Identification	Relation Detection
Flavor (0)			✔
Flavor (1)		✔	✔
Flavor (2)	✔	✔	✔

Relation Detection

- Structured Prediction Problem[15]
  - Neural Scorers[16]

Maximum Subgraph Parsing

- **Start from** a directed graph $G = (V, E)$ that corresponds to $x = w_0, \ldots w_{n-1}$ and a score function that evaluates the *goodness* of a graph.
- **Search for** a subgraph $G' = (V, E' \subseteq E)$ that maximizes the score function:

$$G' = \arg \max_{G^* = (V, E^* \subseteq E)} \textsc{Score}(G^*)$$

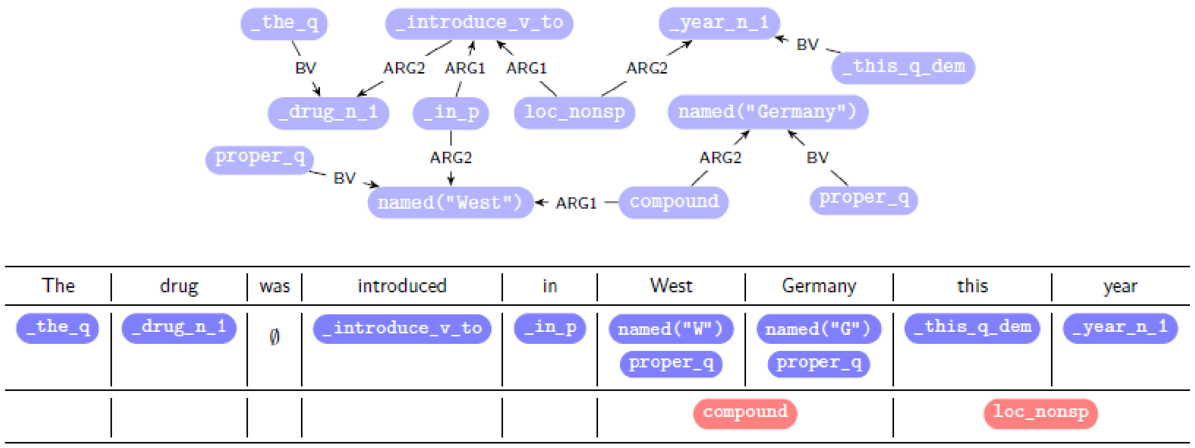
First-order factorization

$$G' = \arg \max_{G^* = (V, E^* \subseteq E)} \sum_{e \in E^*} \textsc{ScorePart}(e)$$

Concept Identification for Flavor (1) Graphs

- Almost Sequence Labeling
  - Some nodes are linked to sub-words.
  - Some nodes are linked to multiple words.
- Solutions
  - Preprocessing: every node is assigned to a single word
  - Chunking: joint segmentation and tagging
    - **B**-*x*: begin of *x*
    - **I**-*x*: inside *x*
  - I Lightweight phrase-structure parsing (UCCA)

Flavor	Name	Example	Type of Anchoring
1	anchored	EDS	free node–sub-string correspondences
2	unanchored	AMR	no sub-string correspondences annotated



Concept Identification for Flavor (1) Graphs

Neural Tagging

Almost Sequence Labeling

- Preprocessing: every node is assigned to a single word
- Chunking: joint segmentation and tagging

Challenge

Like POS tagging but with thousands of labels.

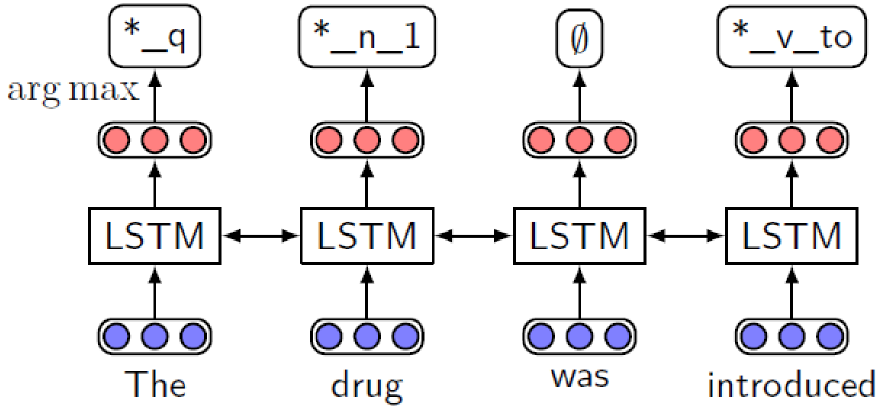
Delexicalization

The	drug	introduced	in	West	Germany	this	year
_the_q	_drug_n_1	_introduce_v_to	_in_p	named("W") compound proper_q	named("G") proper_q	_this_q_dem	_year_n_1 loc_nonsp
*_q	*_n_1	*_v_to	*_p	named compound proper_q	named proper_q	*_q_dem	*_n_1 loc_nonsp

Solution

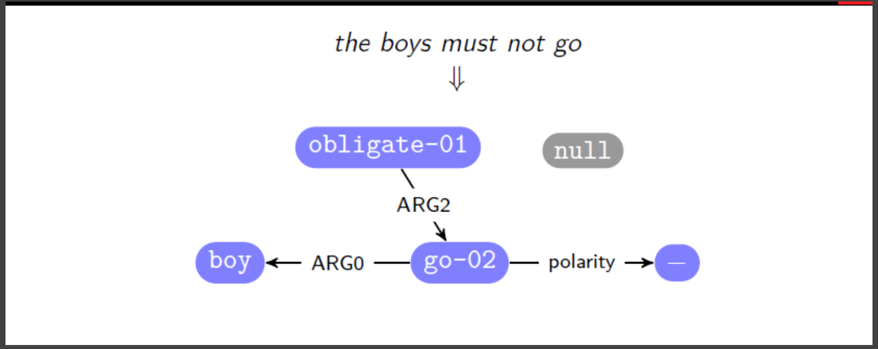
Like POS tagging but with 1000+ labels.

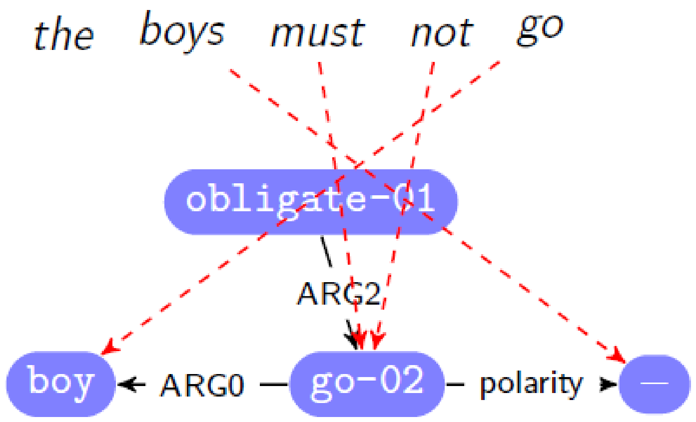
- Pretraining: ELMo, BERT, etc.
- Word encoder: LSTM, Transformer, etc.
- Classification



Concept-to-Word Alignment for Flavor (2) graphs

- Heuristic rules.[17]
- Linearize graphs and reuse word alignment tools, e.g. GIZA++ and BerkeleyAligner, etc. [18]
- Consider all possible alignments. [19]





$$P_{\theta,\phi}(\mathbf{c}, R|\mathbf{w}) = \sum_{\mathbf{a}} P(\mathbf{a}) P_\theta(\mathbf{c}|\mathbf{a}, \mathbf{w}) P_\phi(R|\mathbf{a}, \mathbf{w}, \mathbf{c})$$

1. the concept identification model: $P_\theta(\mathbf{c}|\mathbf{a}, \mathbf{w})$
2. the relation identification model: $P_\phi(R|\mathbf{a}, \mathbf{w}, \mathbf{c})$
3. the alignment model: $P(\mathbf{a})$ $(Q_\psi(\mathbf{a}|\mathbf{c}, R, \mathbf{w}))$

C. Lyu and I. Titov. 2018. AMR Parsing as Graph Prediction with Latent Alignment

Applications of Semantic Graphs

- Using semantic graphs in applications may improve accuracy:
  - Semantic graphs abstract over surface variation.
  - Easier to generalize over graphs than over sentences.
  - ... if semantic parsing is accurate enough.
- Typical applications:
  - machine translation Jones et al. (2012)
  - entity linking / KB population Reddy et al. (2014); Pan et al. (2015)
  - summarization Liu et al. (2015); Hardy & Vlachos (2018)

Conclusions Outlook

- Conclusions

Semantic graph parsing: a success story

- Capture semantic information that is not explicit in syntactic parses.
- Parsers getting increasingly accurate.
- Graphs seem useful in applications.
- Look out for graph parsing papers throughout ACL 2019.

Differences between graphbanks are substantial

- Anchoring of nodes in tokens (flavors 0–2).
- Capture different facets of meaning.
- Different design choices.

Conclusions Outlook

- Outlook

Cross-Framework Semantic Parsing

- Most graph parsers work only for one flavor of graphbank.
- To what degree do they generalize across frameworks?
- Check out CoNLL 2019 Shared Task on Cross-Framework Meaning Representation Parsing (MRP, `http://mrp.nlpl.eu/`).

Facets of Meaning

- Many facets of meaning are not represented by graphbanks.
- What facets are relevant for what applications?
- Push graphbanks so they can be represented, or switch to different meaning representations.
- Check out the ACL 2019 Workshop on Designing Meaning Representations (`https://www.cs.brandeis.edu/~clp/dmr/`).

Reference

- [1] Nerbonne, J. (1994). Book review. Computational linguistics and formal semantics. *Computational Linguistics*, *20*(1), 131 – 136.
- [2] Stephan Oepen, Marco Kuhlmann, Yusuke Miyao, Daniel Zeman, Dan Flickinger, Jan Hajiˇc, Angelina Ivanova, and Yi Zhang. 2014. SemEval 2014 Task 8. Broad-coverage semantic dependency parsing. In Proceedings of the 8th International Workshop on Semantic Evaluation, pages 63–72, Dublin, Ireland.
- [3] Stephan Oepen, Marco Kuhlmann, Yusuke Miyao, Daniel Zeman, Silvie Cinková, Dan Flickinger, Jan Hajiˇc, and Zdeˇnka Urešová. 2015. SemEval 2015 Task 18. Broad-coverage semantic dependency parsing. In Proceedings of the 9th International Workshop on Semantic Evaluation, pages915–926, Denver, CO, USA.
- [4] Hajic, J., Hajicová, E., Panevová, J., Sgall, P., Bojar, O., Cinková, S., Fucíková, E., Mikulová, M., Pajas, P., Popelka, J. and Semecký, J., 2012. Announcing Prague Czech-English Dependency Treebank 2.0. In *LREC* (pp. 3153-3160).
- [5] Yusuke Miyao and Jun'ichi Tsujii. 2008. Feature forest models for probabilistic HPSG parsing. Computational Linguistics, 34(1):35–80.
- [6] Julia Hockenmaier and Mark Steedman. 2007. CCGbank. A corpus of CCG derivations and dependency structures extracted from the Penn Treebank. Computational Linguistics, 33:355–396.
- [7] Computational Linguistics, 33:355–396. Angelina Ivanova, Stephan Oepen, Lilja Øvrelid, and Dan Flickinger. 2012. Who did what to whom? A contrastive study of syntacto-semantic dependencies. In Proceedings of the 6th Linguistic Annotation Workshop, pages 2–11, Jeju, Republic of Korea.
- [8] Abend, O. and Rappoport, A., 2013, August. Universal conceptual cognitive annotation (ucca). In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 228-238).
- [9] Oepen, S. and Lønning, J.T., 2006, May. Discriminant-Based MRS Banking. In *LREC* (pp. 1250-1255).
- [10] Copestake, A., 2009, March. Invited Talk: Slacker semantics: Why superficiality, dependency and avoidance of commitment can be the right way to go. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)* (pp. 1-9).

Reference

- [11] Kuhlmann, M., & Oepen, S. (2016). Towards a catalogue of linguistic graph banks. *Computational Linguistics*, *42*(4), 819 – 827.
- [12] Stephan Oepen, Marco Kuhlmann, Yusuke Miyao, Daniel Zeman, Dan Flickinger, Jan Hajiˇc, Angelina Ivanova, and Yi Zhang. 2014. SemEval 2014 Task 8. Broad-coverage semantic dependency parsing. In Proceedings of the 8th International Workshop on Semantic Evaluation, pages 63–72, Dublin, Ireland.
- [13] Dridan, R., & Oepen, S. (2011). Parser evaluation using elementary dependency matching. In *Proceedings of the 12th International Conference on Parsing Technologies* (pp. 225 – 230). Dublin, Ireland.
- [14] Cai, S., & Knight, K. (2013). Smatch. An evaluation metric for semantic feature structures. In *Proceedings of the 51th Meeting of the Association for Computational Linguistics* (pp. 748 – 752). Sofia, Bulgaria.
- [15] M. Kuhlmann and P. Jonsson. 2015. Parsing to Noncrossing Dependency Graphs.
- [16] T. Dozat and C. Manning. 2018. Simpler but More Accurate Semantic Dependency Parsing.
- [17] Flanigan, J., Thomson, S., Carbonell, J., Dyer, C., & Smith, N. A. (2014). A discriminative graph-based parser for the Abstract Meaning Representation. In *Proceedings of the 52nd Meeting of the Association for Computational Linguistics* (pp. 1426 – 1436). Baltimore, MD, USA.
- [18] Lyu, C., & Titov, I. (2018). AMR parsing as graph prediction with latent alignment. In *Proceedings of the 56th Meeting of the Association for Computational Linguistics* (p. fromto397407). Melbourne, Australia.