Sentence Meaning Representations Across Languages: What Can We Learn from Existing Frameworks?

Zdeněk Žabokrtský
Charles University
Faculty of Mathematics and Physics
Institute of Formal and Applied
Linguistics (ÚFAL)
zabokrtsky@ufal.mff.cuni.cz

Daniel Zeman
Charles University
Faculty of Mathematics and Physics
Institute of Formal and Applied
Linguistics (ÚFAL)
zeman@ufal.mff.cuni.cz

Magda Ševčíková
Charles University
Faculty of Mathematics and Physics
Institute of Formal and Applied
Linguistics (ÚFAL)
sevcikova@ufal.mff.cuni.cz

This article gives an overview of how sentence meaning is represented in eleven deep-syntactic frameworks, ranging from those based on linguistic theories elaborated for decades to rather lightweight NLP-motivated approaches. We outline the most important characteristics of each framework and then discuss how particular language phenomena are treated across those frameworks, while trying to shed light on commonalities as well as differences.

1. Introduction

1.1 Motivation

Distinguishing between semantic (deep) and formal (surface) phenomena in description of natural languages is to be traced back to the notion of language sign as a pairing of form and function (meaning), which has been accepted as a core concept in modern

linguistics since de Saussure (1916, 1978). The dual perspective, most often exemplified on words, applies also to morphemes as subword units, and to more complex structures, such as sentences and texts. At the level of sentence, which is the focus of the present article, the form-meaning opposition has been elaborated to leveled approaches, whose details vary fundamentally in frameworks with different theoretical backgrounds and/or application focus.

The aim of our survey is to compare available approaches to structured sentence meaning representations (deep-syntactic representations),[1] in order to demonstrate that there are basic principles shared by most (if not all) of them, on the one hand, and specific decisions, on the other. The shared principles, being considered the core elements of deep-syntactic representations, will be reformulated into a handful of humble suggestions for a discussion on a unifying approach to sentence meaning. This perspective justifies the inclusion of the Universal Dependencies project, though currently not containing a proper sentence meaning annotation (Schuster and Manning 2016), since the project sets trends in carrying out a unified annotation at the surface-syntactic level.

1.2 Existing Surveys

These days, one can find comprehensive handbooks collecting a number of descriptions of various linguistic issues and language data resources. The recent *Handbook of Linguistic Annotation* (Ide and Pustejovsky 2017) provides an overview of annotation approaches applied in several tens of data resources capturing a wide range of language phenomena. Design decisions on the annotation schemes, evolution, and possible future developments of the particular resources are outlined, usually by the authors of the resources themselves.

The volumes by Ágel et al. (2003, 2006) have a narrower (and more theoretical) focus, dealing with dependency and valency issues. They also include review chapters on individual theories dealing with these concepts.

We can also list many published attempts at comparing various features of deep-syntactic frameworks; however, to our knowledge each of them handles only a very limited number of existing frameworks and/or narrow scope of features compared. Hajičová and Kučerová (2002) compare three frameworks, namely, PropBank (Kingsbury and Palmer 2002), the LCS Database containing Lexical Conceptual Structures introduced by Dorr (1997), and the (pilot) annotation of the Prague Dependency Treebank (Hajič 1998); a possible mapping among these three representations is sketched, with a focus on mapping semantic roles. A mapping from PropBank argument labels to 20 thematic roles used in VerbNet (Kipper, Dang, and Palmer 2000) is designed by Rambow et al. (2003).

Ellsworth et al. (2004) compare PropBank, SALSA (Erk and Pado 2004), and FrameNet (Johnson et al. 2002), with a focus on several selected phenomena such as metaphor, support constructions, words with multiple meaning aspects, phrases realizing more than one semantic role, and non-local semantic roles.

Žabokrtský (2005) points out several parallels between Meaning-Text theory (Žolkovskij and Mel'čuk 1965) and Functional Generative Description (Sgall 1967) in general, and more specifically between the deep-syntactic level of the former one and tectogrammatical level of the latter one.

---

1 There are also entirely different approaches to representing sentence meaning, such as vector space models; they are outside of the focus of our study.

Ivanova et al. (2012) contrast seven annotation schemes for syntactic-semantic dependencies: CoNLL Syntactic Dependencies (Nivre et al. 2007), CoNLL Prop-Bank Semantics (Surdeanu et al. 2008), Stanford Basic and Collapsed Dependencies (De Marneffe and Manning 2008), Enju Predicate–Argument Structures (Yakushiji et al. 2005), as well as Syntactic Derivation Trees and Minimal Recursion Semantics from the DEPLH-IN project,[2] and show a few similarities across the frameworks. Oepen et al. (2015) compare three approaches (DELPH-IN semantic annotation, Enju Predicate–Argument Structures, and the deep-syntactic annotation of the Prague Czech-English Dependency Treebank) in relation to the task of broad-coverage semantic dependency parsing in SemEval 2015. Kuhlmann and Oepen (2016) follow up on the SemEval paper and describe graph properties of the three frameworks from the SemEval task, plus CCG Dependencies and Abstract Meaning Representation.

Zhu, Li, and Chiticariu (2019) take two approaches, namely, the semantic role labeling approach of the PropBank project and Abstract Meaning Representation, as a point of departure to propose and discuss which issues are to be covered by the universal semantic representation (with the focus on temporal features and modality).

To the best of our knowledge, so far the most comprehensive comparison of existing semantic representation accounts (Abend and Rappoport 2017) puts together a list of central semantic phenomena (predicates, argument structure, semantic roles, coreference, anaphora, temporal and spatial relations, discourse relations, logical structure, etc.). The authors' conclusion that "the main distinguishing factors between schemes are their relation to syntax, their degree of universality, and the expertise and training they require from annotators" opens space for further discussion.

What we try to offer in this article is to intensify insights into the core principles of semantic representations, by surveying eleven approaches and comparing their accounts of the most relevant issues.

In our comparison, we focus on how various language phenomena are handled by individual approaches, rather than on what is their general (attested or assumed) utility in end-user NLP applications. Complex software ecosystems with pipelines of NLP components (in which sentence parsers always play the key role) have been implemented for most of the selected approaches. Some of these seem to be more or less abandoned now, such as TectoMT for FGD (Žabokrtský, Ptáček, and Pajas 2008), whereas others are still being actively developed, such as ETAP-3 for MTT (Boguslavsky 2017), or yet another MTT parser described by Ballesteros et al. (2014). There are also NLP pipelines that were focused on surface processing so far, and deeper components are planned to be added only recently—for instance, as in the case of UDPipe for UD (Straka, Hajič, and Straková 2016; Straka and Straková 2017). In our opinion, comparing past performance of such software systems created in a time span longer than two decades would not be fair, and predicting their future usability would not be wise, as it might be conditioned by many extra-scientific factors, especially by the ability of NLP developers to integrate recent Deep Learning advances into their systems. Thus we give only occasional references to NLP applications in this text. Performance comparisons for some narrower semantically oriented NLP tasks can be found, for example, in the NLP shared task literature, such as in Oepen et al. (2015) for the task of semantic dependency parsing. In some NLP frameworks, deep syntactic structures are built in a more or less deterministic way by post-processing outputs of surface-syntactic parsers, and thus shared tasks on surface-syntactic parsing are relevant, too, such as the

---

2 http://www.delph-in.net.

CoNLL-2018 shared task on Multilingual Parsing from Raw Text to Universal Dependencies (Zeman et al. 2018).

1.3 Structure of the Article

The article is organized as follows. Section 2 explains how we selected frameworks for our survey, and lists linguistic notions most relevant for our study, as well as terminological choices adopted in our text. Section 3 outlines the most important characteristics of each framework, and Section 4 brings an orthogonal perspective and discusses how particular language phenomena are treated across frameworks. Section 5 concludes and tries to summarize possible inspirations for the future development of Universal Dependencies.

2. Focus of the Study

2.1 Criteria for Selection of Frameworks to Compare

The present survey is focused on approaches that fall into the field of computational linguistics and aim at designing and implementing formalized representations of sentence meaning.

It is clear from the very beginning that selection is inescapable, as it is beyond the capacity of our team to review hundreds or thousands of research threads relevant for this or that aspect of sentence meaning published in the last five or six decades. However, we should clarify what selection criteria we applied in our selection.

First of all, we limit ourselves to meaning representations whose backbone structure can be described as a graph over words (possibly with added non-lexical nodes) corresponding to entities, processes, properties, or circumstances, with edges representing meaningful relations among them. However, we do not include approaches that represent only surface sentence structure, in the sense that they handle only the original sequence of word forms and do not make any abstraction above overt morphological, lexical, or syntactic means. Thus, for example, dozens of surface-syntactic treebanks (be they based on the dependency or constituency paradigm) are excluded. At the same time, we do not include primarily logical representations that are too distant from sentence structures; this leaves out some prominent frameworks such as the Groningen Meaning Bank (Bos et al. 2017), where the central unit, the discourse representation structure, is a recursive structure mappable to first-order logic. All semantic representations developed primarily for modalities other than natural language, such as scene graphs introduced by Johnson et al. (2015), are excluded, too (although eventually, one could find a potential overlap with NLP, like the task of generating image captions in this particular case).

Second, we select only frameworks capable of analyzing whole authentic sentences of natural languages. We do not review approaches whose aim is only lexicographical, although, for example, valency lexicons could also be understood as collections of elementary dependency trees.

Third, we explore only approaches that seem mature enough to attract supercritical mass of research effort, and are being elaborated for a longer time, are tested on authentic data and possibly also used for natural language applications.

On the other hand, it was not decisive for us whether the framework proclaims a multilingual or even language-independent (universal) ambition and whether or not it declares the ability to represent synonymous sentences with identical representations.

Table 1
Overview of the frameworks described in Sections 3.1 to 3.11.

Framework	Associated corpus	Associated lexical resource	Used in NLP apps	Languages
Paninian framework (400 BC)	HDTB, UDTB		MT	hi, ur, bn, te
Meaning-text theory (MTT; Žolkovskij and Mel'čuk 1965)	SynTagRus, AnCora-UPF	ECD	MT	ru, en, es, fr
Functional Generative Description (FGD; Sgall 1967)	PDT, PCEDT	PDT-VALLEX	MT	cs, en
PropBank (Kingsbury and Palmer 2002)	PropBank + NomBank + PDTB	PropBank lex.	many	en, ar, zh, fi, hi, ur, fa, pt, tr, de, fr
FrameNet-based approaches such as SALSA (Erk and Pado 2004)	e.g. TIGER Treebank (for SALSA)	FrameNet		en, de, fr, ko
Enju (Yakushiji et al. 2005)	Enju Treebank		IE	en, zh
DELPH-IN (Oepen and Lønning 2006)	DeepBank	ERG	many	en, de, es, ja
Sequoia (Candito et al. 2014)	Sequoia			fr
Abstract Meaning Representation (AMR; Banarescu et al. 2013)	AMR Bank	PropBank lex.	many	en, zh, pt, ko, vi, es, fr, de
Universal Conceptual Cognitive Annotation (UCCA; Abend and Rappoport 2013)	English Wiki, parallel fiction, etc.			en, de, fr
Enhanced Universal Dependencies (Schuster and Manning 2016)	Universal Dependencies		relation extraction	ar, bg, cs, en, et, fi, it, lt, lv, nl, pl ru, sk, sv, ta, uk

We ended up with a selection of frameworks, listed in Table 1 (and then described in detail in Section 3) in roughly chronological order of their introduction. Associated corpora and major lexicographical resources are listed, too.

We are aware of quite a few other approaches virtually located on the fuzzy border-line imposed by our selection criteria and implementing just one or two features that are usually considered meaning-related, on top of basically surface-syntactic frameworks. Without the slightest ambition at completeness, we illustrate some such approaches very briefly in a common subsection at the end of Section 3. That said, it might be surprising for the reader that the enhanced version of Universal Dependencies (UD)

receives more attention in our text (Section 3.11), as UD are also primarily focused on surface sentence structures, and attempts at including more semantic features into UD have been relatively modest so far. The reason is rather pragmatic: UD is truly unique among syntactic frameworks in popularity gathered in recent years, and one could expect substantial efforts invested in more semantic extensions soon. Technically, enhanced UD is ready for such a drift already now, as it offers technical means for (for example) adding reconstructed nodes and other than syntactic relations such as coreference.

2.2 Basic Notions and Terminological Choices

As each of the selected approaches uses its own terminology, denoting even very closely delimited notions by different terms (cf. semantic role vs. functor) and, at the same time, using similar terms for different phenomena (e.g., adjunct), we make terminological choices to which we stick in the remainder of the text before we start exploring the diversity of "design decisions" made in individual frameworks.

2.2.1 Graph-Structure Notions. Whenever possible, we use theory-neutral terms from graph theory:

- **Node** – typically capturing a word/lemma/lexeme occurring in the particular sentence. Nevertheless, there are exceptions. Depending on framework and language, selected words may be broken up to smaller meaningful units, and selected groups of words (even discontinuous) may be treated as one meaningful unit. Moreover, there may be **empty nodes**[3] representing a hypothetical word that is part of the meaning the speaker wished to convey but was omitted from the sentence for various reasons. Despite being called "empty," the node may actually be associated with a lexeme if we know what the hypothetical word looks like. That is the case with **copied nodes**—if two or more deep nodes refer to the same surface word, we can regard one of them as the "main" node and the others as copies. The copies are special cases of empty nodes because they represent a hypothetical word that would be identical to some real surface word. Yet another type of node, found in a few frameworks in our survey, are **nonterminal nodes.** Although they do not directly represent a word (not even a hypothetical word), the graph structure may still link them to one or more words, via **terminal nodes** that directly represent them.

- **Edge** – binary relation between two nodes, typically capturing some kind of dependency, coordination, coreference, and so forth, manifested in the sentence. Edges may be **directed**—that is, an edge going from node A to node B is different from an edge going from B to A. In some contexts we will refer to a directed edge as a **dependency;** we will then use the term **governor** or **parent** for the source node and **dependent** or **child** for the target node. Nodes that have no outgoing edges (no children) are called **leaves.**

---

3 Also called *null / fictitious / reconstructed / restored / zero / phantom nodes / traces*, etc., by various approaches.

- Both nodes and edges may have **labels** that specify their type and other attributes.

- **Graph** – the pair (V, E) where V is a set of nodes and E is a set of edges connecting the nodes from V. A graph is **directed** if at least one type of edges in the graph are directed (note that an undirected edge can be represented by two directed edges going in opposite directions).

- **Directed path** – sequence of one or more directed edges $(v_1 \rightarrow v_2, \dots, v_{n-1} \rightarrow v_n)$ where each edge starts in the node in which the previous edge ends. A directed path is a **cycle** if it starts and ends in the same node ($v_1 = v_n$).

- **Rooted tree** – a graph where one node is designated as the "root" and there is exactly one directed path from the root to each non-root node.

- **Directed acyclic graph** (DAG) – directed graph that does not contain cycles. Every tree is acyclic, but not every acyclic graph is a tree. A DAG may be **rooted,** meaning that all non-root nodes are reachable by at least one directed path from a single root node.

- **Undirected path** – a generalized path where direction of edges is ignored, for example ($w \rightarrow x \leftarrow y \rightarrow z$). A graph is **connected** if for any two nodes (x, y) there is an undirected path between x and y. We will call the graph **undirected tree** if for any two nodes (x, y) there is exactly one undirected path that connects x with y. Note that an undirected tree can be converted to a rooted tree if one node is picked as the root and if the set of edge labels is extended to encode the direction of the original edge.

2.2.2 Node Order. The notion of node order is important in some surveyed approaches. In general, there are two basic definitions related to ordering:

- **Partial order** $\leq$ on a set is any binary relation that is reflexive ($n \leq n$), antisymmetric ($n_1 \leq n_2$ and $n_2 \leq n_1$ implies $n_1 = n_2$), and transitive (if $n_1 \leq n_2$ and $n_2 \leq n_3$ then $n_1 \leq n_3$).

- **Total order**, also called linear order or full order, adds the connexity condition ($n_1 \leq n_2$ or $n_2 \leq n_1$ for every pair of nodes n_1, n_2).

The most usual approach is that there is a partial or total node order used in a particular deep-syntactic representation that is more or less directly related to the surface word order. The order becomes partial if, for instance, there are empty nodes without a corresponding surface token, or because of other surface-depth asymmetries indicated in Section 4.2.4. A less common approach is that—by design—there is no node order declared at all, which allows us to treat wider ranges of paraphrases as synonymous (in the sense of having identical deep-syntactic representations). Another rare approach is that node order is reserved for representing some other linguistic notion, not necessarily related only to the surface word order. A more detailed discussion is provided in Section 4.2.4.

When we present a 2D visualization of a sample sentence representation in the following sections, the horizontal axis typically corresponds to the word order in the original sentence. However, this does not imply that a (partial or total) order of nodes is considered a proper theoretical component of that representation; we simply have to

draw the charts somehow, and we prefer to present the words in their original order where possible.

2.2.3 Deletions and Coreference. When analyzing sentence meaning representations in the surveyed approaches with regard to the surface shape of the represented sentences, nodes of the graph structure correspond mainly to items that are present in the surface sentence. However, some of the nodes do not have a surface counterpart, representing an item that is omitted (deleted) in the surface for different either structural or contextual reasons. Here are the main terms that we use for describing deletions.

- **Dropped pronoun** – in "pro-drop languages," the subject may be unexpressed; unlike in English, not even a pronoun is required. Typically, features of the missing pronoun are reflected in the morphological form of the verb, but there is no separate word to represent the subject; cf. the Spanish example *cantamos* "[we] sing." Dropped pronouns may be reconstructed in the semantic representation and represented by an empty node. This option is not limited to subjects: Any argument that is licensed by the verb and deleted on the surface may be reconstructed.

- **Ellipsis in coordination** – there are various situations where coordinate constituents allow or require semantic interpretation that involves ellipsis and thus leads to empty nodes (Hajič et al. 2015; Droganova and Zeman 2017).

    - **Coordinate dependents** may represent multiple modifications of the same parent, or multiple copies of the parent each with its own modification. For instance, *young and beautiful girl* probably refers to one girl that is both young and beautiful. In contrast, *red and white wine* is most likely to be interpreted as a shortcut for *red wine and white wine*.

    - The situation is more complicated if the parent is a verb with multiple arguments. Just like with the red and white wine, multiple instances of the verb may be understood where only one instance is present on the surface: *James flies to Paris and Martha to Prague.* This construction is called **gapping.**[4] It is quite clear that there is a verb missing, its meaning is the same as that of the visible verb *(flies)* but each instance refers to a different event, so we cannot attach all four dependents *James, Paris, Martha, Prague* to a single node representing *flies*.

    - Finally, we can also observe coordinate constituents with one or more **shared dependents,** as in *Harry buys and sells cars.* This sentence can be understood as a shortcut for *Harry buys cars and Harry sells cars*; the noun *Harry* refers to the same person in both clauses, and with some level of simplification, we can say the same about the cars (although one could not say whether it is the same set of cars in both cases).

---

4 The literature distinguishes other related types of ellipsis such as stripping, pseudo-gapping, or VP ellipsis. For simplicity, we do not discuss them further and just note that similar mechanisms can be used to capture them in a sentence meaning representation.

In a close relation to deletions, we use the term coreference, distinguishing between grammatical and textual coreference.

Grammatical coreference refers to cases when a rule of the grammar specifies that two expected constituents are the same or have the same referent. It is not unusual that the grammar also requires that the constituent occurs only once on the surface, that is, the second occurrence is elided. The control verb construction is an example: in *I want to go*, the pronoun *I* is the subject of *want* but it also represents the missing subject of *go*. Other examples are relative clauses or some reflexive constructions (Zikánová et al. 2015, § 3.3).

If two or more expressions refer to the same entity and this fact cannot be deduced from grammatical rules alone, the term **textual coreference** is used. A typical example involves a noun and a later occurring pronoun; this type of coreference is called **anaphora** (or **cataphora**, if the pronoun linearly precedes the noun), the pronoun is an anaphoric expression or **anaphor**, and the noun is its **antecedent**.[5] Coreference can also hold between two nouns or other expressions. Furthermore, a pronoun that participates in coreference can be elided, in particular in pro-drop languages. Textual coreference often crosses sentence boundaries, unlike grammatical coreference.

Another term, **bridging relation**, applies to a semantic, anaphoric relation between two expressions that is not a full identity but rather a weaker association or a set-subset relation; therefore, the relation is not coreference. Example: *I met two people yesterday. The woman told me a story* (Clark 1975).

2.2.4 Dependency and Valency. The combinatorial potential as a general linguistic phenomenon is called **valency** in most frameworks in the European tradition that refer back to Tesnière (1959) (cf. Ágel et al. 2003, 2006), in parallel to the chemical property of an atom of a certain element to combine with a specific number of bindings with other atoms. In other contexts, often the term **argument structure** is used.[6] We adhere to the term valency.

Combinatorial potentials of particular language units, that is, the range of syntactic elements either required or specifically permitted by the unit, are referred to by different terms in linguistic literature. Compare the terms valency frame, case frame, thematic (theta) grid, and subcategorization frame (if only the surface morphosyntactic features of individual slots are considered). Less often, terms such as government patterns (Mel'čuk and Žolkovskij 1984), stereotypical syntagmatic patterns (Pustejovsky, Hanks, and Rumshisky 2004), or complex sentence pattern (Daneš 1994) are found. In the following survey, we stick to the very first option, **valency frame**.

Out of the terms for individual items in valency frames (frame elements, frame slots, case slots, etc.) we choose the term **frame slot** for the survey.

The roles[7] played by an argument that fills a governor's slot in a particular event or situation are labeled, inter alia, with the terms semantic roles, thematic roles (theta roles), (deep) cases (Fillmore 1968), functors (Sgall, Hajičová, and Panevová 1986), or non-specifically, deep-syntactic relations (Kahane 2003). The term **semantic roles** is preferred throughout our survey. Nevertheless, the above defined difference between

---

5 Zikánová et al. (2015) and others note that anaphora is in fact not subsumed by coreference, as there are also examples of anaphora where the anaphor is not coreferential with the antecedent.

6 One could find many other terms in linguistic theories that were not lucky enough to become popular, e.g., "intention" introduced by Pauliny (1943).

7 Such primarily semantically motivated roles are distinguished from grammatical relations such as subject and (direct and indirect) objects, which seem much more clearly defined.

frame slots and semantic roles is not followed consistently in papers on argument structure and valency; for instance, one can find frame slots referred to as semantic roles and conversely.

Typically, inventories of semantic roles are partitioned into two parts, in slightly different ways though.[8] A specific set of roles is often defined for core dependents, which are highly specific for and closely tied to the meaning of a certain lexical unit. A different set of roles applies to the other dependents, which are much less specific (and thus are often not worth listing in valency frames). The core dependents are called arguments, actants (in the tradition of Tesnière), or inner participants, while the other dependents are called adjuncts, circumstants (Tesnière's circonstants), or free modifiers. We adhere to the terminological distinction **arguments** vs. **adjuncts**.

A predicate with its arguments is referred to by the term **predicate–argument structure** in the present article.

A predicate–argument structure together with adjuncts make up a clause whose meaning is denoted by the term **proposition**.

3. Deep-Syntactic Frameworks under Survey

3.1 Paninian Framework

The description of Sanskrit grammar by Panini (probably 4th century BC; Kiparsky 1982; Bharati, Chaitanya, and Sangal 2006) has become a popular base for treebanking and NLP in India. The deep-syntactic representation of the Paninian framework has been applied to Indo-Aryan languages (including Sanskrit), but also to Dravidian languages and English.

The Paninian syntax is the basis for annotation in some treebanks of Indian languages (Hindi, Urdu, Bengali, Telugu, and others; Husain et al. 2010). This framework defines so-called karaka relations, which lie half-way between syntax and semantics. On the one hand, they do not distinguish subject and object in the usual sense, as they rather operate along the actor-patient axis and abstract from different realizations of the roles in active and passive clauses (i.e., from their syntactic diatheses). On the other hand, karaka relations are very coarse-grained and do not directly correspond to semantic roles. For example, the relation k1 *karta* is the most independent participant in the event, and it often corresponds to the actor, but there are clauses in which the karta is not what other theories would want to describe as actor. Thus in (1) the karta is *the boy*; but it is *the key* in (2) and *the lock* in (3).

(1)   *The boy opened the lock.*

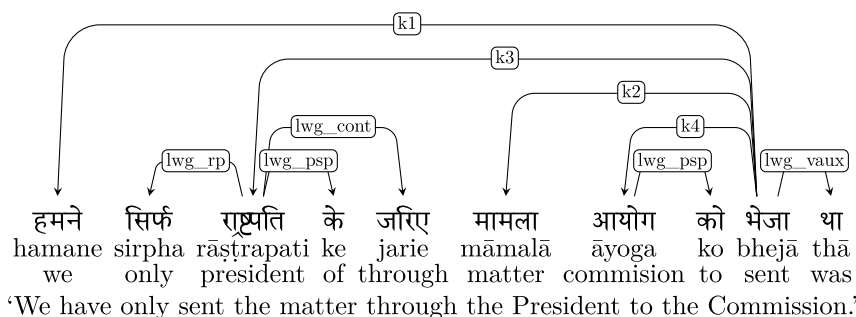(2)   *The key opened the lock.*

(3)   *The lock opened.*

Table 2 briefly introduces the six main karaka relations and their labels in Paninian treebanks. Figure 1 shows an example sentence from the Hindi treebank with four karakas.

---

8 As Przepiórkowski (2016) says: "Probably all modern linguistic theories assume some form of the Argument-Adjunct dichotomy, which may be traced back to Lucien Tesnière's 1959 distinction between actants and circumstants."
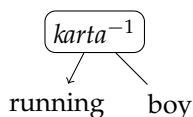
Table 2
The six karaka relations of the Paninian syntax. Note that there is no karaka labeled `k6`, at least not in modern annotation schemes referring to the Paninian grammar, such as Begum et al. (2008). Relation number 6 denotes possession but it does not have the karaka status and is labeled `r6`.

`k1`	karta	doer / agent / subject
`k2`	karma	patient / object
`k3`	karana	instrument
`k4`	sampradaana	recipient / beneficiary
`k5`	apaadaana	source
`k7`	adhikarana	location in space or time

हमने सिर्फ़ राष्ट्रपति के जरिए मामला आयोग को भेजा था
hamane sirpha rāṣṭrapati ke jarie māmalā āyoga ko bhejā thā
we only president of through matter commision to sent was
'We have only sent the matter through the President to the Commission.'

Figure 1
A Hindi sentence with the first four karaka relations. The relations prefixed `lwg` are chunk-internal, i.e., they are not part of the main structure.

- Structure: rooted tree.[9] The nodes can be ordered following the surface word order; the order is partial if the tree contains an empty node.

- Nodes generally correspond to nominal or verbal chunks. Function words are "second-class citizens." They are just chunk members, although one could see additional relations between the head of the chunk and its other members. Empty nodes may be used to represent elided predicates.

- Edges: karaka relations (see Table 2) are what makes the framework "deep":
  - The karma-nominal stays karma even when the sentence is transformed to passive.
  - Attributively used participle: The relation is still karta (rather than the type used for adjectival modifiers) but with $^{-1}$ marking the reversed direction:


  - Same for verbal nouns / infinitives.

---

9 For readers' comfort, selected structural properties of individual frameworks are listed in a summarized form in Table 4 in Section 4.

- Case (vibhakti) is either a morphological case, or a postposition, or a combination of both.

3.2 Meaning–Text Theory

In the Meaning-Text theory (MTT), which is rooted in pioneering efforts in machine translation in the early 1960s (Žolkovskij 1964; Žolkovskij and Mel'čuk 1965, 1967), a descriptive framework has been elaborated that decomposes the relation between form and meaning into seven representations. The surface-phonological representation (text), deep-phonological, surface-morphological, deep-morphological, surface-syntactic, deep-syntactic, and semantic representation (meaning) are distinguished. The deep-syntactic representation is in focus here, though being "certainly the least defined level of representation of MTT" according to Kahane (2003, page 556).

- Structure: rooted directed acyclic graph (but cycles will arise when coreference links are counted as edges; see Figure 2). The nodes can be ordered following the surface word order.

- Nodes correspond to content words; function words are not part of the deep-syntactic representation; copied nodes are used to represent controlled subjects. Nodes are labeled with a deep lexeme, a set of grammeme attributes, a coreference attribute, and attributes capturing the deep-syntactic communicative structure.

  - The deep lexeme corresponds to the basic, dictionary form of a word, to Lexical Functions (see below), or to fictitious lexemes (for representing peripheral phenomena). If a string in the surface sentence forms a multiword expression (phraseme), the whole string is represented with a single node in the deep-syntactic representation.

  - Grammeme attributes capture grammatical categories that are not imposed by government and agreement and are relevant for the meaning of the sentence, for example, grammatical number and definiteness.

  - Attributes of the deep-syntactic communicative structure capture the differences between a question, affirmation, irony, doubt, and so on.

- Edges represent dependency relations between nodes; they are labeled with a small set of semantic relations, distinguishing between arguments and other items.

  - Arguments are numbered starting with *I* for the "most salient" argument, followed by *II* for the second most salient argument, and so forth.

  - Another three roles are defined for other items, namely, ATTR for attributes and other modifiers, COORD for relations between coordinated items, and APPEND for parentheses, interjections,
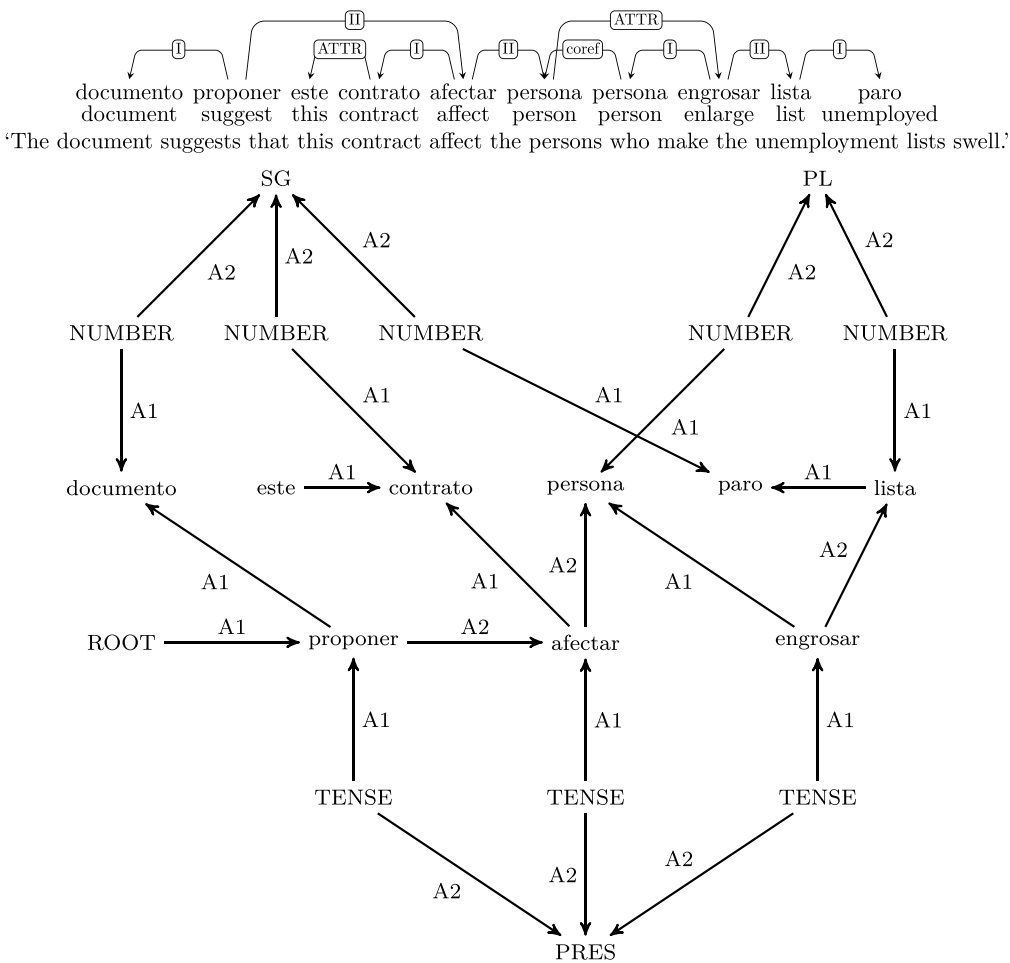
Figure 2
Deep-syntactic representation and semantic representation of the sentence *El documento propone que este contrato afecte a las personas que engrosen las listas del paro.* 'The document suggests that this contract affect the persons who make the unemployment lists swell.' in the AnCora-UPF treebank (adopted from Mille, Burga, and Wanner 2013). In the deep-syntactic graph (above the sentence), nodes correspond to content words and are labeled by their lemmas. Note the double occurrence of the node corresponding to *personas* (labeled *persona*, linked by coreference). In the semantic graph (in the bottom), these two nodes are merged. All nodes are treated as predicates and/or arguments, and there are additional "predicates" representing grammatical meaning such as number and tense.

and other items that are attached to a sentence without a proper dependency relation.

Being considered a prototype representative of lexicalist-oriented descriptions, the MTT relocates a significant part of deep-syntactic information into the dictionary. Lexical Functions (LFs) have been proposed as the main means for capturing different relations among lexemes in the lexicon. They are defined as mathematical functions whose arguments and values are lexical units. Two types of LFs are distinguished,

namely, paradigmatic LFs and syntagmatic LFs. Paradigmatic LFs capture information about derivational and lexical-semantic relations in the lexicon (e.g., nominalizations and other derivatives, synonymy and antonymy relations). Syntagmatic LFs make it possible to represent a substantial part of syntactic information in the dictionary, for example, information on collocability and light-verb constructions. See examples of LFs (based on Wanner 1996):

- S_0 function outputs a semantically corresponding noun for the input word, e.g., $S_0(to\ analyze) = analysis$, $S_0(fast) = rapidity$,

- A_0 provides a semantically corresponding adjective for the input, for example, $A_0(fish) = fishy$, $A_0(countryside) = rural$,

- S_1 outputs an agent noun semantically related to the input word, for example, $S_1(drive) = driver$, $S_1(talk) = speaker$,

- similarly, S_2 is used to get a patient meaning noun, for example, $S_2(drive) = vehicle$,

- Syn provides synonyms, for example, $Syn(positive) = favorable$,

- $Magn$ determines for a noun or verb which lexeme it is typically combined with in order to intensify its meaning, for example, $Magn(patience) = infinite$,

- $Oper$ assigns a light verb to a noun in order to form a light-verb construction, for example, $Oper1(analysis) = carry\ out$ _ , $Oper2(analysis) = to\ undergo$ _ .

LFs of both types are a core element applied in the Explanatory Combinatorial Dictionary (Mel'čuk 2006; Mel'čuk and Žolkovskij 1984) and are used as a type of lexical strings in the deep-syntactic representation of the sentence. They are substituted for lexemes at the next, lower level (surface-syntactic representation).

Some features that are captured within the deep-syntactic representation in other approaches (e.g., topic-focus articulation in Functional Generative Description, see Section 3.3) are described at a separate, more abstract level, so-called semantic representation in MTT.

The multileveled scheme proposed by the MTT is applied in a corpus for Russian (SynTagRus) and in a treebank for Spanish (AnCora-UPF Treebank):

- In SynTagRus (Apresjan et al. 2006), Russian sentences were assigned a morphological annotation and a surface-syntactic dependency tree (these annotations are available at http://www.ruscorpora.ru/). In addition, a lexical semantic annotation and lexical-functional annotation were announced by Boguslavsky (2014). While lexical semantic annotation consisted in disambiguating ambiguous words that have different lemmas and/or different part-of-speech tags, the aim of the lexical functional annotation was to identify LFs and their arguments and values in the texts. Semantic annotation, as described by Apresjan et al. (2006), does not seem to be available with the SynTagRus data yet.

- In the Ancora-UPF Treebank (Mille, Burga, and Wanner 2013), Spanish sentences are annotated at four layers, namely, at the morphological,

surface-syntactic, deep-syntactic, and semantic one. The deep-syntactic annotation has been semi-automatically derived from the MTT-like surface-syntactic annotation, which was developed first on the basis of the AnCora corpus (Martí et al. 2007; see other AnCora-related treebanks in Section 3.4). The deep-syntactic annotation follows the principles summarized above: A sentence is represented as a rooted directed acyclic graph whose nodes correspond to content words and are connected with edges labeled with numbered argument labels. Coreference links are added to the graph. See Figure 2 for an example.

3.3 Functional Generative Description

Functional Generative Description was introduced by Sgall (1967) and since then has been elaborated in dozens of papers and monographs (e.g., Sgall, Hajičová, and Panevová 1986). The multiple-leveled scheme for description of the form–meaning continuum, as proposed by Sgall (1967), has been revised into a three-layer annotation style applied in the Prague Dependency Treebank (PDT; Hajič 1998; Hajič et al. 2006). With annotations at the morphological, surface-syntactic, and deep-syntactic (so-called tectogrammatical) layer, the PDT is "the first complex linguistically motivated treebank based on a dependency syntactic theory" (Hajič et al. 2017).

At the deep-syntactic layer, each sentence is represented as a dependency tree with labeled nodes and edges. The main characteristics are as follows:

- Structure: rooted tree or DAG,[10] verb as the core item. The nodes are ordered and, unlike in most other deep-syntactic frameworks, the deep order in FGD is not a mere projection of the surface word order. Instead, it reflects the information structure of the sentence (topic-focus articulation).

- Nodes generally correspond to content words, except for empty nodes (representing pro-dropped subjects and ellipsis in coordination) and for nodes in coordination and apposition structures, in which a coordinating conjunction is captured as the root of the structure in order to fit it into the dependency layout (cf. Section 4.4 for comparison of the Prague style with other frameworks). Nodes are labeled with a lexical string and with a set of attributes that capture semantically relevant grammatical categories (grammateme attributes), grammatical coreference, and basic features of information structure:

  – The lexical string corresponds to the basic (dictionary) form of a lexeme (infinitive with verbs, nominative singular with nouns, etc.; so-called tectogrammatical lemma), to the dictionary form of a lexeme's base word (with a limited set of highly regular and productive derivatives), to a non-lexical value (with prodrops),

  – Grammateme attributes are used to store information on grammatical categories that are relevant for the meaning of the

---

10 The tectogrammatical graph in PDT is a rooted tree only if coreference links are not considered edges. This is indeed the perspective taken in most descriptions of FGD and PDT; however, in the context of the present survey, coreference qualifies as a special type of edges. That makes the structure either a directed acyclic graph or a general directed graph, depending on the direction of the coreference edges.

sentence but can be inferred neither from the tree structure nor from the semantic role labels (e.g., the category of number),

- The coreference attributes capture the relations between a pronoun node and a noun or string it refers to, or between items governed by a control verb; although stored as node attributes in PDT, these relations can actually be viewed as a special type of (directed) edges added on top of dependency trees,

- The information structure attribute (topic-focus articulation) is used to represent whether a node belongs to the contextually bound or non-bound part of the sentence.

- Edges correspond to dependency relations (with the exception of coordination and apposition structures). They are assigned semantic role labels (functors). Semantic roles are divided into arguments and adjuncts according to both semantic and formal criteria specified within the valency theory by Panevová (1974–1975):

  - There are five argument roles (Actor, Patient, Addressee, Effect, Origin), which correspond mostly to the surface-syntactic slots of a subject and of direct and indirect objects of the verb (as the most prominent item of the sentence).

  - More than 50 adjunct roles are assigned with different types of temporal, local, and other circumstances of an event expressed by the verb.

  - Another approx. 10 labels distinguish between different paratactic syntactic structures such as coordination and apposition.

- Semantic role labeling is linked to the valency lexicon which specifies which of the roles constitute the valency frame of the verb (being either obligatory or optional).

- Annotation at the deep-syntactic layer is interlinked with annotation on the surface-syntactic layer and morphological layer.

The deep-syntactic annotation in the above-specified extent is available in the PDT from the version 2.0 onward (PDT 2.0, Hajič et al. 2006; PDT 2.5, Bejček et al. 2011). In version 3.0 (PDT 3.0; Bejček et al. 2013), the deep-syntactic annotation was enriched with annotation of further coreference types, bridging relations, discourse relations, genre specification, multiword expressions, and quotation.

The PDT annotation scheme, including the deep-syntactic representation, was implemented also in the PDT of Spoken Czech and in two parallel treebanks, namely, in the Czech-English Parallel Corpus and in the Prague Czech-English Dependency Treebank (PCEDT), whose English part contains the Wall Street Journal Section of the Penn Treebank. See figures 3 and 4 for examples of PCEDT annotation.

A handful of treebank projects adopted the PDT annotation scheme for other languages; however, only a few of them contain a sentence meaning representation; cf. Latin Dependency Treebank, whose deep-syntactic annotation is close to PDT, or Croatian Dependency Treebank, which uses 20 semantic role labels at the deep layer, whereas in the Greek Dependency Treebank, Slovene Dependency Treebank, or the
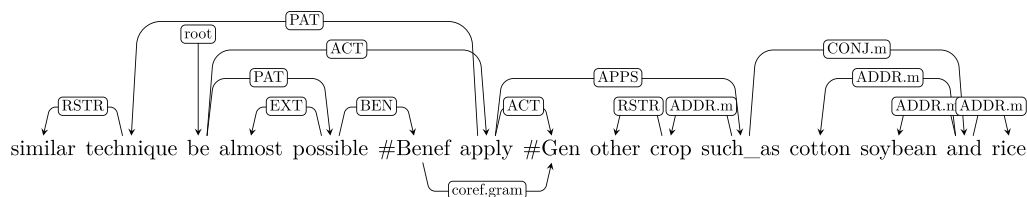
Figure 3
Deep-syntactic (tectogrammatical) representation of the sentence *A similar technique is almost impossible to apply to other crops, such as cotton, soybeans and rice.* in the Prague Czech-English Dependency Treebank (PCEDT, rooted in FGD). Nodes correspond to content words and are labeled with basic forms of the lexemes (tectogrammatical lemmas). Negation expressed by a prefix in *impossible* is captured by a grammateme value (cf. 3.3 and 4.7.2). Function words are linked to content words but have no nodes of their own; on the other hand, there are empty nodes for valency-licensed arguments that are not represented on the surface. The edges labeled APPS and CONJ are special: rather than expressing a relation between the source and the target node, they only mark the target node as a head of a paratactic structure (apposition and coordination, respectively). The .m suffix in edge labels marks members of paratactic structures (see Section 4.4). The coreferential edge depicted below the sentence shows that the benefactor argument of *possible* and the actor of *apply* are coreferential, even though none of them is overtly expressed on the surface.
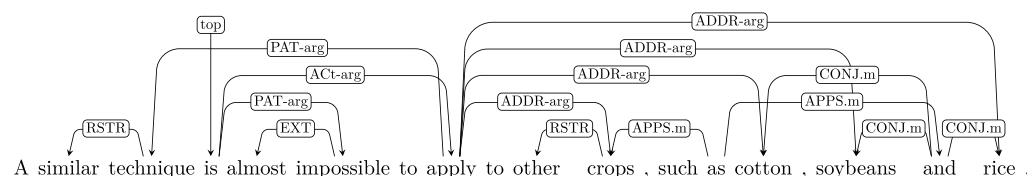
Figure 4
The so-called PSD graphs (standing for Prague Semantic Dependencies) were used in two SemEval (SDP) shared tasks (Oepen et al. 2015), as well as in the CoNLL 2019 shared task (http://mrp.nlpl.eu/). They result from a lossy conversion from the tectogrammatical annotation of PCEDT (Figure 3). The SemEval tasks assumed that all nodes correspond to surface tokens; hence there are no empty nodes and the coreference edge has disappeared, too. On the other hand, nodes corresponding to function words are not connected to the rest of the graph (although it would be possible to link them via secondary edges to "their" content word); in the original tectogrammatical graph, they are not nodes and thus not shown in Figure 3. Another difference is that dependencies in PSD are propagated across paratactic structures (coordination and apposition), while the edges labeled CONJ.m and APPS.m connect the members of the paratactic structure.

syntactic annotation in the Slovak National Corpus the deep-syntactic layer is not available.

3.4 Proposition Bank and Closely Related Resources

The Proposition Bank (PropBank) project aimed at "adding a layer of predicate–argument information, or semantic role labels, to the syntactic structures of the Penn Treebank" (Palmer, Gildea, and Kingsbury 2005, page 71). The project started by marking clause nuclei composed of verbal predicates and their arguments (predicate–argument structure); PropBank annotation pointed to constituents in the original Penn Treebank annotation (Kingsbury and Palmer 2002). Later, "modifiers of event variables" were added (e.g., Babko-Malaya et al. 2004), broadening the predicate–argument

structures with adjuncts. The main features of PropBank annotation as presented by Palmer, Gildea, and Kingsbury (2005) are:

- Structure: directed acyclic graph, typically consisting of multiple unconnected components. The nodes can be ordered following the surface word order.

- Nodes are constituents of the Penn Treebank surface tree (but in PropBanks of other languages, the surface structure may be a dependency tree). Predicates are represented by terminal nodes, arguments are represented by their highest-spanning non-terminal.
  - Predicates are disambiguated, divided into senses.
  - Arguments correspond to whole syntactic phrases (namely, noun phrases, prepositional phrases, and dependent infinite or finite clauses) as delimited in the surface-syntactic annotation.
  - Split (discontinuous) constituents are linked together and assigned a single semantic role.
  - Empty nodes (traces) represent subjects of controlled verbs, and they are (often) co-indexed with the node of the corresponding surface word.

- Edges go from predicates to their arguments. Co-indexing of traces with their antecedents can be viewed as a special type of edges.

- Arguments are assigned semantic role labels: `ARG0` to `ARG5`, `ARG0` being a "Prototypical Agent," `ARG1` a "Prototypical Patient or Theme," and so forth; the labels are assigned consistently for a given predicate across different syntactic alternations (as analyzed by Levin 1993), cf. [$_{ARG0}$*John*] *broke* [$_{ARG1}$*the window*] and [$_{ARG1}$*The window*] *broke*.

- Another label `ARGM` defined for adjuncts (which are not required by the verb but are part of the sentence). 11 `ARGM` subtypes are distinguished: `LOC, EXT, DIS, ADV, NEG, MOD, CAU, TMP, PNC, MNR, DIR`.

- Two other labels can be associated with numbered arguments (namely, `EXT` indicating a numerical nature of an argument, and `PRD` for secondary predication).

- A set of semantic roles defined for a verb sense (roleset) is associated with a set of syntactic frames (frameset).

The PropBank annotation (Figure 5) has been applied to multiple languages: Arabic Proposition Bank (Zaghouani, Hawwari, and Diab 2012), Chinese Proposition Bank, which was added to the Chinese Treebank (Xue and Palmer 2009), Finnish Proposition Bank (Haverinen et al. 2015), Hindi Proposition Bank (Vaidya et al. 2011), Persian Proposition Bank (Mirzaei and Moloodi 2016), Proposition Bank of Brazilian Portuguese (Duran and Aluísio 2012), Turkish Proposition Bank (Sahin and Adali 2018), Proposition Bank for Urdu (Nomani et al. 2016), and Basque Verb-Index (Estarrona, Aldezabal, and Díaz de Ilarraza 2018).
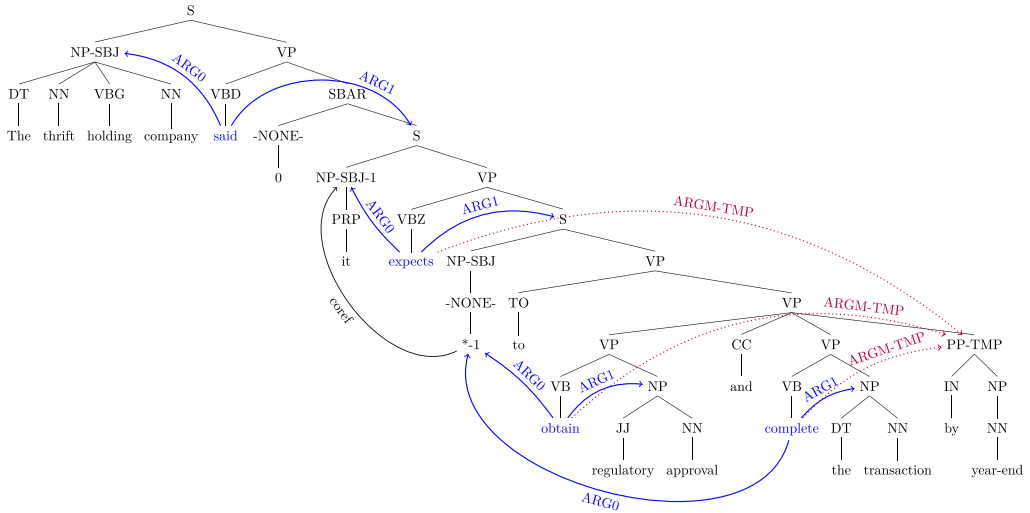
Figure 5
PropBank annotation over the constituents of the Penn Treebank for the sentence *The thrift holding company said it expects to obtain regulatory approval and complete the transaction by year-end.* The `ARGM-TMP` edge between *expects* and *by year-end* seems disputable but it appears in the annotated data, so we include it, too. Traces and their antecedents are connected to chains identifying grammatical coreference within sentence boundaries. Note, however, that the textual coreference between *it* and *the thrift holding company* is not annotated.

Pustejovsky et al. (2005) announced a project of merging the English PropBank with four other resources that focused on other parts considered as belonging to sentence meaning in English, namely with:

- NomBank (Meyers et al. 2004), in which argument structure was assigned with eventive nouns occurring in PropBank (data of the Wall Street Journal Corpus of the Penn Treebank). First, "markable" noun instances were identified among common nouns, that is, eventive nouns that are accompanied by a PropBank-defined argument or adjunct. In each noun phrase with such a noun, the head was identified and its arguments and adjuncts were marked and assigned a semantic role label from the PropBank label set (`ARG0` to `ARG5` and different `ARGM` labels; see `https://nlp.cs.nyu.edu/meyers/NomBank.html` for detailed annotation instructions).

- Penn Discourse Treebank (PDTB), in which the relations between propositions (i.e., meanings of individual clauses made up of a predicate–argument structure and related adjuncts) are annotated. Propositions are marked as arguments with regard to discourse connectives, which are either explicit or implicit (Miltsakaki et al. 2004a, 2004b).

- TimeBank (Pustejovsky et al. 2003), in which temporal features of propositions (expressed by temporal adjuncts, temporal prepositions and connectives, tensed verbs, etc.) and temporal relations between propositions are annotated.

- • Coreference Annotation created at the University of Essex, which contained texts from a subset of the Penn Treebank (Poesio and Vieira 1998) and the Gnome Corpus (Poesio 2004) annotated with coreference relations.

Merging these resources meant that the clause nuclei composed of verbal predicates and their arguments, as captured in PropBank, were broadened with the argument structures for instances of common nouns (NomBank) and, finally, the isolated islands were connected with discourse relations (PDTB). By also having an explicit temporal and coreference annotation, the initially limited focus of PropBank was substantially extended, providing a more complex semantic annotation than available in the particular resources. A more general goal of the merging project was to define a "Unified Linguistic Annotation" (ULA). ULA was presented at the ACL 2005 workshop *Frontiers in Corpus Annotations II – Pie in the Sky*, at the ACL 2007 Linguistic Annotation Workshop, at the ULA workshop co-located with TLT 2007, and several others. It was used in the Unified Linguistic Annotation Collection, which consisted of two corpora (The Language Understanding Annotation Corpus and REFLEX Entity Translation Training/DevTest) and was released by LDC in 2009. The former subcorpus contained English and Arabic texts annotated for temporal relations, coreference, committed belief, and dialog acts. The latter subcorpus consisted of English, Chinese, and Arabic texts translated into each of the other two languages; it contained named entity annotation and annotation of temporal features.

As another project, Akbik, Guan, and Li (2016) propose what they call the Universal Proposition Banks. They stick to PropBank-style annotation. They are interested in building proposition banks for other languages; they have done and evaluated it for German, French, and Chinese. They project the English frames across word alignments in the Open Subtitles parallel corpus (Lison and Tiedemann 2016). They currently cannot handle target verbs that can be only expressed as complex predicates in English (e.g., French *rentrer* corresponds to English *go home* or *come home*).

PropBank argument structure has been applied as one type of annotation to three languages in the OntoNotes project. In the final release of the data (OntoNotes 5.0; Weischedel et al. 2013), English, Chinese, and Arabic texts are assigned three types of annotation, namely, a surface-syntactic (Penn Treebank-style) annotation, the PropBank predicate–argument structure, and "shallow semantics," which consists of word sense disambiguation for nouns and verbs (with senses connected to an ontology) and coreference annotation.

With its primary focus on semantic role labeling of verb predicates, PropBank is related to yet another resource, VerbNet, which consists of hierarchically arranged verb classes that are based on Levin's approach (Kipper, Dang, and Palmer 2000; Kipper, Palmer, and Rambow 2002). For each class, subclass, and the corresponding set of verbs, a list of arguments (selected from a set of 23 roles in total) is assigned, and syntactic and semantic information is given. A two-step mapping between VerbNet and PropBank has been created (Loper, Yi, and Palmer 2007). First, a "lexical mapping" was applied to link VerbNet records to the verbs in PropBank. Second, if more mappings were offered for a verb, an "instance classifier" decided which of them is most appropriate.

PropBank-style annotation was adopted as a part of semantic annotation in AnCora treebanks for Spanish and Catalan (AnCora-Es and AnCora-Ca, Taulé, Martí, and Recasens 2008; both based on previous annotation efforts, Martí et al. 2007). The data are assigned morphological annotation, surface-syntactic (constituency) trees, and semantic annotation. In addition to PropBank semantic role labels (`Arg0` to `Arg4`, `ArgM`, `ArgA` for

so-called external agents, and `ArgL` for complementations of light verbs), the semantic annotation also contains thematic roles (`Agent`, `Cause`, `Patient`, and 17 other roles), word senses (with each noun, based on the respective derivative of WordNet), and named entity tags. More recently, AnCora-Es was enriched with annotation of implicit arguments of deverbal nominalizations (resulting in the Spanish Iarg-AnCora corpus, Taulé, Peris, and Rodríguez 2016).

3.5 FrameNet-Based Approaches

There is a (rather heterogenous) family of approaches in which semantic relations in a sentence are represented using the FrameNet semantics framework (Fillmore 1976; Baker, Fillmore, and Lowe 1998; Johnson et al. 2002).

In contrast to PropBank, FrameNet started as a primarily lexicographic project, from the definition of semantic frames, which consist of frame elements whose labels are chosen with regard to the particular situation. For instance, BUYER is one of the frame elements of the frames "Commerce_buy" and "Commerce_sell" whereas COOK is contained in the "Apply_heat" semantic frame; BUYER with the verb *to buy* and COOK with the verb *to cook* correspond to the `ARG0` semantic role in the PropBank label set. For each frame, a set of predicates is listed that evoke the particular frame.

Typically, the FrameNet-based annotation is added on top of an existing constituency treebank (analogously to PropBank annotation being added on top of Penn Treebank constituency trees), such as in the case of

- the original Berkeley FrameNet (Ruppenhofer et al. 2006), in which the core lexicographic database is accompanied by frame-annotated fragments of the Penn Treebank (besides other frame-annotated running-text samples which are not tied to any treebank annotation),

- in the SALSA corpus (Saarbrücken Lexical Semantics Annotation and Analysis), which is a resource developed for German (Erk et al. 2003; Burchardt et al. 2006) built on top of the TIGER Treebank (Brants et al. 2002),

- the annotation over the KAIST Treebank for Korean (Hahm et al. 2018).

Using an underlying constituency treebank simplifies delimitation of frame elements manifested in a given sentence, as well as of frame-evoking elements.

Figure 6 illustrates the overlapping of constituency trees with the FrameNet-based annotation in SALSA. In SALSA, verb predicates (and possibly nouns) are handled as frame-evoking elements:

- Structure: As in PropBank, all frames identified in a sentence make up an unconnected directed acyclic graph where edges connect elements of one frame. Nodes correspond to surface strings; therefore a partial order can be defined for them.

- Frame-evoking elements are assigned an appropriate FrameNet frame (cf. the verb *fordern* assigned the *Request* frame in Figure 6).

- For each frame-evoking element, frame elements are identified in the sentence and labeled according to the particular semantic frame (cf. the
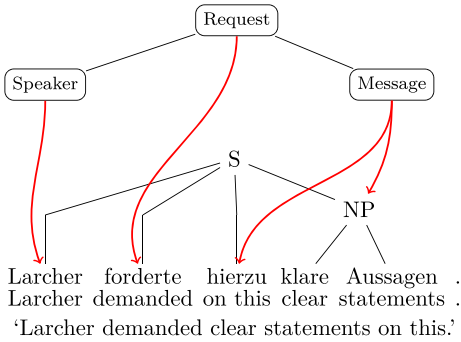
Figure 6
SALSA: Shallow constituent structure and frame structure of the sentence *Larcher forderte hierzu klare Aussagen.*'Larcher demanded clear statements on this.' (Adapted from Erk and Pado 2004).

> semantic role labels Speaker and Message in Figure 6). Frame elements
> correspond to syntactic phrases or their parts as delimited in the
> surface-syntactic annotation (in the TIGER corpus).

FrameNet offers much finer-grained inventory of roles compared with that of Prop-Bank. For more details on differences between the semantic role labeling in FrameNet and in PropBank see examples (1) to (3) vs. (4) to (6), and Ellsworth et al. (2004) for a comparison of semantic roles and some other linguistic phenomena (e.g., metaphor, light-verb constructions) in FrameNet vs. PropBank.

(1)   [$_{Goods}$*A car*] *was bought* [$_{Buyer}$*by Chuck*].

(2)   [$_{Goods}$*A car*] *was sold* [$_{Buyer}$*to Chuck*] [$_{Seller}$*by Jerry*].

(3)   [$_{Buyer}$*Chuck*] *was sold* [$_{Goods}$*a car*] [$_{Seller}$*by Jerry*].

(4)   [$_{Arg1}$*A car*] *was bought* [$_{Arg0}$*by Chuck*].

(5)   [$_{Arg1}$*A car*] *was sold* [$_{Arg2}$*to Chuck*] [$_{Arg0}$*by Jerry*].

(6)   [$_{Arg2}$*Chuck*] *was sold* [$_{Arg1}$*a car*] [$_{Arg0}$*by Jerry*].

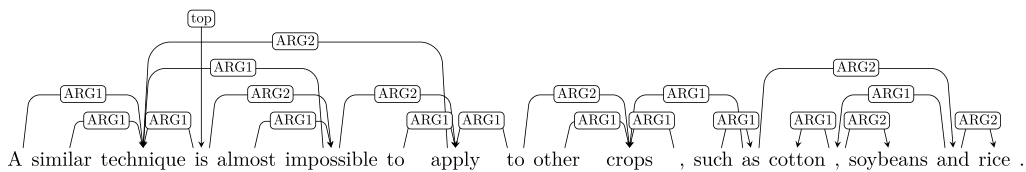### 3.6 Enju Predicate–Argument Structures

Enju (Yakushiji et al. 2005) is a parser trained on HPSG-style annotations automati-cally converted from the Penn Treebank; the parser further converts surface forms to Predicate–Argument Structures.[11] According to the authors, strong normalization of syntactic variations (illustrated in Table 3) should lead to more efficient Information Extraction. See also Figure 7 showing a full Enju graph. Basic features of the Enju structures are as follows:

- Structure: A sentence is assigned a set of predicate–argument structures
  (PAS). The term PAS is sometimes used to denote the graph constructed as
  a combination of all predicate–argument structures in a sentence (graph
  nodes = words, graph edges = binary relations from individual PAS); the

---

11 http://kmcs.nii.ac.jp/enju.

Table 3
Syntactic variation examples that all contain the predicate–argument structure
Entity1–ARG1–activate–ARG2–Entity2 in their Enju PASs (adopted from Yakushiji et al. 2005).

|	
Active Main Verb	*Entity1* recognizes and **activates** *Entity2*.
After an Auxiliary	*Entity1* can **activate** *Entity2* through a region in its carboxy terminus.
Passive	*Entity2* are **activated** by *Entity1a* and *Entity1b*.
Past Participle	*Entity2* **activated** by *Entity1* are not well characterized.
Relative Clause	The herpesvirus encodes a functional *Entity1* that **activates** *Entity2*.
Infinitive	*Entity1* can functionally cooperate to synergetically **activate** *Entity2*.
Gerund in PP	The *Entity1* play key roles by **activating** *Entity2*.

Figure 7
Enju Predicate–Argument Structures of the sentence *A similar technique is impossible to apply to other crops, such as cotton, soybeans and rice.* (Adapted from Oepen et al. 2015). Although the direction and the labels of the edges are deep-syntactic, all surface words including function words are included as graph nodes.

resulting graph is connected and acyclic (Hashimoto et al. 2014). Due to the direct correspondence between nodes and words, the nodes are totally ordered.

- Words are converted to their base forms and augmented with their POS tags. Every word in a sentence is treated as a predicate, an argument, or both.

- A predicate has a certain category and governs zero or more arguments.

- Predicate categories are relatively fine-grained and rather syntactically oriented, such as verb_arg123_relation for a verb that takes two NP objects or a verb that takes one NP object and one sentential complement, conj_arg12_relation for subordinating conjunctions that take two arguments, or aux_relation for an auxiliary verb; in total, 36 values are distinguished.[12]

- Types of arguments are coarse-grained, values such as `ARG1` (semantic subject), `ARG2` (semantic object), and `MODARG` (modifier) are distinguished.

- A predicate with all its arguments constitutes a PAS, which represent deep-syntactic relations between the predicate and its arguments.

---

12 https://mynlp.is.s.u-tokyo.ac.jp/enju/enju-manual/pas.html.
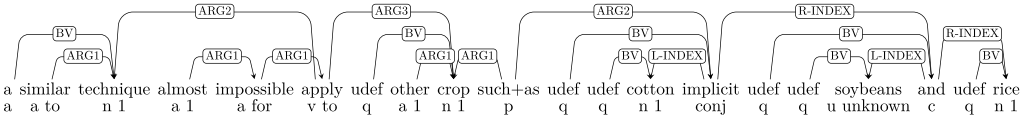
Figure 8
DELPH-IN Elementary Dependency Structure of the sentence *A similar technique is almost impossible to apply to other crops, such as cotton, soybeans and rice.* (Adapted from Ivanova et al. 2012.) There are six empty nodes labeled `udef q`, which represent underspecified quantifiers for the bare noun phrases. An additional empty node is labeled `implicit conj` and ties together *cotton* with *soybeans and rice*.
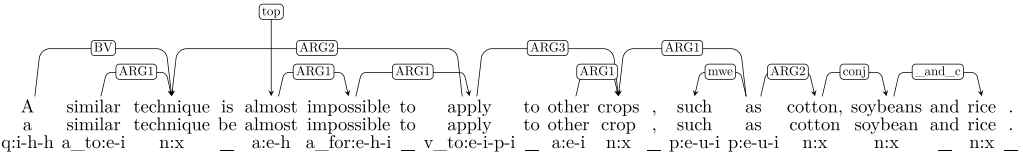
Figure 9
DELPH-IN Minimal Recursion Semantics-derived bi-lexical dependencies (DM) of the sentence *A similar technique is almost impossible to apply to other crops, such as cotton, soybeans and rice.* (Adapted from Oepen et al. 2015). In comparison to EDS (Figure 8), the underspecified quantifiers have been removed, coordination has been restructured as a chain headed by the first conjunct *(cotton)*, and the multiword expression *such as* is now formally treated as two nodes.

3.7 DELPH-IN Semantic Graphs

DELPH-IN[13] is a research network that constructs natural language descriptions using two formalisms: HPSG (Head-driven Phrase Structure Grammar; Pollard and Sag 1994) for the syntactic part, and MRS (Minimal Recursion Semantics; Copestake et al. 2005) for the semantic part. MRS is designed for smooth integration with HPSG, and it uses typed feature structures as its data structure. As such, it stands beyond the scope of the present survey (see also Section 2.2). However, there are also graph representations derived from MRS. First, a full-fledged underspecified logical form is reduced into a localized variable-free dependency graph, dubbed Elementary Dependency Structure (EDS; Oepen and Lønning 2006). EDS may contain graph nodes that do not correspond to individual surface words, for example, underspecified quantifiers for bare noun phrases or implicit conjunctions (Figure 8). An EDS graph node can be optionally annotated with a set of property-value pairs in order to store information (often determined morphologically) such as number and tense. In the second step, EDS is transformed into "pure" bilexical semantic dependencies (Ivanova et al. 2012, Figure 9). Both conversion steps are lossy. The target dependency representation is referred to as DM[14] and has been used in the SemEval shared tasks on semantic dependency parsing (Miyao, Oepen, and Zeman 2014; Oepen et al. 2015).

One of the resources created in DELPH-IN is DeepBank, a manual re-annotation of sections 00–21 of the WSJ corpus (Flickinger, Zhang, and Kordoni 2012). See Figure 9 for an example of a semantic dependency graph obtained by the MRS → EDS → DM conversion from DeepBank; see Figure 8 for the EDS of the same sentence.

---

13 `http://www.delph-in.net/`.
14 Standing for DELPH-IN MRS Bilexical Dependencies.

- Structure: EDS is a general directed graph; cycles are uncommon but possible. Some of its nodes are lexical units that correspond to surface words, multiword expressions, or subword units. There are also empty nodes. Following the anchoring of some nodes in the surface text, a partial order can be defined for the nodes.

- In contrast, the nodes of DM are surface tokens. Some of them are considered semantically void and unconnected to any other node in the graph.

- Edges correspond to predicate–argument relations between lexical units.

- In case of grammatical coreference, the same lexical unit (node) serves as argument of multiple predicates.

- Semantically ambiguous predicates and their valency frames are disambiguated.

- Coordination: Conjunction is treated as the head, that is, like a predicate, in EDS; empty nodes are used where an overt conjunction is not available. In DM, coordination is transformed to a left-to-right chain (see the "Mel'čuk/Moscow style" in Section 4.4).

3.8 Sequoia French Treebank

In the Sequoia corpus, the deep-syntactic representation (Candito et al. 2014) was built on top of the existing surface-syntactic representation (Candito and Seddah 2012), which followed the annotation scheme used in the French Treebank (Abeillé and Barrier 2004). The surface-syntactic annotation in the Sequoia corpus, originally based on constituent trees, was converted into dependencies and used for specification of the dependency-oriented deep-syntactic representation.

The main features of the deep-syntactic representation can be summarized as follows (Candito and Perrier 2016):

- Structure: directed graph; may contain cycles (Figure 10). The nodes can be ordered following the surface word order.

- Nodes of the graph correspond to content words. Function words are not included even if they are parts of multiword expressions (except for grammatical multiword expressions). Annotation of verbal multiword expressions was added in version 9.0 of the treebank (Candito et al. 2017).

- Edges correspond to dependency relations between content words. A node can be connected with the same parent by more edges with the same direction but different labels (in particular, in reflexive constructions, cf. Figure 11).

- When playing multiple roles in the sentence, a node can have multiple parents (Figure 11). For instance, infinitives are connected with the item that semantically fills its subject position whenever available in the same sentence (in raising and control constructions, in infinitival clauses modifying a noun, etc.). Similarly, if a noun is shared by several verbs in coordination (i.e., it is elided with one or more of the verbs) in the surface
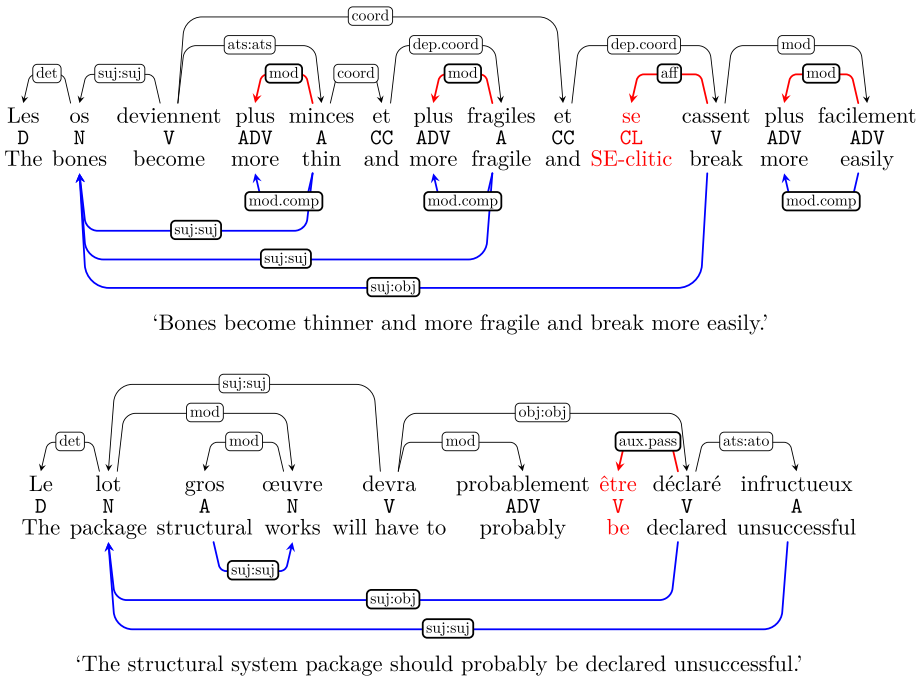
Figure 10
The complete representation of two French sentences in the Sequoia treebank (adopted from Candito et al. 2014). The red edges are surface-syntactic, the blue edges are deep-syntactic, and the black edges belong to both structures (surface and deep). The two edges connecting *gros* and *œuvre* thus exemplify a cycle in the deep graph. The double functions (for instance, `suj:obj`) specify the final grammatical function (here, subject) first and the canonical grammatical function (object) after that. Note that the labels of deep edges contain both functions (while the surface structure involves only final functions).
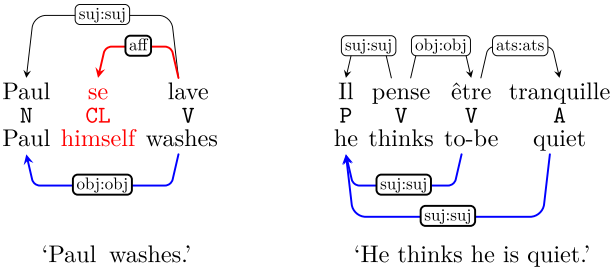
Figure 11
Two examples illustrating non-tree structures in Sequoia (adopted from Candito and Perrier 2016). In the left example, two deep edges go from *lave* to *Paul*, one marking *Paul* as the canonical subject and the other as canonical object. In the right example, three incoming edges mark the pronoun *il* 'he' as the canonical subject of three different predicate nodes.

structure, in the deep-syntactic representation all verbs are connected with the shared item. Moreover, each adjective is linked by a deep-syntactic edge to either the noun it modifies (adjectives in attributive position), or to both the verb and its subject or object (adjectives in predicative position).

- Edges are labeled with "canonical grammatical functions," which are basically surface-syntactic functions that the node would be assigned with the particular verb in a finite form in a non-elliptical construction (the account of grammatical functions is rooted in the Relational Grammar by Perlmutter 1980). For instance, an object with the preposition *by* in a passive sentence fulfills the canonical grammatical function of a subject; a noun is assigned as a canonical subject with both the finite form of a control verb and the dependent active infinitive verb in a controlled position (with the controlled subject).

- In this way, the deep-syntactic representation abstracts from syntactic diatheses (active vs. passive clause). However, only those diatheses that are marked overtly in the surface sentence (e.g., by the preposition *by*) are assigned the same deep-syntactic annotation; for instance, the same set of canonical functions is assigned with a verb in an active sentence and in its passive counterpart with a subject introduced by a preposition. On the contrary, diatheses without overt marking are not linked in the deep-syntactic representation.

- Semantically ambiguous predicates are not disambiguated.

- Coordination: The head of the first conjunct is the head of the coordinating structure.

The deep-syntactic representation combines with the surface-syntactic representation into a "complete representation." Unlike the deep-syntactic representation using directed graphs, the surface-syntactic structure is represented by a rooted tree. The surface-syntactic tree consists of nodes corresponding to all words in the particular sentence and of edges labeled with "final grammatical functions," namely, surface-syntactic relations of subject, object, and so on, with regard to the particular diathesis. For instance, a noun with the preposition *by* with a passive verb is assigned the final grammatical function of an object in the surface structure.

When displayed in a linear sentence, all words of the sentence are parts of the surface-syntactic representation and are connected with final grammatical functions. Those nodes that correspond to content words in the sentence enter the deep-syntactic representation and are assigned also a canonical grammatical function; see Figure 10.

3.9 Abstract Meaning Representation

Abstract Meaning Representations (AMRs) introduced by Banarescu et al. (2013) represent sentences as graphs in which non-leaf nodes stand for variables and only leaf nodes capture lexical content (i.e., only leaves are labeled with concepts). An example of such structure is depicted in Figure 12. Compared with most other approaches under our survey, the correspondence between AMR structures and surface-syntactic structures such as surface dependency trees is relatively limited, as the origins of AMR go back rather to a knowledge representation tradition.

- Structure: A directed graph, typically acyclic, although cycles are not completely excluded (Kuhlmann and Oepen 2016). Any correspondence between nodes and surface strings is hidden by design, hence the nodes are unordered.
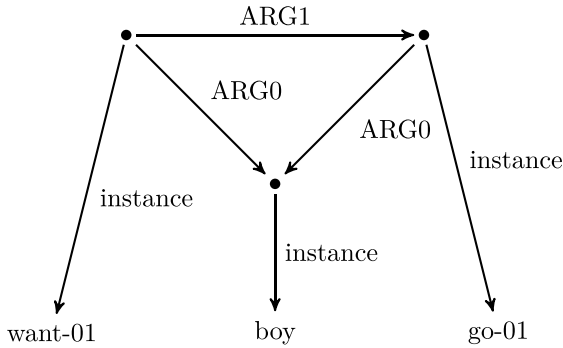
Figure 12
Abstract meaning representation of the sentence *The boy wants to go.* (Adapted from Banarescu et al. 2013).

- When an entity plays multiple roles in a sentence, variable nodes can have multiple parents. In this way, AMR abstracts away from co-reference devices like pronouns, zero-pronouns, reflexives, control structures, and so on. However, AMR annotates sentences independent of context, so if a pronoun has no antecedent in the sentence, its nominative form is used.

- AMR makes use of PropBank framesets to abstract away from English syntax (though AMR is not claimed to be an Interligua); verb frames are not assigned only to verbs, but also to their derivations such as deverbal nouns.

- AMR uses basic PropBank style labels `ARG0`–`ARG5` for core arguments. Around 40 additional semantic roles such as :condition, :direction, :duration, and :manner are distinguished, as well as around 20 roles expressing quantities and values such as :day and :year.

- Time and location prepositions are preserved in the semantic role value if they carry additional semantically indispensable information such as in :prep-against and :prep-on; subordinating conjunctions are treated analogously (altogether around 20 distinct values).

- In total, there are more than 100 role values distinguished according to the current web documentation of AMR,[15] including rather technical roles :snt1-snt10 that serve for merging multiple subsequent sentences into a single structure if needed.

- AMR represents negation with :polarity, both in the case of clause negation and in the case of word-formation negative prefixes (*inappropriate* → `appropriate :polarity -`).

- AMR does not represent semantic counterparts of inflectional categories expressing tense and aspect (however, an AMR augmentation for capturing tense and aspect because of their importance in NLP

---

15 https://www.isi.edu/~ulf/amr/lib/roles.html.

applications has been suggested recently by Donatelli et al. 2018). AMR omits articles.

In spite of the fact that Banarescu et al. (2013) openly admitted that AMR is heavily biased toward English, AMR has been applied on a variety of other languages.[16]

3.10 Universal Conceptual Cognitive Annotation (UCCA)

Like AMR, the Universal Conceptual Cognitive Annotation (UCCA; Abend and Rappoport 2013) puts words on leaf nodes only. However, UCCA is still anchored in the surface text and all surface words have their nodes. Here are some basic characteristics of UCCA graphs:

- Structure: A directed acyclic graph. A partial order can be defined for the nodes as some of them correspond to surface strings.

- Leaves of the graph are called *terminals* and represent atomic meaning-bearing units (words and multiword chunks). Their labels are surface forms rather than lemmas. *Nonterminal* nodes bear no labels, but edges do. A nonterminal node can be characterized in terms of the categories of its outgoing edges.

- The nodes of an UCCA graph are also called *units*. More precisely, the unit represented by a nonterminal is the subgraph headed by the nonterminal, and it contains embedded smaller units. Terminal nodes are atomic units. A unit in UCCA "expresses a relation along with its arguments." See Figure 13 for an example. A unit may cover discontinuous parts of the text.

- An important special type of nonterminal unit is called *scene*. It describes a movement, action, or temporally persistent state, which is the scene's *main relation*.[17] The main relation is represented as a sub-unit, often a verb, but it can also be an adjective, an eventive noun, and so forth. Other sub-units are participants of the scene. There can also be secondary relations, which are marked as *adverbials* (Figure 14).

- There may be *implicit* terminal units that do not correspond to a stretch of text. For example, *playing games is fun* has an implicit sub-unit, connected via an A edge, and corresponding to the people playing the game. In our terminology, implicit terminals are empty leaf nodes.

- A unit may participate in more than one relation; that is why the graph is not necessarily tree (Figure 14).

- Relations are labeled with coarse-grained categories; the inventory contains 12 values such as P – Process, A – Participant, D – Adverbial, E – Elaborator, and N – Connector.

---

16 See https://nert-nlp.github.io/AMR-Bibliography/. In some languages this required language-specific modifications of the annotation scheme because of phenomena that have no analogy in English, viz. coreference of noun classifiers in Vietnamese as discussed by Linh and Nguyen (2019), or third-person clitic pronouns in Spanish as discussed by Migueles-Abraira, Agerri, and de Ilarraza (2018).

17 Unlike some other frameworks, in UCCA the term *relation* does not mean an edge. It is one of two types of concepts of which an utterance is constructed, the other being an *entity*.
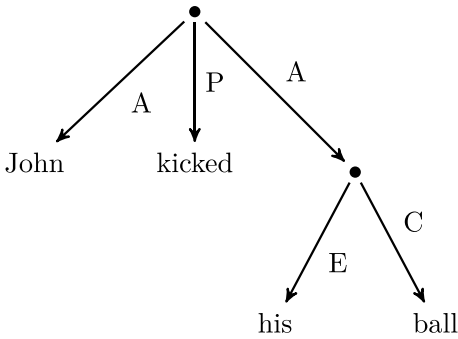
Figure 13
UCCA graph of the sentence *John kicked his ball* (Adapted from Abend and Rappoport 2013). The non-scene unit *his ball* is represented as a subgraph with one non-terminal and two terminal nodes; the C edge marks *ball* as the "center," while E means "elaborator." The other non-terminal represents a scene (roughly corresponding to the semantic content of a clause) and its main relation is a "process" (as opposed to a "state"), hence the edge label P. Participants of the process are attached via edges of the category A (besides arguments, A can also attach locations).
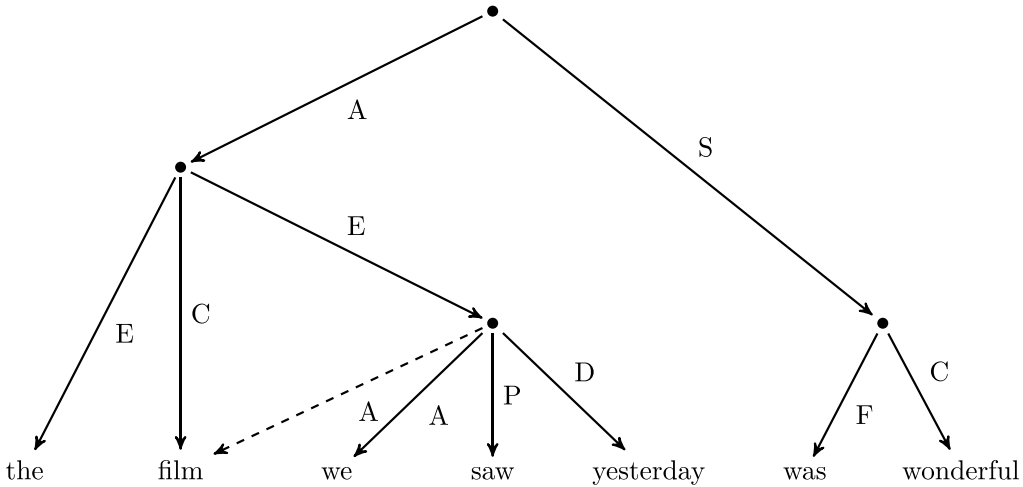
Figure 14
UCCA graph of the sentence *the film we saw yesterday was wonderful* (Adapted from Abend and Rappoport 2013). There are three new edge categories: S denotes a main relation that is a state rather than a process; D is an adverbial modifier of a scene; and F ("function") is a word in a non-scene unit that is not an elaborator. Note that the terminal node *film* participates in two larger units: it is the center of *the film we saw yesterday*, and it is also a participant in the scene *we saw film yesterday*. Note that UCCA distinguishes primary edges (C to *film*) from remote edges (A to *film*).

- UCCA annotates text, which typically comprises multiple sentences and paragraphs. Linkage of scenes can cross sentence boundaries.

UCCA has been designed as a multilayer framework. What we describe here is referred to as the *foundational layer* in UCCA to which additional layers can be added that "may refine existing relations or otherwise annotate a complementary set of distinctions" (Abend and Rappoport 2013). In a recent paper (Prange, Schneider, and Abend

2019), coreference annotation is proposed as a layer to be annotated on top of the UCCA foundational layer. Coreference links are assigned with the units as delimited in the foundational layer.

UCCA does not model syntax explicitly or build on other annotation layers, assuming that semantic annotation can be mapped directly to surface form.

UCCA is relatively insensitive to syntactic variation, giving similar analyses to syntactically different but semantically close scenes. This increases parallelism both within a language and across languages. For example, compare the annotations of English *John took a shower* and *John showered*: in both cases, we have a single scene with one participant and with the main relation whose center expresses the notion of showering: (1) *John$_A$ [took$_F$ [a$_E$ shower$_C$]$_C$]$_P$* (2) *John$_A$ showered$_P$*. The structure is also preserved under translation to other languages, such as German (*John$_A$ duschte$_P$*, lit. *John showered*) or Portuguese (*John$_A$ [tomou$_F$ banho$_C$]$_P$*, lit. *John took shower*.

UCCA is not tied to a particular lexical resource. The authors mention some resemblances with the FrameNet project, where frames can be seen as a context-independent abstraction of UCCA's scenes.

There is an English Wikipedia corpus of approximately 5,000 tokens annotated in UCCA; there is also a pilot annotation of a parallel English-French-German corpus, and a trainable parser.[18] Parsing text into UCCA was the topic of a SemEval shared task in 2019, and UCCA was also one of five target schemes (along with AMR, the FGD-based PSD scheme, and DELPH-IN's DM and EDS) of the 2019 CoNLL shared task[19] (Oepen et al. 2019).

3.11 Enhanced Universal Dependencies

UD[20] (Nivre et al. 2020) is a community project that strives to define one set of guidelines for morphological and syntactic annotation that could be applied to all human languages. It defines two layers: the basic representation and the enhanced representation. The former must be a rooted tree, while the latter is a connected directed graph. Even the basic tree is somewhat closer to semantics than is usual in surface dependency treebanks. UD puts considerable emphasis on cross-linguistic parallelism, and one way of making the structures parallel is to push function words out and make them leaves of the tree. Furthermore, function words are attached to content words via specific relation types, hence they can be viewed as mere features of content words (cross-linguistically it makes sense because in other languages they may correspond to morphological features of the content word).

The enhanced representation (Schuster and Manning 2016) is still less developed. Its concrete specification exists since UD guidelines version 2 (December 2016)[21] and there are still discussions about whether and how the specification should be extended. In any case, enhanced UD does not aspire to become a full-fledged deep structure of the utterance, in any sense of the term. The current guidelines consist of five separate "enhancement areas" whose common denominator could be roughly described as "phenomena that may be useful for downstream language-understanding tasks such as relation extraction, and that cannot be represented using a labeled rooted tree." This

---

18 See the resource list at `http://www.cs.huji.ac.il/~oabend/ucca.html`.
19 `http://mrp.nlpl.eu/`.
20 `https://universaldependencies.org/`.
21 `https://universaldependencies.org/u/overview/enhanced-syntax.html`.

objective ultimately introduces certain annotation items that we find in other "deep" (semantic) frameworks.

As of UD release 2.5 (November 2019), only 16 languages have a UD treebank with some enhanced structures (while there are 90 languages with basic treebanks). Moreover, the five defined enhancements are all optional and a treebank[22] may choose only to annotate some of them.

- Structure: Directed graph (connected, but not necessarily acyclic). The nodes are ordered following the surface word order. Annotation guidelines do not specify the position of empty nodes, thus the node order is partial.[23]

  - All the nodes of the surface (basic) tree also participate in the deep (enhanced) graph but dependency relations (edges) may be added as well as removed.

  - As in basic UD, orthographic words may be declared multiword tokens and split to several syntactic words (nodes).

  - Empty nodes may be added in cases of gapping and stripping (ellipsis with coordination). Their attributes *may* be copied from the corresponding node in the overt conjunct.

  - Types of added relations are from the same inventory as those of basic tree structure (with one exception: `ref`). The main distinctions: does it modify a clause, or a nominal? And is the modifier itself a clause, a nominal, or another modifier word? Is it a core argument, or an oblique argument/adjunct? (UD claims not to mark argument/adjunct distinction, nor the semantic roles.)

  - As in basic UD, the emphasis is on relations between content words. Adpositions, articles, auxiliary verbs, and other function words are attached as leaves to "their" content word and can also be understood as features of the content word.

  - Coordination: Stanford-style backbone with additional propagated relations to/from non-head conjuncts. Representation of nested coordination is limited.

  - Ellipsis other than gapping and stripping is solved by promotion, that is, there is no overt annotation that announces the ellipsis.

- The guidelines list five cases where edges and nodes are added and removed with respect to the basic representation. All of them are optional, that is, a structure is enhanced if it contains just one of the possible enhancements. However, UD treebanks are not expected to use the mechanism for other enhancements that are not listed.

---

22 For simplicity, the corpora are still called treebanks although the enhanced representation is no longer a tree. A UD treebank always contains the basic trees and, optionally, there might be the enhanced graph encoded side-by-side with the tree.

23 In fact, the data format requires that even empty nodes get a specific position with respect to surface words, but this position is arbitrary.
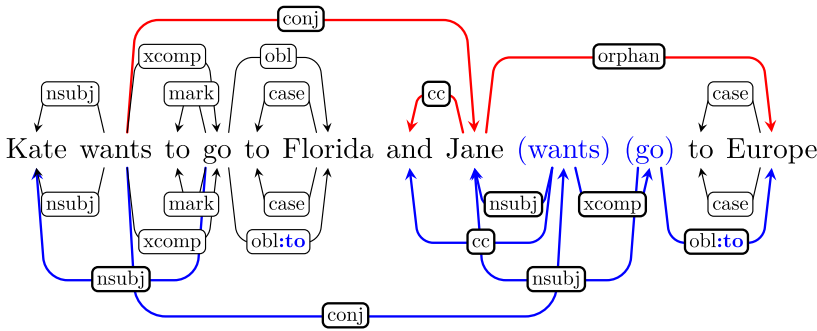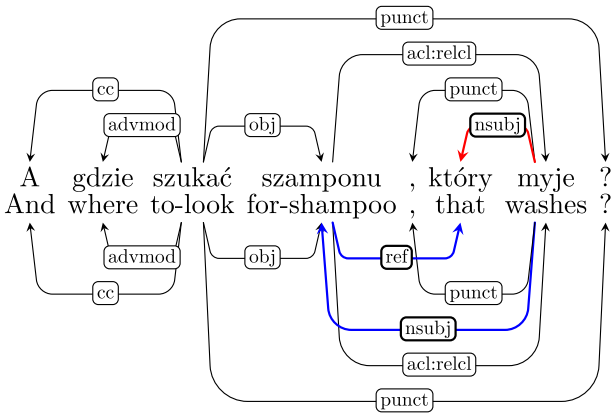
Figure 15
Basic (above) and enhanced (below) UD representation of an English sentence, featuring empty nodes that reverse gapping, subject relations projected across control verbs, and adjunct relation labels enhanced with the preposition lemma.
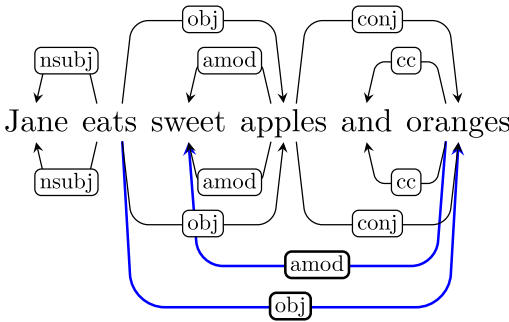
Figure 16
Basic (above) and enhanced (below) UD representation of an English sentence, showing dependencies projected to and from the second conjunct in coordination.

- Controlled / raised subjects.

- Ellipsis with coordination (gapping and stripping).

- Propagation of dependencies across conjuncts.

- Relative clauses (modified nominal re-attached as argument of the relative clause).

- Case (or adposition) added as a language-specific subtype of the dependency relation. This could be viewed as a sort of "semanticization" of the edge labels but unlike in the basic set of relations, there is no attempt to unify the labels across languages. Quite the contrary: the lemma of a preposition becomes part of the label even if it is a Chinese character.

See figures 15, 16, and 17 for examples of basic vs. enhanced UD graphs.

'And where to look for shampoo that works?'

Figure 17
Basic (above) and enhanced (below) UD representation of a Polish sentence, showing relation reversal in relative clauses. Note the directed cycle between the words *szamponu* and *myje*.

3.12 More Surfacy Approaches

The syntactic-semantic opposition is clearly not black and white, and there are a number of approaches in which only a few ideas that could be considered "deep" are implemented, often in a rather ad-hoc heuristic manner and without any underlying theory; the motivation for such partial extensions often comes from the application perspective. An exhaustive overview of such approaches is most likely impossible. For illustration purposes, we select (admittedly arbitrarily) only a few such "mildly deep" approaches and describe them briefly, without including them in the systematic comparison in the next section:

- Microsoft's machine translation system described by Menezes and Richardson (2003) makes use of Logical Forms similar to those introduced by Jensen (1993); the expected advantage of using Logical Forms for such a purpose is that "additional generality obtained by normalizing both the lexical and syntactic form of examples, they may then be matched and applied more broadly when new sentences are translated." A Logical Form is an unordered graph representing the relations among the most meaningful elements of a sentence. Nodes are identified by the lemma of a content word and directed, labeled arcs indicate the underlying semantic relations.

- Filippova and Strube (2008) present an unsupervised method for sentence compression which relies on a dependency tree representation and shortens sentences by removing subtrees. A tree of an original sentence is pruned, that is, edges are removed in an optimized way so that retained edges form a valid tree and their total edge weight is maximized. Finally, a shortened sentence is synthetized from the pruned compression tree. The trees to be pruned result from a transformation of surface dependency trees. During this transformation, function words like determiners, auxiliary verbs, and negative particles are removed from the tree and

saved as attributes of their lexical heads. Nodes (corresponding to content words) are labeled with their lemmas.

- PropS introduced by Stanovsky et al. (2016) is an automatic converter of Stanford dependency trees into so called proposition structures in the form of directed graphs. Because of downstream applications, semantically equivalent yet syntactically different constructions should receive the same representation, and thus "non-core syntactic details" are hidden during the conversion (e.g., auxiliary words are turned into features, compound word forms are merged into single nodes).

- Very similarly to PropS, PredPat described by Zhang, Rudinger, and Durme (2017) is a software system that extracts predicates and arguments from surface dependency trees, this time from UD-shaped trees. Manually designed patterns that are claimed to be language-agnostic are used for the extraction.

4. Commonalities and Differences in Handling Core Linguistic Phenomena

Basic structural properties as listed for the individual frameworks in Section 3 are summarized in Table 4. A more detailed discussion of the particular phenomena follows. Section 4.1 compares the data-structure types of the sentence meaning representations. Section 4.2 reviews how the sentence meaning representation is related to the surface representation with a special focus on the opposition of content and function words, the synonymy issue, and node order. Section 4.3 discusses the accounts of valency and semantic role inventories across the frameworks. Attention is paid also to paratactic structures (Section 4.4), deletions and coreference (Section 4.5), discourse relations (Section 4.6), inflectional and derivational morphology (Section 4.7), and complex word forms (Section 4.8).

4.1 Basic Data-structure Types of Deep Representations

The surveyed frameworks differ in what class of graphs they use to describe semantic relations in the sentence: For example, some frameworks allow cycles and some do not. It is also important to note that a framework typically defines several types of relations (edges), and certain graph properties may only hold for edges of a particular type. Furthermore, some surface words may be considered semantically void and not awarded a graph node; if we treated them as nodes, the graph would not be connected. On the other hand, the graph may contain empty nodes that do not correspond to any surface word.

Paninian structures are rooted trees. The same can be said about the main tectogrammatical structure of FGD; however, when coreference relations are added, the graph is no longer a tree, and some coreference links even cross the sentence boundary.

The deep structure of MTT is a rooted directed acyclic graph. Individual projects inspired by MTT do not necessarily stay within the DAG constraints: For instance, the coreference edges in AnCora-UPF (Figure 2) cause cycles. Directed acyclic graphs as the data structure are used also in AMR and UCCA.

In PropBank, the individual predicate–argument structures are simple rooted trees, just the root and leaves. But the representation of the sentence is a general graph: Some nodes may participate in multiple predicate–argument structures, some in none. Some proposition banks are built on top of dependency treebanks, for example, the

Table 4
A simplified overview of characteristics of the frameworks surveyed in Sections 3.1 to 3.11.

Framework (Deep representation name)	Sentence Meaning representation structure	Surface-depth interface	Nodes and their order	Edge type inventory	Semantically relevant morphological categories	Coref.
Panini	rooted tree	overlap in a single structure	syntactic phrases (nominal/ verbal chunks); surface-related order	karaka relations (k1,k2…)	—	(yes)
MTT / deep-syntactic representation	rooted directed acyclic graph (on top of dep. trees)	two separate structs.	content words; surface-related order	I, II, …& ATTR, COORD, APPEND	yes (grammemes)	yes
FGD / tectogrammatical representation	rooted tree plus addit. edges (on top of dep. trees)	two separate structs.	content words; specific order	ACT, PAT, ADDR, ORIG, EFF & 10 roles for paratact. & 50+ adjunct roles	yes (grammatemes)	yes
PropBank and related annotations	directed graph (on top of const. trees)	two separate structs.	syntactic phrases; surface-related order	ARG0-ARG5 & ARGM*	—	yes
FrameNet-based approaches	directed graph (on top of const. trees)	two separate structs.	syntactic phrases; surface-related order	FrameNet frame element labels (Buyer, Seller, Speaker…)	—	—
Enju / PASs	directed graph (on top of HPSG)	two separate structs.	tokens; surface-related order	ARG1, ARG2, …	—	yes
DELPH-IN dependency structures EDS & DM	directed graph (on top of HPSG)	two separate structs.	content words; surface-related order	ARG1, ARG2,…& conj, mwe…	yes	yes
Sequoia / deep-syntactic representation	directed graph (on top of const. trees)	overlap in a single structure	content words; surface-related order	subject, object, mod,…	—	yes
AMR	directed acyclic graph	no direct relation to surf.	concepts, nonterminals; no order by design	ARG0-ARG5 & 40+ general semantic roles …	only negation, recently also tense and aspect	yes
UCCA	directed acyclic graph	no direct relation to surf.	tokens, nonterminals; surface-related order	12 semantic categories: P - Process, A - Participant, D - Adverbial…	—	yes
UD / enhanced repr.	directed graph (on top of dep. trees)	two separate structs.	tokens; surface-related order	37 universal syntactic relations: nsubj, obj, conj, det…	—	yes

Finnish PropBank (Haverinen et al. 2015). Nevertheless, the original English PropBank is defined over the constituent trees of the Penn Treebank, and arguments are phrases rather than individual words (see Figure 5); it is possible that the argument phrase contains other predicates and arguments. Finally, when we consider the combination of the PropBank with the PDTB, there will be edges that cross sentence boundaries, hence a graph corresponds to a larger portion of the discourse.

The remaining frameworks need general directed graphs, even if still somewhat restricted at places. The graph in enhanced UD is typically quite close to the rooted tree of the basic representation and for some sentences the two structures are identical. Propagation of dependencies across coordination may cause the tree to become a DAG,[24] and so can propagation of arguments of control verbs. Cycles appear only in sentences with relative clauses. The deep structure in Sequoia bears some similarities to UD: It is also relatively close to the surface tree but may include additional paths and even cycles.

The dependency graphs derived from DELPH-IN MRS and Enju PAS are connected as long as semantically void function words are not counted as nodes. Although a typical graph in these frameworks seems to be an undirected tree, cycles are not completely excluded. Kuhlmann and Oepen (2016) note that a small percentage of DELPH-IN EDS and AMR graphs contain cycles, and that the same would hold even for DELPH-IN DM if graphs with cycles were not excluded from the published data set. Nevertheless, they do not give examples nor discuss the linguistic perspective of the problem.

4.2 Surface vs. Depth

4.2.1 Relation Between Surface and Deep Representations. It is not uncommon that a framework recognizes multiple layers of analysis, with different layers for surface syntax on the one hand, and deep syntax or semantics on the other hand. However, several frameworks surveyed in this article have only one layer of representation (which seems to be at least partially "deep"). These include Panini, AMR, and UCCA.

In contrast, the following frameworks explicitly distinguish surface and deep relations:

- MTT: Surface representation vs. deep representation. There are two independent graphs. (In addition, there is a third layer called *semantic representation*; deep syntax, which is the layer we focus on in the present survey, lies between surface syntax and semantics.)

- FGD: Surface-syntactic layer (called 'analytical' layer in the Prague Dependency Treebank) vs. tectogrammatical layer. There are two separate trees for each sentence, but their nodes are interlinked.

- PropBank: A new annotation layer built on top of existing surface structure: the Penn Treebank constituent trees in the case of English, the Turku Dependency Treebank in the case of Finnish, and so on. Selected words are connected by new predicate–argument edges. In theory these edges can be viewed as building a separate graph, although in practice a PropBank file contains stand-off annotation where nodes are just references to the original (surface) tree.

---

24 There may be multiple roots if the top-level predicates are coordinated.

- FrameNet-based approaches: Analogously to PropBank, additional
  annotation level of frame semantics (this time based on FrameNet, though)
  is added on top of constituency trees of the TIGER Treebank.

- In ENJU, predicate–argument structures are obtained automatically from
  HPSG structures, which in turn result from a conversion of Penn Treebank
  or similar constituent trees; while these could be considered the
  corresponding surface representation, Enju is still a relatively independent
  project.

- In DELPH-IN, the complete framework consists of a syntactic and a
  semantic component, viz. an HPSG derivation and a MRS structure; each
  HPSG feature structure deterministically determines an MRS. The two
  graph representations discussed in this survey are derived from MRS
  through simplification. First, EDS is extracted from MRS, then it is further
  reduced and converted to DM (DELPH-IN MRS Bilexical Dependencies).

- Sequoia: Surface representation vs. deep representation. Deep edges
  connect only content words. Some edges are shared between both graphs,
  some edges appear only in the surface tree, and some only in the deep
  graph.

- Universal Dependencies: Basic representation vs. enhanced representation.
  The two graphs are stored in the same file side-by-side. In many cases the
  basic tree is a subset of the enhanced graph but it is not guaranteed:
  Sometimes a basic edge is omitted from the enhanced graph.

Clearly, the nature of interlinking deep-syntactic representations with correspond-
ing surface dependency trees (Figure 18) differs from the case in which constituency
trees are used to represent surface syntax (Figure 19). Conceptually, interlinking two
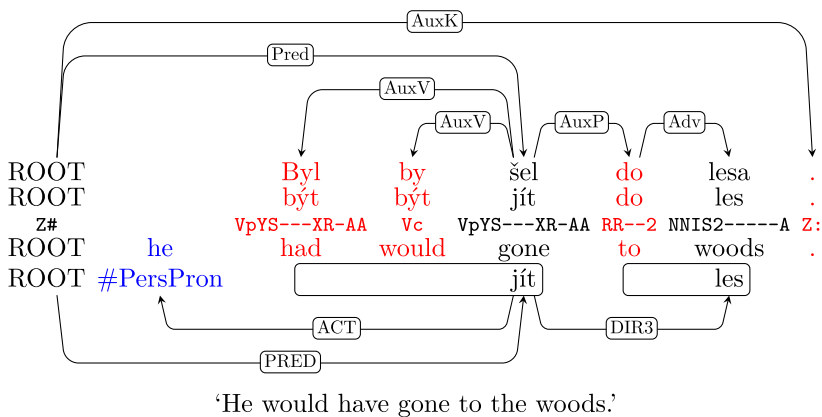types of dependency trees seems more straightforward.

'He would have gone to the woods.'

Figure 18
Interlinking between two dependency structures in FGD: The surface-syntactic (analytical) tree
is above the sentence, the deep-syntactic (tectogrammatical) graph is below, and the middle
line contains morphological tags (adopted from Hajič et al. 2006). The artificial root node and
the content words *šel* 'went' and *lesa* 'woods' are nodes in both graphs, although in the
deep-syntactic graph they are represented by their lemmas *jít* and *les*, respectively. The red
function words *byl*, *by*, *do*, as well as the punctuation, are nodes in the surface graph but not in
the deep graph; in the deep graph, function words are converted to mere attributes of content
words, which is indicated by the boxes. Finally, there is a blue empty node that exists only in the
deep graph and corresponds to a dropped personal pronoun.
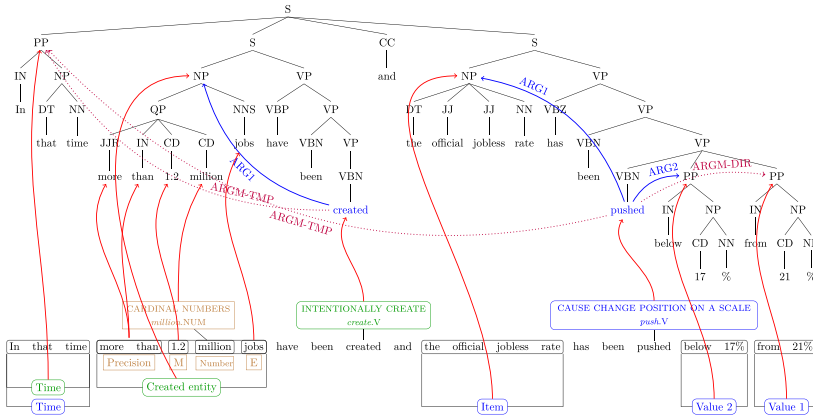
Figure 19
Constituency-dependency interlinking between Penn Treebank constituency trees and PropBank predicate–argument structure (top). FrameNet frame-semantic structures (bottom) for the same sentence (*million, create, push* as lexical items evoking semantic frames; adapted from Das et al. 2014).

4.2.2 Content vs. Function Words. The precise definition of function words varies across languages and frameworks; vaguely put, it is a class of words that are important for syntax, but their semantic content is negligible. They are also called auxiliary or semantically void. Words that are not function words are content words (also called autosemantic).

In practice, there is a scale rather than a sharp boundary, and frameworks must decide where to draw the line. For example, we may observe the following scale of verb types: (1) auxiliary verbs or particles used to construct periphrastic tenses, passive and the like (*to be, to have*); (2) modal verbs (*can, must*); (3) quasi-modal verbs (*ought to, used to*); (4) phase verbs (*to begin, to stop*); (5) aktionsart-modifying verbs (*to keep*); (6) lexical (content) verbs (*to kill, to eat*). Another blurry border is that of multiword or secondary prepositions (postpositions, conjunctions) on one side (such as *because of, in spite of, according to*), and the core closed class on the other.

Intuitively, content words are more important for any semantic description than function words. Indeed, many frameworks treat function words differently or ignore them completely. We observed the following approaches to the content-function distinction:

- No difference – function words have nodes of their own and they are connected to other words via edges of the same type that is used for content words. Observed in Enju.

- Second-class citizens – function words have nodes of their own but they are explicitly distinguished from content words. They are leaves in the graph and they can be viewed as mere features of the content word they are attached to. Observed in UCCA, UD (both basic and enhanced) and in Panini.

- Hidden nodes – function words are considered as existing on surface level only, and are hidden or removed during the transformation from surface to deep structure. Observed in MTT, FGD, Sequoia, DELPH-IN, and AMR.

Still, certain function words may be preserved for technical reasons (e.g., conjunctions to represent coordination, see Section 4.4).

- Edge labels – function words (especially prepositions) become edge labels. Not observed in frameworks included in the main comparison, but only in some more surfacy frameworks (in compression trees of Filippova and Strube, and in Microsoft logical forms, Section 3.12). In enhanced UD, prepositions and conjunctions can be copied to labels of edges incoming to "their" content word but at the same time they are still kept as separate nodes.

- PropBank and FrameNet-based approaches operate only on selected parts of a sentence corresponding to predicates and their arguments and ignore everything else (but recall that the complete sentence surface structure is already captured in the underlying surface-syntactic treebank, i.e., in the Penn Treebank or in the TIGER treebank, respectively). The related parts are typically content words or non-terminals. Note, however, that the PDTB, a project that we consider together with PropBank, uses a similar data structure for discourse relations: It treats a discourse connective (typically a function word) as a predicate, and the connected propositions (that are conveyed by individual clauses) as its arguments.

In the case of multiword function words, all tokens composing the multiword function words are usually handled in the same way (for instance, all are absent in the tectogrammatical representation in FGD).

Even if a representation hides function words, it usually keeps pronouns. While pronominal forms could be classified as function words, they are indispensable as arguments in propositions, members of coreference chains, and so forth.

Note that punctuation tokens may be treated either the same way as function words, or even as something less important (e.g., when function words are second-class citizens and punctuation is simply ignored).

4.2.3 Synonymy. Synonymy as, generally speaking, one type of asymmetry between form and meaning is observed with items of different complexity, from morphemes through words to sentences or even more complex units. Concerning syntactic issues, this topic is addressed in most of the surveyed approaches.

The approaches oscillate between two competing requirements. On the one hand, the sentence meaning representation is to be defined broadly enough in order to reflect the speaker's freedom to choose different surface-syntactic means in expressing a particular proposition (in other words, the speaker can choose from multiple paraphrases). On the other hand, the sentence meaning representation should be specific enough not to lose meaningful features contained in the surface structure. The latter one seems to be important when proceeding from form to function, for instance, when annotating particular sentences in a lexical resource.

A sort of minimal account is to assign an active and passive diathesis of the same predicate an identical deep representation (and possibly also a few more diathesis alternations, such as the dative shift alternation). While, for instance, the Sequoia treebank limits itself to this step, some other approaches proceed further to achieve a more abstract structure. The next step is the underspecificaton of grammatical and/or lexical information, as exemplified by the deep-syntactic representation in MTT. Another step is documented in AMR graphs, which abstract from words to concepts.

The amount of paraphrases that are treated as formally synonymous also depends on how far a given framework goes with abstracting from inflectional and derivational morphology (such as in the case of nominalizations, as discussed in Section 4.7).

4.2.4 Ordered and Unordered Representations. Because of the predominantly sequential nature of spoken language, a total (linear) order of individual *signifiers* is present in every utterance. And this holds even more for the written form, in which other communication channels such as intonation are usually suppressed.

It is also well known that languages differ quite a lot in how they use the inevitable presence of word order. For example, if a language does not use its word order for manifesting sentence constituents, then its word order may be used for conveying some other type of meaning, such as information structure; clearly, leaving word order completely unused would not be economic (and thus we believe that the term "free word order" is rather a misnomer).

When it comes to rendering a sentence's word order in a surface-syntactic formal representation, be it constituency- or dependency-oriented, then in a vast majority[25] of approaches the linear precedence of tokens in the original sentence is simply preserved (be the writing system oriented left-to-right as in Latin-based scripts, or right-to-left or top-down as in Arabic and Japanese, respectively). There is a rich body of literature dealing with word order in surface-syntactic formalisms, for example, from the following perspectives:

1. What empirical evidence on word-order phenomena can be found in syntactically annotated data, such as in the studies on non-projectivity[26] structures occurring in dependency treebanks (Kuhlmann and Nivre 2006), (Havelka 2007), or a typological view of Alzetta et al. (2018). Word-order phenomena that are non-trivial to handle and require, for example, using traces in constituency formalisms or allowing non-projectivities in dependency formalisms, sometimes also lead to introducing finer-grained categories such as mildly context-sensitive grammar (Joshi, Shanker, and Weir 1990) or mildly non-projective dependency grammar (Gómez-Rodríguez, Carroll, and Weir 2011).

2. What the impact is of various word-order-related requirements on parsing, in terms of complexity and efficiency. For instance, some graph-based models are able to produce non-projective dependencies natively (McDonald and Satta 2007), while special techniques had to be developed to adapt transition-based models for non-projective parsing (Kuhlmann and Nivre 2010). More recently, extensions from parsing into trees to parsing into more general graphs (which is supposed to be beneficial for downstream semantic processing) have been studied, too (Kuhlmann and Jonsson 2015).

When it comes to the order of nodes in deep-syntactic representations, the discussion in the literature seems to be much less structured. The dominating approach is that

---

25 There are exceptions, such as systems in which tokens are artificially permuted in a specific way in order to facilitate parsing, for instance, by reducing the length of long-distance dependencies (Bommasani 2019).

26 A dependency tree is projective if and only if an edge from node x to node y implies that x is an ancestor of all nodes located linearly between x and y.

the node order in deep-syntactic structures is not paid much attention to, but the structures are presented as ordered and it is assumed that the linear order can be induced from the linear order of corresponding surface strings. In fact, most deep-syntactic nodes are somehow anchored in the totally ordered sequence of sentence tokens, implicitly or explicitly. However, depending on a chosen deep-syntactic approach, one faces various structural asymmetries between surface and depth that require specific order-related decision-making, for instance, in the following cases of non-one-to-one correspondences:

- when multiple surface nodes collapse into a single deep node,

- if a deep node has no surface counterpart,

- in cases of ellipsis, if a single surface node has multiple deep counterparts.

In the opposite extreme case, no linear order of deep nodes is introduced, which is the case for AMR. This is done on purpose, as it allows reaching a higher level of abstraction: disclosing the original word order could lead to excluding some of the potential synonymous utterances, which is not desirable. This seems to correlate with other design decisions aimed at reaching high abstraction in these approaches (and thus, in a sense, at "hiding" what the original sentence exactly was), such as in the following cases discussed in more detail elsewhere in this article:

- representing deverbal derivatives and their base verb by the same label (Section 4.7.2),

- concealing an actual preposition if the semantic role label captures the meaning sufficiently (Section 4.2.2),

- or representing a set of co-referring expressions by a single node (since the information on pronominalization could also provide a clue about the original sentence ordering, because anaphora is more frequent than cataphora; Section 4.5).

To our knowledge, the only approach that uses deep-syntactic node order as a representational means for something different from (even though not completely unrelated to) the surface word order is FGD. In FGD, the node order represents Communicative Dynamism (Buráňová, Hajičová, and Sgall 2000). A total order is specified locally for a node and all its children using the notions of the topic-focus articulation, and the total order on the set of nodes of the whole tree is then only induced recursively; thus FGD's tectogrammatical trees are projective by definition.[27]

In addition, FGD in its earlier versions (Sgall, Hajičová, and Panevová 1986) considered ordering of conjuncts in a coordination structure as a special ordering, unrelated to the ordering induced from the topic-focus articulation of the dependency tree. In other words, coordination was considered a separate dimension of tectogrammatical trees (added to the dimensions of dependency subordination and of linear precedence). Even more interestingly, the number of such additional dimensions was supposed to grow in the case of nested coordinations. However, such a psychologically intricate algebraic

---

27 MTT represents the topic-focus articulation formally too, but on a different level of representation (on the semantic one) and not by node order.

model did not prove attractive for other researchers and is probably not used in any contemporary approach.

4.3 Valency and Semantic Roles

Different language units manifest clearly different combinatorial potentials, in the sense that they require different contexts (surroundings) to constitute acceptable utterances. Given that our study assumes the dependency-oriented syntactic paradigm with its head-dependent asymmetry, the combinatorial potentials of language units can be viewed from two perspectives: an *active* potential (what arguments a language unit requires) and a *passive* potential (to what other language units it can attach). We use exclusively the former one in the following text. Attempts at formalizing the latter (passive) perspective exist but none of the surveyed frameworks make use of them.

If we adopt the corpus-based methodology of contemporary linguistics, we can observe manifestations of the combinatory potentials in almost every utterance in a corpus. However, such positively observed instances are more or less where the "empirical truth" ends. If we want to have a (linguistically) interpretable model, we can proceed further only after introducing some more abstract notions, for which we need to adopt some assumptions first.

The assumptions adopted (though sometimes silently) in all or almost all deep-syntactic approaches under our study are the following:

- The combinatory potential of a language unit can be decomposed into a set of abstract "slots," which are saturated individually.

- The number of such slots is very low and stable for a given lexical unit, and thus it can be captured as lexicographic information about the lexical unit in a special dictionary.

- In a given utterance, it is possible for a linguist to recognize which argument fills which slot of a given language unit occurring in that utterance.

- The relation between the "dependent" (argument, participant, modifier) and the "governor" (predicate, head) can be labeled using a discrete (and relatively small, again) set of semantic role labels.

- In addition, the slot can be labeled with information on possible (surface) morphosyntactic forms of expressions that can saturate the slot. In other words, slots can contain both deep and surface information.

- To avoid redundancy, only such slots are to be described that are somehow specific (i.e., are specifically required or specifically permitted) for the given lexical unit. It makes no sense to describe such slots that can appear with every language unit.

- Combinatorial potentials of different language expressions are clearly not independent. For example, different inflected word forms from a conjugation paradigm of a given verb are typically related in a quite systematic way, and there is a strong dependency also when it comes to morphological derivations. Thus combinatory potentials are not described for each and every graphemically different expression separately, but

rather for whole "clouds" of morphologically (inflectionally or derivationally) related expressions.

Now we are finished with the underlying theory-neutral intuition. More detailed operational criteria are chosen for each of these assumptions in each formal framework. The comparison of inventories of semantic roles recalls a quotation from Dowty (1991, page 547):

> There is perhaps no concept in modern syntactic and semantic theory which is so often involved in so wide a range of contexts, but on which there is so little agreement, as [semantic role] . . . .

Almost thirty years later the situation does not seem to be any better; the fact that various semantic role inventories have been used in corpus annotation did not lead to substantial convergence, though one can find recurring patterns. We believe that the diversity is worth studying separately for arguments and for adjuncts.

As for argument semantic roles,[28] at first glance we notice highly different numbers of distinguished argument roles:

1. only very few indexing labels such as Arg0,

2. more weakly descriptive labels such as Actor,

3. hundreds of strongly descriptive labels such as Buyer.

Actually the difference is not only in granularity, but is a more principled one. The first approach with a very low number of labels is used in the following framework families:

- Frameworks that refer back to Dowty's proto-roles proto-agent and proto-patient (Dowty 1991), such as PropBank, in which labels `ARG0` to `ARG5` are used (cf. also Enju and DELPH-IN).

- MTT (Kahane 2003), in which Roman numerals (`I`, `II`) are assigned to arguments.

- FGD, in which labels `ACT`, `PAT`, `ADDR`, `ORIG`, and `EFF` are used; according to the shifting principle of Panevová (1974–1975), the `ACT` slot is present when there is at least one argument, the `PAT` slot is present if there are at least two arguments and it is the less volitional one, and more "semantics" is considered only if there are three or more arguments. A similar shifting principle applies also to the karaka relations of Panini.

- In UCCA, 12 categories such as `P` – Process, `A` – Participant, and `D` – Adverbial are distinguished. They are very coarse-grained and do not distinguish, for instance, roles of individual predicate's participants from each other.

The common denominator of such approaches is that they do not attempt to find meaningful partitioning of all possible argument roles played in all possible situations;

---

28 Sequoia and UD, whose relation type inventories are clearly surface-oriented, are not included in this particular analysis.

Table 5
Argument semantic roles assigned within the sentence meaning representation of the sentences *The boy opened the lock*, *The key opened the lock*, and *The lock opened* in the frameworks under survey.

	The boy	opened	the lock.	The key	opened	the lock.	The lock	opened.
Panini	k1		k2	k1		k2	k1	
MTT	I		II	I		II	I	
FGD	ACT		PAT	ACT		PAT	ACT	
PropBank	ARG0		ARG1	ARG2		ARG1	ARG1	
FrameNet	Agent	Closure	C_portal	Instrument	Closure	C_portal	C_portal	Closure
ENJU	ARG1		ARG2	ARG1		ARG2	ARG1	
DELPH-IN	ARG1		ARG2	ARG1		ARG2	ARG1	
Sequoia	suj:suj		obj:obj	suj:suj		obj:obj	suj:suj	
AMR	ARG0		ARG1	ARG2		ARG1	ARG1	
UCCA	A	P	A	A	P	A	A	P
EUD	nsubj		obj	nsubj		obj	nsubj	

instead, the labels serve almost exclusively for indexing purposes: they only identify which frame slot is being filled, and the cognitive meaning of the role is left up to the lexical semantics of the governor. What we find interesting is that—at least to our knowledge—the indexing approaches mentioned above developed basically without being influenced by each other.

The most prominent representative of the opposite end of the granularity scale is clearly FrameNet. In this approach (applied in SALSA, for instance), the labels are extremely detailed, but there is some reuse: The sets of labels are shared, for example, across verbs describing the same situation (such as buying and selling). See Table 5 for a basic comparison of core semantic labels across all the frameworks surveyed.

When it comes to adjunct semantic roles, there is also a range of different partitionings introduced in the individual frameworks. For the purpose of illustration, we draw the granularity scale as follows:

- Raw granularity: MTT is an extreme with only one adjunct semantic label (`ATTR`);

- Medium granularity: 11 types of modifiers distinguished in PropBank, such as `ARGM-LOC`, `ARGM-TIME`;

- High granularity: PDT with about 60 adjunct roles (called functors), or AMR distinguishing around 90 adjunct roles.

In some cases, a two-level system of adjunct roles is introduced, in which the second level can possibly be used for capturing finer-grained distinctions.

FGD introduces so-called subfunctors that distinguish, for instance, `LOC.below` from `LOC.above` (actually this finer-grained division basically reflects prepositions or subordinating conjunctions used in the sentence).

In AMR, temporal and spatial prepositions are kept if they carry additional information, such as time-after.

4.4 Paratactic Structures

As discussed by Popel et al. (2013) in their survey of coordination representations in dependency treebanking, paratactic syntactic structures such as coordination and apposition are notoriously difficult to be represented by dependency formalisms. The

reason is that the nature of paratactic structures is symmetric (two or more conjuncts play the same role), as opposed to the head-modifier asymmetry of dependencies. The dominant solution in treebank design is to introduce artificial rules for the encoding of coordination structures within dependency trees using the same means that express dependencies, that is, by using edges and by labeling of nodes or edges.

Three major families of representations of paratactic structures are distinguished in the referred survey: In the Prague family (instantiated, e.g., in the PDT), all the conjuncts are siblings governed by one of the conjunctions (or a punctuation fulfilling its role); in the Moscow family (instantiated in MTT), the conjuncts form a chain where each node in the chain depends on the previous (or following) node; in the Stanford family (instantiated in UD), the conjuncts are siblings except for the first (or last) conjunct, which is the head. Several other dimensions of variability are identified and illustrated on treebanks for 26 languages (Popel et al. 2013). Basic possible representations are shown in Table 6. The said survey focuses on surface-syntactic representations where the required structure is a rooted tree. There are more options in general graphs. For example, one type of edges can connect conjuncts and another type can link each conjunct separately to a shared governor or dependent.

- The Paninian treebanks, FGD, Enju, the Elementary Dependency Structures of DELPH-IN, and AMR use generalized Prague-style structures to capture coordination: The head node either directly corresponds to a surface conjunction word, or it represents an abstract joining concept (especially AMR).

- MTT, the DM graphs of DELPH-IN, and Sequoia use Moscow-style structures.

- Enhanced UDs use Stanford-style structures combined with dependency propagation across conjuncts.

Table 6
Possible representations of the coordination structure "dogs, cats and rats" in different dependency approaches (adopted from Popel et al. 2013).

Family	Prague family	Moscow family	Stanford family
Choice of head			
Head on left			
Head on right			

- In UCCA, conjuncts are grouped as parallel scenes under one non-terminal, and the non-terminal unit also covers the linking conjunction (but the conjunction is not the head of the unit).

- In PropBank, dependencies are propagated across conjuncts without explicitly showing coordination. An argument shared by coordinate predicates is linked from each of the predicates separately; in contrast, coordinate arguments enter the deep structure as a single constituent. Similarly in SALSA, coordination is only annotated in the underlying surface-syntactic structure and if it fills a slot in a frame, the slot points to the constituent that covers all conjuncts.

4.5 Deletions and Coreference

Paninian treebanks have empty nodes for deleted predicates. The Paninian grammar defines rules for certain instances of grammatical coreference (called "karaka sharing"; Bharati, Chaitanya, and Sangal 2006, §5.6.2); however, treebanks based on Panini do not annotate the shared arguments explicitly. Other types of ellipsis and coreference are not annotated either.

Various corpora that refer to MTT take different approaches to ellipsis and coreference. SynTagRus has empty nodes (called "phantom"[29] nodes) for deleted predicates in gapping constructions. The deep-syntactic layer of the AnCora-UPF treebank has node copies for shared arguments in control verb constructions (cf. the two copies of the word *persona* 'person' in Figure 2). The two deep nodes are connected by a coreference edge. Dropped subject pronouns are reconstructed as empty nodes and may enter coreference relations, too. Kahane (2003) shows in his theoretical overview that a deep-syntactic tree where all pronouns are expanded to the lexemes they represent, and where additional links denote the coreference, is equivalent to a DAG (Figure 20); he further argues that a DAG is easier to work with when the coreferential noun has dependents.

In the tectogrammatical layer of FGD, empty nodes are used for all missing, valency-licensed arguments, including pro-drop subjects. In the case of grammatical coreference, gapping, and so on, an empty node is generated first, then it is linked to its antecedent using a coreference edge (Figure 3). Prague treebanks also annotate textual coreference (including cross-sentence edges) and bridging anaphora.

In Sequoia, grammatical coreference is solved by linking the shared arguments directly from all predicates that share them. There are no empty nodes. The same holds for PropBank, DELPH-IN (Figure 9), and Enju PAS (Figure 7; in the last two, arguments can be shared even between content and auxiliary verbs). However, the English PropBank is built over the Penn Treebank, which, despite being a rather surface-syntactic representation, has empty nodes (called *traces*) to account for grammaticalized ellipsis and coreference (Figure 5).

In AMR, inner nodes representing abstract predicates are connected with the inner node representing their shared abstract argument. Inner nodes are connected with nodes of concrete words via special "instance" edges. The nodes of concrete words are leaves.

In UCCA, coreference annotation was added as a separate (additional) layer on top of the foundational layer in a recent pilot experiment (Prange, Schneider, and Abend

---

29 The actual spelling is "fantom."
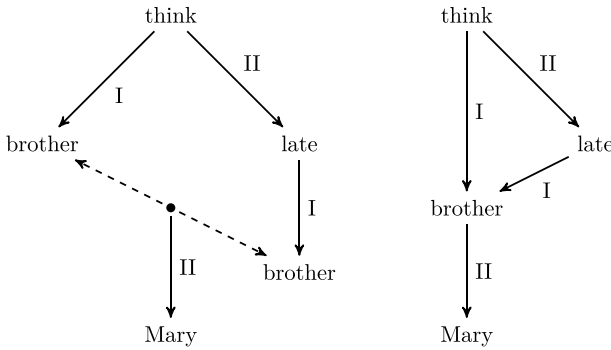
Figure 20
Deep-syntactic tree vs. DAG in MTT for the sentence *Mary's brother thinks he is late* (adapted from Kahane 2003). The left-hand side illustrates that a tree would need two nodes corresponding to *brother* and that it is then unclear where the modifier *Mary* should be attached. This problem disappears if the two nodes for *brother* are merged and the structure becomes the DAG on the right-hand side.

2019). The coreference relations are assigned with units delimited at the foundational layer. UCCA also has empty ("implicit") nodes to represent elided participants.

Enhanced UD provides empty (copied) nodes for deleted predicates so that their arguments and adjuncts have a reasonable attachment option (Figure 15). Other than that, empty nodes are not used. Grammatical coreference in control verb constructions is solved by connecting the shared argument directly with both verbs. Similarly, the nominal modified by a relative clause is simultaneously attached as a dependent node within the clause according to the role it plays there. The relative pronoun is connected with its antecedent via a coreference edge labeled `ref` (Figure 17). Other instances of ellipsis or coreference are not explicitly annotated. If there are orphaned dependents, for instance, within a noun phrase, one of them is simply promoted to the position of the missing head noun.

SALSA does not seem to include any rules for coreference or ellipsis, although specific types of ellipsis occurring in coordination constructions are captured by so-called secondary edges in the TIGER treebank (Harbusch and Kempen 2007).

4.6 Discourse Relations

The term "discourse relations" refers to semantic relations between propositions, which are conveyed by clauses based on individual predicate–argument structures. In parallel to predicate–argument structures, propositions are described as discourse arguments that are related either explicitly by a discourse connective (a subordinating or coordinating conjunction, or a discourse adverbial like *instead*), or implicitly.

Discourse relations between propositions that are expressed by individual clauses within a single sentence are handled as a part of the sentence meaning representation across the reviewed approaches. While subordinating conjunctions fall into the class of function words and predicates of subordinated clauses are represented as arguments of the governing predicate (see Section 4.2.2), coordinating conjunctions are often part of the sentence meaning representation (similarly to content words) and different accounts of coordination are documented in Section 4.4.

Unlike subordination and coordination as intrasentential relations, relations between propositions that are separated into different sentences (inter-sentential relations) are omitted in most approaches or, if considered, they are annotated at a separate layer.

- PDTB (Miltsakaki et al. 2004b; Prasad et al. 2008) is a project related to Penn Treebank and PropBank (cf. Section 3.4). It annotates the Wall Street Journal Section of the Penn Treebank with discourse relations. If an explicit discourse connective is found in the sentence or between two sentences, it is assigned a sense tag. If no discourse connective is present, a connective expression is added into the structure (being encoded as a lexical item or with a special label). Discourse relations are assigned between clauses in a sentence and between each successive pair of sentences within paragraphs.

- In Prague Dependency Treebank (Section 3.3), annotation of discourse relations was added on top of existing annotation layers (morphological, surface-syntacic, and deep-syntactic annotation). The discourse annotation was released under the title Prague Discourse Treebank (Poláková et al. 2012; Rysová et al. 2016). In addition to discourse relations based on both explicit and implicit discourse connectives (as described with the PDTB project), extended textual coreference, bridging relations, annotation of elliptical constructions, apposition, and parentheses, many of them annotated already at the deep-syntactic layer, are considered a part of the discourse annotation.

4.7 Partitioning the Lexical Space

It is a generally accepted assumption in linguistics that in many languages some words can occur in two or more different forms, to mark distinctions such as tense or number. The process of modifying the word form in order to express such grammatical categories is called **inflection**. Word forms can be grouped together and each group is represented by a canonically selected word form, usually called a lemma. Inflection is distinguished from derivation, in which additional meanings are also added to a base word (by adding morphemes), the result of which, however, is considered a different word, not just a different word form.[30]

To a certain extent, it is a matter of linguistic convention to distinguish inflection from derivation (such as in the case of aspectual verb counterparts in Czech), and it is also a matter of linguistic (especially lexicographic) tradition to choose a lemma within a cluster of inflectionally related word forms. These fuzzy boundaries bring one extra dimension of diversity into contemporary language data resources.

Another dimension of diversity, this time more specific for "deep" approaches, is related to how a given approach compensates for the information removed from a word form during lemmatization. Given a deep-syntactic graph, some pieces of information become really redundant (such as those verbal features that are only imposed by subject-verb agreement), while some other pieces of information are still semantically indispensable (such as number with nouns).

---

30 It should be emphasized that a word form used in a sentence might be composed of more tokens, such as the future tense in English.

4.7.1 Inflectional Morphology. We can observe the following range of approaches to how inflection is tackled in the frameworks under our survey, from minimalistic to theoretically founded ones:

1. Nodes are simply labeled with original word forms, without any attempt at lemmatization.

2. Word forms as well as lemmas, and possibly also detailed POS tags (with values of all inflectional categories) are kept in nodes and no abstract model is used to remove the apparent redundancy.

3. Only selected morphological categories are represented, while all other inflectional categories are lost after lemmatization.

4. The most semantically oriented approaches were developed in FGD and in MTT, as described in Žabokrtský (2005, page 553): "each lexeme is associated with appropriate semantically full grammemes (grammatemes in FGD terminology); grammemes imposed only by government and agreement are excluded." Thus out of the inflectional categories removed by lemmatization, only the semantically indispensable ones (such as number with nouns, but not case, or tense with verbs, but not number) get their deep counterpart. A similar approach is enabled also in DELPHIN-MRS, in which, however, such extra attributes for capturing morphologically expressed meanings seem to be considered rather an optional and less elaborated extension.[31]

4.7.2 Derivational Morphology. With a bit of oversimplification, derivation can be considered as prolonged lemmatization (in the sense that it merges clusters of word forms induced by lemmatization into even bigger clusters), and again, we can observe a gradual range of solutions:

1. Most approaches do not go beyond lemmatization (if they consider lemmatization at all).

2. The middle way: FGD (in PDT) converts some of the most productive derivations back to base words, for example, in the case of possessive adjectives, which are represented by lemmas of their base nouns, or in the case of deadjectival adverbs, which are represented by their base adjectives in tectogrammatical trees. DELPH-IN handles selected most productive types of morphological derivation in the same way. AMR represents many deverbal derivatives (such as nominalizations) by their base verbs, and associates them with selected PropBank frames of these verbs; in addition, AMR removes derivational prefixes for negation from lemmas and adds a negative polarity marker instead.

3. MTT goes probably farthest, as it systematically captures derivative relations among lexemes using the notion of paradigmatic lexical function,

---

31 `http://moin.delph-in.net/EdsTop`.

which allows replacing a derived lexeme using a lexical function applied on the base lexeme in the deep-syntactic representations. For instance, the word *decision* would be represented by a deep-syntactic node labeled with S_0*(decide)* (Milićević 2006).

It should be mentioned that the boundary between inflection and derivation is not dependent only on a chosen linguistic tradition but also on the language: For instance, whereas negation of nouns in Czech is pretty regular and can be thus easily considered as an inflectional category, the nature of negated nouns in English is rather derivational.

4.8 Complex Word Forms

Many languages express certain inflectional categories by adding auxiliary words, such as in the case of English future tense (complex verb phrases).[32] Similarly to the case of "single-token" inflection, the auxiliary tokens can be removed during the transfer from surface-syntactic to deep-syntactic representation, and—if semantically indispensable—represented by a specific grammeme value in MTT (or grammateme in FGD). In some cases, a grammeme value is an abstraction over typologically different surface morphological means: for example, superlatives in English are formed either by adding a suffix morpheme *-est*, or by adding the auxiliary *most*.

In some frameworks, light verb constructions (which are another example of expressions resulting from a grammaticalized combination of several words) receive particular attention. Typically, a light verb construction consists of a verb that bears very general lexical meaning, and of a noun that carries the main lexical meaning of the entire phrase, such as in the case of *to put emphasis* (instead of *to emphasize*) and *to give a kiss* (instead of *to kiss*). In FGD, the verb is still marked as the main predicate, but the noun receives a special semantic role (analogously, dependent parts of a multiword idiom expression are marked with another special role). In PropBank and AMR, it is the whole light verb construction what is assigned a predicate–argument frame. UD currently allows both approaches: In many languages including English, it attaches the nominal part of a light verb construction simply as a direct object (`obj`). However, languages where light verb constructions are dominant, such as Persian or Hindi, can opt for using a special edge type, which marks the whole construction as one lexical unit (`compound:lvc`).

5. Summary: Can We Converge?

Having surveyed a number of diverse approaches to representing sentence meaning of natural languages, we can now turn to our initial question: Is there or can there be any convergence that would eventually result in one unified approach that is not only applicable, but also applied to dozens of different languages?

Many frameworks discussed in this article have been applied to multiple languages; virtually all of them at least assume that they can be applied to any language. On the other hand, none of them has reached as far and wide as Universal Dependencies did on the surface-syntactic level. What should or could be done for this to actually happen? In the community around UD, there seems to be demand to "go deeper." The first step is the enhanced UD representation, but a real semantic representation would probably

---

32 We prefer to avoid the term "multiword entities," as it often subsumes very heterogeneous types of expressions such as named entities or idioms.

have to be a new project, backwards-compatible with UD and built on top of it.[33] It seems inevitable that it will provide annotated data for fewer languages than UD itself, simply because it requires more work, and even within UD some languages have only tiny text samples due to lack of available manpower. Therefore, it is important to identify the most desirable aspects of a sentence meaning representation. By recommending these aspects as the core minimum, and making the other aspects optional, the entrance barrier could be sufficiently lowered.

We are not going to strictly define what this minimum should be—that should come out of a broader discussion to which we hope to contribute. However, we want to point out in this summary what the tendencies and recurring solutions are that we found. It is striking how many similar features one can observe across such a diverse set of intellectual works, despite their being created by mostly disjoint groups of people, often distant both in time and space.

Looking back at the individual items in Section 4, we can make a number of generalizations about covered phenomena and prevailing approaches. At the same time, these generalizations can serve as a kind of recommendation for what should be considered as the minimal core and what would be better left optional:

- Deep syntax can be well represented by a directed graph. Requiring a tree would be too restrictive; a DAG may be enough for most phenomena but not all (recall that Enhanced UD contain occasional cycles).

- Only content words are normally important in the graph. Still, the graph can be trivially connected and span all surface words if function words are attached via a special edge to a suitable content node. It also follows that at least partial node order can be deduced from the order of the surface words.

- Empty nodes are useful means of representing elided material. However, the exact extent of their usage is a matter of further discussion. Some instances of copied nodes in some frameworks can actually be replaced by redirecting edges to the original node.

- Deep representations often exist together with corresponding surface representations. In case of "deep UD," the natural surface counterpart would be the basic UD tree.

- Normalization of diathesis is the common minimum that is done with predicate–argument structure.

- Most frameworks also solve at least some instances of grammatical coreference such as control verbs. This way the predicate is connected to all arguments that are overtly represented on surface. Reconstruction of dropped, valency-licensed pronouns is less common, although arguably useful.

- Some frameworks are accompanied by large lexical resources with predicates, their valency frames, detailed semantic roles of the arguments,

---

33 During the preparation of the present article, a first version of Deep UD corpora was released by Droganova and Zeman (2019). However, this version does not go beyond simple heuristics that take an Enhanced UD graph and normalize diathesis.

etc. Such lexicons are invaluable but also extremely costly, hence not likely to become available for many languages. A "deep UD" framework should be able to incorporate them but it should also be able to exist without them.

- Textual coreference, bridging relations, and any discourse relations that cross sentence boundaries are relatively rare. On the other hand, they do not require significantly different data structures; the only technical problem is finding a file format that allows links between sentences.

- Lexical synonymy, normalization of derivational morphology, complex word forms, and gramm(at)emes are also dealt with less frequently. They could be optional.

- Some of the surveyed frameworks are closer to syntax (e.g., Enhanced UD or PDT); others are more abstract (e.g., AMR). It may not be tractable to define one all-inclusive scheme; instead, we may end up with multiple "deep" layers, similarly to the deep-syntactic vs. semantic layer of MTT. If a multilingual resource is built around the UD treebanks, then it seems natural to start with the layer that is less abstract and closer to syntax, and then to enrich the scheme with more semantic components gradually. In the long term, we could go further and approach phenomena behind the scope of our survey, such as temporal and spatial semantics, logical elements such as quantification and entailment, rhetorical structures, and so forth—some of them reviewed by Abend and Rappoport (2017).

It has yet to be seen what exactly is selected for pilot annotation on top of existing UD treebanks. We believe that the present survey, first of its kind, will contribute to shaping a new layer of extensive multilingual annotation.

References

Abeillé, Anne and Nicolas Barrier. 2004. Enriching a French treebank. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, pages 2233–2236, Paris.

Abend, Omri and Ari Rappoport. 2013. Universal Conceptual Cognitive Annotation (UCCA). In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, pages 228–238, Sofia.

Abend, Omri and Ari Rappoport. 2017. The state of the art in semantic representation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017). Volume 1: Long Papers*, pages 77–89, Vancouver.

Ágel, Vilmos, Ludwig M. Eichinger, Hans-Werner Eroms, Peter Hellwig, Hans Jürgen Heringer, and Henning Lobin. 2003, 2006. *Dependency and Valency*. An International Handbook of Contemporary Research. Volumes 1, 2, De Gruyter, Berlin.

Akbik, Alan, Xinyu Guan, and Yunyao Li. 2016. Multilingual aliasing for auto-generating proposition banks. In

Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016). Technical Papers, pages 3466-3474, Osaka.

Alzetta, Chiara, Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. 2018. Universal dependencies and quantitative typological trends. A case study on word order. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*, pages 4540–4549, Paris.

Apresjan, Juri D., Igor Boguslavsky, Boris Iomdin, Leonid L. Iomdin, Andrei Sannikov, and Victor G. Sizov. 2006. A syntactically and semantically tagged corpus of Russian: State of the art and prospects. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, pages 1378–1381, Paris.

Babko-Malaya, Olga, Martha Palmer, Nianwen Xue, Aravind Joshi, and Seth Kulick. 2004. Proposition Bank II: Delving deeper. In *Proceedings of the Workshop Frontiers in Corpus Annotation at HLT-NAACL 2004*, pages 17–23, Boston, MA.

Baker, Collin F., Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL 1998) and 17th International Conference on Computational Linguistics (COLING 1998). Volume 1*, pages 86–90, Montreal.

Ballesteros, Miguel, Bernd Bohnet, Simon Mille, and Leo Wanner. 2014. Deep-syntactic parsing. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1402–1413, Dublin.

Banarescu, Laura, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186.

Begum, Rafiya, Samar Husain, Arun Dhwaj, Dipti Misra Sharma, Lakshmi Bai, and Rajeev Sangal. 2008. Dependency annotation scheme for Indian languages. In *Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP 2008)*, pages 721–726, Hyderabad.

Bejček, Eduard, Jan Hajič, Jarmila Panevová, Jiří Mírovský, Johanka Spoustová, Jan Štěpánek, Pavel Straňák, Pavel Šidák, Pavlína Vimmrová, Eva Šťastná, Magda Ševčíková, Lenka Smejkalová, Petr Homola, Jan Popelka, Markéta Lopatková, Lucie Hrabalová, Natalia Klyueva, and Zdeněk Žabokrtský. 2011. Prague Dependency Treebank 2.5. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Bejček, Eduard, Eva Hajičová, Jan Hajič, Pavlína Jínová, Václava Kettnerová, Veronika Kolářová, Marie Mikulová, Jiří Mírovský, Anna Nedoluzhko, Jarmila Panevová, Lucie Poláková, Magda Ševčíková, Jan Štěpánek, and Šárka Zikánová. 2013. Prague Dependency Treebank 3.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Bharati, Akshar, Vineet Chaitanya, and Rajeev Sangal. 2006. *Natural Language Processing: A Paninian Perspective*, Prentice-Hall of India, New Delhi, India.

Boguslavsky, Igor. 2014. SynTagRus—a deeply annotated corpus of Russian. In Peter Blumenthal, Iva Novakova and Dirk Siepmann, editors, *Les émotions dans le discours [Emotions in Discourse]* Frankfurt am Main: Peter Lang, pages 367–380.

Boguslavsky, Igor. 2017. Semantic Descriptions for a text understanding system. In *Computational Linguistics and Intellectual Technologies. Papers from the Annual International Conference "Dialogue"(2017)*, pages 14–28.

Bommasani, Rishi. 2019. Long-distance dependencies don't have to be long: Simplifying through provably (approximately) optimal permutations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 89–99.

Bos, Johan, Valerio Basile, Kilian Evang, Noortje J. Venhuizen, and Johannes Bjerva. 2017. The Groningen Meaning Bank. In *Handbook of Linguistic Annotation*. Springer, pages 463–496.

Brants, Sabine, Stefanie Dipper, Silvia Hansen, Wolfgang Lezius, and George Smith. 2002. The TIGER treebank. In *Proceedings of the First Workshop on Treebanks and Linguistic Theories*, pages 24–41, Sozopol.

Buráňová, Eva, Eva Hajičová, and Petr Sgall. 2000. Tagging of very large corpora: Topic-focus articulation. In *COLING 2000 Volume 1: The 18th International Conference on Computational Linguistics*, pages 139–144.

Burchardt, Aljoscha, Katrin Erk, Anette Frank, Andrea Kowalski, Sebastian Padó, and Manfred Pinkal. 2006. The SALSA corpus: A German corpus resource for lexical semantics. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, pages 969–974. European Language Resources Association, Paris.

Candito, Marie, Mathieu Constant, Carlos Ramisch, Agata Savary, Yannick Parmentier, Caroline Pasquer, and Jean-Yves Antoine. 2017. Annotation d'expressions polylexicales verbales en français. In *TALN 2017-24e conférence sur le Traitement Automatique des Langues Naturelles*, pages 1–9.

Candito, Marie and Guy Perrier. 2016. Guide d'annotation en dépendances profondes pour le français. `https://hal.inria.fr/hal-01249907`.

Candito, Marie, Guy Perrier, Bruno Guillaume, Corentin Ribeyre, Karën Fort, Djamé Seddah, and Éric Villemonte de La Clergerie. 2014. Deep syntax annotation of the Sequoia French treebank. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, pages 2298–2305, European Language Resources Association, Paris.

Candito, Marie and Djamé Seddah. 2012. Le corpus Sequoia: Annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical. In *TALN 2012-19e conférence sur le Traitement Automatique des Langues Naturelles*, pages 321–334, Grenoble.

Clark, Herbert H. 1975. Bridging. In *Theoretical Issues in Natural Language Processing*, pages 169–174, New York.

Copestake, Ann, Dan Flickinger, Carl Pollard, and Ivan A Sag. 2005. Minimal recursion semantics: An introduction. *Research on Language and Computation*, 3(2-3):281–332.

Daneš, František. 1994, The sentence-pattern model of syntax, Luelsdorff, P. A., editor, *The Prague School of Structural and Functional Linguistics*. John Benjamins Publishing Company, Amsterdam – Philadelphia, pages 197–221.

Das, Dipanjan, Desai Chen, André F. T. Martins, Nathan Schneider, and Noah A. Smith. 2014. Frame-Semantic Parsing. *Computational Linguistics*, 40:9–56.

De Marneffe, Marie Catherine and Christopher D. Manning. 2008. The Stanford typed dependencies representation. In *Proceedings of the Workshop on Cross-Framework and Cross-Domain Parser Evaluation at COLING 2008*, pages 1–8.

de Saussure, Ferdinand. 1916. *Cours de linguistique générale*, Payot, Paris.

de Saussure, Ferdinand. 1978. *Cours de linguistique générale*, Payot, Paris.

Donatelli, Lucia, Michael Regan, William Croft, and Nathan Schneider. 2018. Annotation of tense and aspect semantics for sentential AMR. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 96–108.

Dorr, Bonnie J. 1997. Large-scale acquisition of LCS-based lexicons for foreign language tutoring. In *Proceedings of the 5th Conference on Applied Natural Language Processing (ANLC 1997)*, pages 139–146, Stroudsburg.

Dowty, David. 1991. Thematic proto-roles and argument selection. *Language*, 67(3):547–619.

Droganova, Kira and Daniel Zeman. 2017. Elliptic constructions: spotting patterns in UD treebanks. NoDaLiDa 2017 Workshop on Universal Dependencies, pages 48–57, Gteborgs universitet, Göteborg.

Droganova, Kira and Daniel Zeman. 2019. Towards deep Universal Dependencies. In *Proceedings of the 5th International Conference on Dependency Linguistics (DepLing 2019 – Syntaxfest 2019)*, pages 144–152, Paris.

Duran, M. S. and S. M Aluísio. 2012. Propbank-Br: A Brazilian Treebank annotated with semantic role labels. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012)*, pages 1862–1867, European Language Resources Association, Paris.

Ellsworth, Michael, Katrin Erk, Paul Kingsbury, and Sebastian Padó. 2004. PropBank, SALSA, and FrameNet: How design determines product. In *Proceedings of the LREC 2004 Workshop on Building Lexical Resources from Semantically Annotated Corpora*, pages 17–23, Lisbon.

Erk, Katrin, Andrea Kowalski, Sebastian Padó, and Manfred Pinkal. 2003. Towards

a resource for lexical semantics: A large German corpus with extensive semantic annotation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2003)*, pages 537–544.

Erk, Katrin and Sebastian Padó. 2004. A Powerful and Versatile XML Format for Representing Role-semantic Annotation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, pages 799–802, European Language Resources Association, Paris.

Estarrona, Ainara, Izaskun Aldezabal, and Arantza Díaz de Ilarraza. 2018. How the corpus-based Basque Verb Index lexicon was built. *Language Resources and Evaluation*, 52:1–23.

Filippova, Katja and Michael Strube. 2008. Dependency tree based sentence compression. In *Proceedings of the 5th International Natural Language Generation Conference*, pages 25–32, Stroudsburg, PA.

Fillmore, Charles J. 1968. The case for case, E. Bach and R. Harms, editors, *Universals in Linguistic Theory*. New York, pages 1–88.

Fillmore, Charles J. 1976. Frame semantics and the nature of language. *Annals of the New York Academy of Sciences*, 280(1):20–32.

Flickinger, Dan, Yi Zhang, and Valia Kordoni. 2012. DeepBank. A dynamically annotated treebank of the Wall Street Journal. In *Proceedings of the 11th International Workshop on Treebanks and Linguistic Theories*, pages 85–96.

Gómez-Rodríguez, Carlos, John Carroll, and David Weir. 2011. Dependency parsing schemata and mildly non-projective dependency parsing. *Computational Linguistics*, 37(3):541–586.

Hahm, Younggyun, Jiseong Kim, Sunggoo Kwon, and Key-Sun Choi. 2018. Semi-automatic Korean FrameNet annotation over KAIST treebank. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*, pages 83–87, European Language Resources Association, Paris.

Hajič, Jan. 1998. Building a syntactically annotated corpus: The Prague Dependency Treebank. In Eva Hajičová, editor, *Issues of Valency and Meaning*. Prague: Karolinum, pages 106–132.

Hajič, Jan, Eva Hajičová, Jaroslava Hlaváčová, Václav Klimeš, Jiří Mírovský, Petr Pajas, Jan Štěpánek, Barbora

Vidová Hladká, and Zdeněk Žabokrtský. 2006. *PDT 2.0 – Guide*, ÚFAL MFF UK, Praha, Czechia.

Hajič, Jan, Eva Hajičová, Marie Mikulová, and Jiří Mírovský. 2017, Prague Dependency Treebank, In *Handbook of Linguistic Annotation*. Springer, pages 555–594.

Hajič, Jan, Eva Hajičová, Marie Mikulová, Jiří Mírovský, Jarmila Panevová, and Daniel Zeman. 2015. Deletions and node reconstructions in a dependency-based multilevel annotation scheme. In *Proceedings of CICLING 2015*, pages 17–31, Springer, Cham.

Hajič, Jan, Jarmila Panevová, Eva Hajičová, Petr Sgall, Petr Pajas, Jan Štěpánek, Jiří Havelka, Marie Mikulová, Zdeněk Žabokrtský, Magda Ševčíková-Razímová, and Zdeňka Urešová. 2006. Prague Dependency Treebank 2.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Hajičová, Eva and Ivona Kučerová. 2002. Argument/Valency Structure in PropBank, LCS Database and Prague Dependency Treebank: A comparative pilot study. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC 2002)*, pages 846–851, European Language Resources Association, Paris.

Harbusch, Karin and Gerard Kempen. 2007. Clausal coordinate ellipsis in German: The TIGER treebank as a source of evidence. In *Proceedings of the 16th Nordic Conference of Computational Linguistics (NODÁLIDA 2007)*, pages 81–88.

Hashimoto, Kazuma, Pontus Stenetorp, Makoto Miwa, and Yoshimasa Tsuruoka. 2014. Jointly learning word representations and composition functions using predicate–argument structures. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1544–1555.

Havelka, Jiří. 2007. Beyond projectivity: Multilingual evaluation of constraints and measures on non-projective structures. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 608–615.

Haverinen, Katri, Jenna Kanerva, Samuel Kohonen, Anna Missilä, Stina Ojala, Timo Viljanen, Veronika Laippala, and Filip

Ginter. 2015. The Finnish Proposition Bank. *Language Resources and Evaluation*, 49(4):907–926.

Husain, Samar, Prashanth Mannem, Bharat Ambati, and Phani Gadde. 2010. The ICON-2010 tools contest on Indian language dependency parsing. In *Proceedings of ICON-2010 Tools Contest on Indian Language Dependency Parsing*, pages 1–8, Kharagpur.

Ide, Nancy and James Pustejovsky. 2017. *Handbook of Linguistic Annotation*, Springer.

Ivanova, Angelina, 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.

Jensen, Karen. 1993. PEGASUS: Deriving argument structures after syntax. In *Natural Language Processing: The PLNLP Approach*, Springer, pages 203–214.

Johnson, Christopher R., Charles J. Fillmore, Miriam R. L. Petruck, Collin F. Baker, Michael Ellsworth, Josef Ruppenhofer, and Esther J. Wood. 2002. FrameNet: Theory and practice. `https://nats-www.informatik.uni-hamburg.de/pub/CDG/FrameNet/book.pdf`.

Johnson, Justin, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. 2015. Image retrieval using scene graphs. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3668–3678.

Joshi, Aravind K., K. Vijay Shanker, and David Weir. 1990. The convergence of mildly context-sensitive grammar formalisms. *Technical Reports (CIS)*.

Kahane, Sylvain. 2003. The Meaning-Text Theory. In *Dependency and Valency. An International Handbook of Contemporary Research.* Volume 1, De Gruyter, Berlin, pages 546–570.

Kingsbury, Paul and Martha Palmer. 2002. From TreeBank to PropBank. In *Proceedings of the 3th International Conference on Language Resources and Evaluation (LREC 2002)*, pages 1989–1993, European Language Resources Association, Paris.

Kiparsky, Paul. 1982. *Some Theoretical Problems in Panini's Grammar*. Bhandarkar Oriental Research Institute, Pune, India.

Kipper, Karin, Hoa Trang Dang, and Martha Palmer. 2000. Class-based construction of a verb lexicon. In *Proceedings of the 17th National Conference on Artificial Intelligence and 12th Conference on Innovative Applications of Artificial Intelligence*, pages 691–696.

Kipper, Karin, Martha Palmer, and Owen Rambow. 2002. Extending PropBank with VerbNet semantic predicates. In *Proceedings of the 5th Conference of the Association for Machine Translation in the Americas (AMTA-2002)*, pages 6–12.

Kuhlmann, Marco and Peter Jonsson. 2015. Parsing to noncrossing dependency graphs. *Transactions of the Association for Computational Linguistics*, 3:559–570.

Kuhlmann, Marco and Joakim Nivre. 2006. Mildly non-projective dependency structures. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 507–514.

Kuhlmann, Marco and Joakim Nivre. 2010. Transition-based techniques for non-projective dependency parsing. *Northern European Journal of Language Technology (NEJLT)*, 2(1):1–19.

Kuhlmann, Marco and Stephan Oepen. 2016. Towards a catalogue of linguistic graph banks. *Computational Linguistics*, 42(4):819–827.

Levin, Beth. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago.

Linh, Ha and Huyen Nguyen. 2019. A case study on meaning representation for Vietnamese. In *Proceedings of the First International Workshop on Designing Meaning Representations*, pages 148–153.

Lison, Pierre and Jörg Tiedemann. 2016. Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 923–929, European Language Resources Association, Paris.

Loper, Edward, Szu-Ting Yi, and Martha Palmer. 2007. Combining lexical resources: Mapping between PropBank and VerbNet. In *Proceedings of the 7th International Workshop on Computational Linguistics*, pages 118–129, Tilburg.

Martí, Maria Antònia, Mariona Taulé, Manu Bertran, and Lluís Màrquez. 2007. Ancora: Multilingual and multilevel annotated corpora. Unpublished manuscript, Universitat de Barcelona.

McDonald, Ryan and Giorgio Satta. 2007. On the complexity of non-projective data-driven dependency parsing. In *Proceedings of the 10th International Conference on Parsing Technologies*, 121–132.

Mel'čuk, Igor A. 2006, Explanatory Combinatorial Dictionary. In G. Sica, editor, *Open Problems in Linguistics and Lexicography*, Polimetrica, Monza, pages 225–355.

Mel'čuk, Igor A. and Aleksandr Žolkovskij. 1984. *Tolkovo-kombinatornyj slovar' russkogo jazyka*. Wiener Slawistische Almanach, Sonderband 14, Vienna.

Menezes, Arul and Stephen D. Richardson. 2003. A best-first alignment algorithm for automatic extraction of transfer mappings from bilingua corpora, In *Recent Advances in Example-Based Machine Translation*. Springer, pages 421–442.

Meyers, Adam, Ruth Reeves, Catherine Macleod, Rachel Szekely, Veronika Zielinska, Brian Young, and Ralph Grishman. 2004. The NomBank project: An interim report. In *Proceedings of the Workshop Frontiers in Corpus Annotation at HLT-NAACL 2004*, pages 24–31, Boston, MA.

Migueles-Abraira, Noelia, Rodrigo Agerri, and Arantza Diaz de Ilarraza. 2018. Annotating abstract meaning representations for Spanish. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*, pages 3074–3078, European Language Resources Association, Paris.

Milićević, Jasmina. 2006. A short guide to the Meaning-Text linguistic theory. *Journal of Koralex*, 8:187–233.

Mille, Simon, Alicia Burga, and Leo Wanner. 2013. AnCora-UPF: A multi-level annotation of Spanish. In *Proceedings of the 2nd International Conference on Dependency Linguistics (DepLing 2013)*, pages 217–226.

Miltsakaki, Eleni, Aravind Joshi, Rashmi Prasad, and Bonnie Webber. 2004a. Annotating discourse connectives and their arguments. In *Proceedings of the Workshop Frontiers in Corpus Annotation at HLT-NAACL 2004*, pages 9–16, Boston, MA.

Miltsakaki, Eleni, Rashmi Prasad, Aravind K. Joshi, and Bonnie L. Webber. 2004b. The Penn Discourse Treebank. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, pages 2237–2240, European Language Resources Association, Paris.

Mirzaei, Azadeh and Amirsaeid Moloodi. 2016. Persian Proposition Bank. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 3828–3835, European Language Resources Association, Paris.

Miyao, Yusuke, Stephan Oepen, and Daniel Zeman. 2014. In-house: An ensemble of pre-existing off-the-shelf parsers. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 335–340.

Nivre, Joakim, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, and Deniz Yuret. 2007. The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 915–932, Prague.

Nivre, Joakim, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the 12th International Conference on Language Resources and Evaluation (LREC 2020)*, pages 4027–4036, European Language Resources Association, Paris.

Nomani, Maaz Anwar, Riyaz Ahmad Bhat, Dipti Misra Sharma, Ashwini Vaidya, Martha Palmer, and Tafseer Ahmed. 2016. A proposition bank for Urdu. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 2379–2386, European Language Resources Association, Paris.

Oepen, Stephan, Omri Abend, Jan Hajič, Daniel Hershcovich, Marco Kuhlmann, Tim O'Gorman, Nianwen Xue, and Milan Straka. 2019. MRP 2019: Cross-framework meaning representation parsing. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pages 1–20, Hong Kong.

Oepen, Stephan, Marco Kuhlmann, Yusuke Miyao, Daniel Zeman, Silvie Cinková, Dan Flickinger, Jan Hajič, and Zdeňka Urešová. 2015. SemEval 2015 task 18: Broad-coverage semantic dependency parsing. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 915–926, Denver, CO.

Oepen, Stephan and Jan Tore Lønning. 2006. Discriminant-Based MRS Banking. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, pages 1250–1255, European Language Resources Association, Paris.

Palmer, Martha, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Panevová, Jarmila. 1974–1975. On verbal frames in functional generative description. *The Prague Bulletin of Mathematical Linguistics*, 22–23:3–40, 17–52.

Pauliny, Eugen. 1943. *Štruktúra slovenského slovesa*, SAVU, Bratislava.

Perlmutter, David M. 1980. Relational grammar in current approaches to syntax. *Syntax and Semantics*, 13:195–229.

Poesio, Massimo. 2004. The MATE/GNOME proposals for anaphoric annotation, revisited. In *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue at HLT-NAACL 2004*, pages 154–162, Boston, MA.

Poesio, Massimo and Renata Vieira. 1998. A corpus-based investigation of definite description use. *Computational Linguistics*, 24(2):183–216.

Poláková, Lucie, Pavlína Jínová, Šárka Zikánová, Eva Hajičová, Jiří Mírovský, Anna Nedoluzhko, Magdaléna Rysová, Veronika Pavlíková, Jana Zdeňková, Jiří Pergler, and Radek Ocelák. 2012. Prague Discourse Treebank 1.0.

Pollard, Carl and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. University of Chicago Press.

Popel, Martin, David Mareček, Jan Štěpánek, Daniel Zeman, and Zdeněk Žabokrtský. 2013. Coordination structures in dependency treebanks. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, pages 517–527, Sofija.

Prange, Jakob, Nathan Schneider, and Omri Abend. 2019. Semantically constrained multilayer annotation: The case of coreference. In *Proceedings of the 1st International Workshop on Designing Meaning Representations*, pages 164–176, Florence.

Prasad, Rashmi, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Rabaldo, Aravind K. Joshi, and Bonnie L. Webber. 2008. The Penn Discourse TreeBank 2.0. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, pages 2961–2968, European Language Resources Association, Paris.

Przepiórkowski, Adam. 2016. Against the argument-adjunct distinction in Functional Generative Description. *The Prague Bulletin of Mathematical Linguistics*, 106(1):5–20.

Pustejovsky, James, Patrick Hanks, and Anna Rumshisky. 2004. Automated Induction of Sense in Context. In *Proceedings of the 5th International Workshop on Linguistically Interpreted Corpora at COLING 2004*, pages 55–58.

Pustejovsky, James, Patrick Hanks, Roser Sauri, Andrew See, Robert Gaizauskas, Andrea Setzer, Dragomir Radev, Beth Sundheim, David Day, Lisa Ferro, and others. 2003. The Timebank Corpus. In *Proceedings of Corpus Linguistics 2003*, volume 2003, pages 647–656, Lancaster.

Pustejovsky, James, Adam Meyers, Martha Palmer, and Massimo Poesio. 2005. Merging PropBank, NomBank, TimeBank, Penn Discourse Treebank and Coreference. In *Proceedings of the Workshop Frontiers in Corpus Annotations II: Pie in the Sky*, pages 5–12.

Rambow, Owen, Bonnie Dorr, Ivona Kučerová, and Martha Palmer. 2003. Automatically deriving tectogrammatical labels from other resources: A comparison of semantic labels across frameworks. The Prague. In *Bulletin of Mathematical Linguistics*, pages 23–35, Citeseer.

Ruppenhofer, Josef, Michael Ellsworth, Myriam Schwarzer-Petruck, Christopher R. Johnson, and Jan Scheffczyk. 2006. *FrameNet II: Extended Theory and Practice*. International Computer Science Institute, Berkeley, CA.

Rysová, Magdaléna, Pavlína Synková, Jiří Mírovský, Eva Hajičová, Anna Nedoluzhko, Radek Ocelák, Jiří Pergler, Lucie Poláková, Veronika Pavlíková, Jana Zdeňková, and Šárka Zikánová. 2016. Prague Discourse Treebank 2.0.

Sahin, Gözde Gül and Esref Adali. 2018. Annotation of semantic roles for the Turkish Proposition Bank. *Language Resources and Evaluation*, 52(3):673–706.

Schuster, Sebastian and Christopher D. Manning. 2016. Enhanced English Universal Dependencies: An improved representation for natural language understanding tasks. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 2371–2378, European Language Resources Association, Paris.

Sgall, Petr. 1967. Functional sentence perspective in a generative description. *Prague Studies in Mathematical Linguistics*, 2:203–225.

Sgall, Petr, Eva Hajičová, and Jarmila Panevová. 1986. *The Meaning of the Sentence in its Semantic and Pragmatic Aspects*. Springer Science & Business Media.

Stanovsky, Gabriel, Jessica Ficler, Ido Dagan, and Yoav Goldberg. 2016. Getting more out of syntax with props. *arXiv preprint arXiv:1603.01648*.

Straka, Milan, Jan Hajič, and Jana Straková. 2016. Udpipe: Trainable pipeline for processing CONLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4290–4297, European Language Resources Association, Paris.

Straka, Milan, and Jana Straková. 2017. Tokenizing, POS tagging, lemmatizing and parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Association for Computational Linguistics, Vancouver.

Surdeanu, Mihai, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. 2008. The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of the 12th Conference on Computational Natural Language Learning (CoNLL 2008)*, pages 159–177.

Taulé, Mariona, M. Antònia Martí, and Marta Recasens. 2008. AnCora: Multilevel annotated corpora for Catalan and Spanish. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, pages 96–101, European Language Resources Association, Paris.

Taulé, Mariona, Aina Peris, and Horacio Rodríguez. 2016. Iarg-AnCora: Spanish corpus annotated with implicit arguments. *Language Resources and Evaluation*, 50:549–584.

Tesnière, Lucien. 1959. *Éléments de syntaxe structurale*. Libraire C. Klincksieck, Paris.

Vaidya, Ashwini, Jinho Choi, Martha Palmer, and Bhuvana Narasimhan. 2011. Analysis of the Hindi Proposition Bank using dependency structure. In *Proceedings of the 5th Linguistic Annotation Workshop*, pages 21–29.

Wanner, Leo. 1996. *Lexical Functions in Lexicography and Natural Language Processing*. John Benjamins Publishing Company, Amsterdam – Philadelphia.

Weischedel, Ralph, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, et al. 2013. OntoNotes release 5.0.

Xue, Nianwen and Martha Palmer. 2009. Adding semantic roles to the Chinese Treebank. *Natural Language Engineering*, 15(1):243–272.

Yakushiji, Akane, Yusuke Miyao, Yuka Tateisi, and Junichi Tsujii. 2005. Biomedical information extraction with predicate–argument structure patterns. In *Proceedings of the 1st International Symposium on Semantic Mining in Biomedicine (SMBM)*, pages 60–69, Hinxton.

Žabokrtský, Zdeněk. 2005. Resemblances between meaning-text theory and functional generative description. In *Proceedings of the 2nd International Conference of Meaning-Text Theory*, pages 549–557, Moscow.

Žabokrtský, Zdeněk, Jan Ptáček, and Petr Pajas. 2008. TectoMT: Highly modular MT system with tectogrammatics used as transfer layer. In *Proceedings of the Third Workshop on Statistical Machine Translation at ACL 2008*, pages 167–170, Columbus, OH.

Zaghouani, Wajdi, Abdelati Hawwari, and Mona Diab. 2012. A pilot PropBank annotation for Quranic Arabic. In *Proceedings of the NAACL-HLT 2012 Workshop on Computational Linguistics for Literature*, pages 78–83. Montréal.

Zeman, Daniel, Jan Hajič, Martin Popel, Martin Potthast, Milan Straka, Filip Ginter, Joakim Nivre, and Slav Petrov. 2018. CoNLL 2018 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–21.

Zhang, Sheng, Rachel Rudinger, and Benjamin Van Durme. 2017. An evaluation of PredPatt and open IE via stage 1 semantic role labeling. In *Proceedings of the 12th International Conference on Computational Semantics (IWCS 2017)*, pages 1–7, Montpellier.

Zhu, Huaiyu, Yunyao Li, and Laura Chiticariu. 2019. Towards universal semantic representation. In *Proceedings of the 1st International Workshop on Designing Meaning Representations*, pages 177–181, Association for Computational Linguistics, Florence.

Zikánová, Šárka, Eva Hajičová, Barbora Hladká, Pavlína Jínová, Jiří Mírovský, Anna Nedoluzhko, Lucie Poláková, Kateřina Rysová, Magdaléna Rysová, and Jan Václ. 2015. *Discourse and Coherence. From the Sentence Structure to Relations in Text. Studies in Computational and Theoretical Linguistics*. ÚFAL, Praha, Czechia.

Žolkovskij, Aleksandr K. 1964. O pravilax semantičeskogo analiza [On rules for semantic analysis]. *Mašinnyj perevod i prikladnaja lingvistika*, (8):17–32.

Žolkovskij, Aleksandr K. and Igor A. Mel'čuk. 1965. O vozmožnom metode i instrumentax semantičeskogo sinteza [On a possible method and instruments for semantic synthesis]. *Naučno-texničeskaja informacija*, (5):23–28.

Žolkovskij, Aleksandr K. and Igor A. Mel'čuk. 1967. O semantičeskom sinteze [On semantic synthesis]. *Problemy kybernetiki*, (19):177–238.