Universal Stanford Dependencies: A cross-linguistic typology

Marie-Catherine de Marneffe[◦], Timothy Dozat[⋆], Natalia Silveira[⋆], Katri Haverinen[•], Filip Ginter[•], Joakim Nivre[◁], Christopher D. Manning[⋆◇]

[◦]Linguistics Department, The Ohio State University
[⋆]Linguistics and [◇]Computer Science Departments, Stanford University
[•]Department of Information Technology, University of Turku
[◁]Department of Linguistics and Philology, Uppsala University

Abstract

Revisiting the now de facto standard Stanford dependency representation, we propose an improved taxonomy to capture grammatical relations across languages, including morphologically rich ones. We suggest a two-layered taxonomy: a set of broadly attested universal grammatical relations, to which language-specific relations can be added. We emphasize the lexicalist stance of the Stanford Dependencies, which leads to a particular, partially new treatment of compounding, prepositions, and morphology. We show how existing dependency schemes for several languages map onto the universal taxonomy proposed here and close with consideration of practical implications of dependency representation choices for NLP applications, in particular parsing.

Keywords: dependency grammar, Stanford Dependencies, grammatical taxonomy

1. Introduction

The Stanford Dependencies (SD) representation (de Marneffe et al., 2006) was originally developed as a practical representation of English syntax, aimed at natural language understanding (NLU) applications. However, it was deeply rooted in grammatical relation-based syntactic traditions, which have long emphasized cross-linguistic description. Faithfulness to these origins was attenuated by desiderata from our NLU applications and the desire for a simple, uniform representation, which was easily intelligible by non-experts (de Marneffe and Manning, 2008). Nevertheless, it is reasonable to suppose that these additional goals do not detract from cross-linguistic applicability.

In this paper, we attempt a (post-hoc) reconstruction of the underlying typology of the Stanford Dependencies representation. This not only gives insights into how the approach might be applied to other languages, but also gives an opportunity to reconsider some of the decisions made in the original scheme, aiming to propose an improved taxonomy, even for English. We suggest a taxonomy which has at its core a set of very broadly attested grammatical relations, supplemented as needed by subtypes for language-particular relations, which capture phenomena important to the syntax of individual languages or language families.

We attempt to make the basic core more applicable, both cross-linguistically and across genres, and more faithful to the design principles in de Marneffe & Manning (2008). We consider how to treat grammatical relations in morphologically rich languages, including achieving an appropriate parallelism between expressing grammatical relations by prepositions versus morphology. We show how existing dependency schemes for other languages which draw from Stanford Dependencies (Chang et al., 2009; Bosco et al., 2013; Haverinen et al., 2013; Seraji et al., 2013; McDonald et al., 2013; Tsarfaty, 2013) can be mapped onto the new taxonomy proposed here. We emphasize the lexicalist stance of both most work in NLP and the syntactic theory on which Stanford Dependencies is based, and hence argue for a particular treatment of compounding and morphology. We also discuss the different forms of dependency representation that should be available for SD to be maximally useful for a wide range of NLP applications, converging on three versions: the basic one, the enhanced one (which adds extra dependencies), and a particular form for parsing.

2. A proposed universal taxonomy

Table 1 gives the taxonomy we propose for the universal grammatical relations, with a total of 42 relations. These relations are taken to be broadly supported across many languages in the typological linguistic literature.[1]

2.1. The representation builds on lexicalism

An under-elaborated part of the first design principle in (de Marneffe and Manning, 2008) is that SD adopts the *lexicalist hypothesis* in syntax, whereby grammatical relations should be between whole words (or *lexemes*). There is a longstanding, unresolved debate in linguistics between theories which attempt to build up both words and phrases using the same compositional syntactic mechanisms (and in which the notion of a word has minimal privileged existence) versus those theories where the word is a fundamental unit and which see the morphological processes that build up words as fundamentally different from and hidden to those that build up sentences, sometimes termed the *lexical integrity principle* (Chomsky, 1970; Bresnan and Mchombo, 1995; Aronoff, 2007). For a practical computational model, there are great advantages to a lexicalist approach. However, there remain certain difficult cases, such as how to deal with certain *clitics* (Zwicky and Pullum, 1983), phonologically bound words which behave like syntactic words (we follow many treebanks in separating them as words and having them participate in the syntax) and how to treat words that are split in unedited writing (see section 2.4.).

[1]This is not to say that all languages have all these grammatical relations. E.g., many languages, from English to Chicheŵa, allow verbs to take more than one object, but other languages, such as French, do not. Nevertheless, *iobj* is broadly attested.

Core dependents of clausal predicates		
Nominal dep	*Predicate dep*	
nsubj	csubj	
nsubjpass	csubjpass	
dobj	ccomp	xcomp
iobj		
Non-core dependents of clausal predicates		
Nominal dep	*Predicate dep*	*Modifier word*
	advcl	advmod
	nfincl	neg
nmod	ncmod	
Special clausal dependents		
Nominal dep	*Auxiliary*	*Other*
vocative	aux	mark
discourse	auxpass	punct
expl	cop	
Coordination		
conj	cc	
Noun dependents		
Nominal dep	*Predicate dep*	*Modifier word*
nummod	relcl	amod
appos	nfincl	det
nmod	ncmod	neg
Compounding and unanalyzed		
compound	mwe	goeswith
name	foreign	
Case-marking, prepositions, possessive		
case		
Loose joining relations		
list	parataxis	remnant
dislocated		reparandum
Other		
Sentence head	*Unspecified dependency*	
root	dep	

Table 1: Dependencies in universal Stanford Dependencies. Note: *nmod*, *ncmod*, *nfincl*, and *neg* appear in two places.

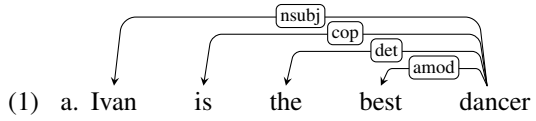
2.2. Dependents of clausal predicates

The system of clausal dependents most closely follows Lexical-Functional Grammar (LFG) (Bresnan, 2001). However, the taxonomy differs from LFG in several respects. First, the clear distinction between core arguments and other dependents is made, but the distinction between adjuncts and oblique arguments (Radford, 1988) is taken to be sufficiently subtle, unclear, and argued over that it is eliminated.[2] Second, the model reverts to the traditional grammar notion of direct object and other objects. In cases of multiple objects, the theme/patient argument is the direct object. Finally, the taxonomy aims to clearly indicate in the dependency label (i) a non-canonical voice subject (where the proto-agent argument is not subject, i.e., in passives) and (ii) whether dependents are noun phrase (NP) arguments versus introducing a new clause/predicate. A design goal of SD has been to distinguish in dependency names where a new clause is being introduced (and so we distinguish *nsubj* and *csubj*; *dobj* and *ccomp*, but also *advmod* and *advcl*). We follow LFG in including a distinction between *ccomp* and *xcomp* for clausal complements that are standalone (have an internal subject) versus those having necessary control (omission) of the dependent predicate's subject (have an external subject).[3]

Other aspects of the typology are less regular but still important. The non-core clausal dependents are all modifiers. The distinction between a full adverbial clause *advcl* and a participial or infinitive nonfinite clause *nfincl* is similar but not exactly parallel to the *ccomp*/*xcomp* distinction.[4] Clause heads have many other special dependents, including periphrastic auxiliaries, markers (complementizers and subordinating conjunctions) as well as vocatives and discourse elements (like *well* or *um*). Conjunctions combine elements of many categories. Under the SD design principles, the *conj* relation connects lexical heads.

2.3. Treatment of copulas

SD has advocated a treatment of the copula "be" where it is not the head of a clause, but rather the dependent of a lexical predicate, as in (1a). Such an analysis is motivated by the fact that many languages often or always lack an overt copula in such constructions, as in the Russian (1b). Similar constructions arise even in English if we consider so-called raising-to-object or small clause constructions. Under the basic analysis proposed for SD, the predicate complement is not linked to its subject argument, but in the enhanced representation (see below), the linkage is then parallel to the treatment in a zero copula language, as in (1c).

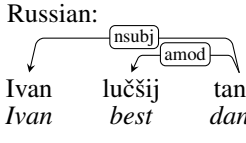


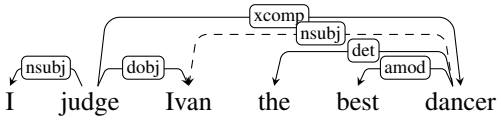
[2]The original Penn Treebank annotators also decided not to try to mark arguments vs. adjuncts (Taylor et al., 2003). Conversely, the Penn Chinese Treebank does try to make an adjunct/argument distinction (Xue et al., 2005) and effectively PropBank (Palmer et al., 2005) adds an argument/adjunct distinction overlay to the English Penn Treebank.

[3]The latter is used to represent control and raising constructions, including cases of raising-to-object or so-called "exceptional case marking". The treatment of this last group of phenomena is one of the largest substantive breaks with the Penn Treebank annotation tradition, which follows Chomskyan approaches from the extended standard theory, Government-Binding Theory (Chomsky, 1981) et seq., in treating such constructions as having a complete clausal complement with "exceptional case marking", rather than an object in the higher clause.

[4]For *ccomp* vs. *xcomp*, the defining difference is a controlled subject. While an *xcomp* is always non-finite, there are also non-finite *ccomp*, such as in a *for-to* infinitive ("I arranged [for her to go by bus]"). For *advcl* vs. *nfincl* the distinction is a finite clause vs. a non-finite clause, though usually a reduced one lacking a subject. We generalize the previous *partmod* and *infmod* to *nfincl* for more generality and cross-linguistic applicability. We use *nfincl* rather than *vmod* since this modifier can also be an adjective, as in "She hesitated [unable to remember his name]".

b. Russian:





c.

2.4. Modification vs. compounding

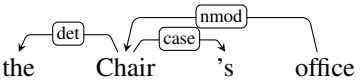
The universal scheme keeps the rich taxonomy of noun dependents that are one of the strengths of SD. An important part of the typology is to differentiate compounds (multi-root lexemes) from modification or complementation. Under a lexicalist approach, compound words are fundamentally different from cases of phrasal modification.

There are three relations for compounding. We use *mwe* for fixed grammaticized expressions with function words (e.g., *instead of*: *mwe*(of,instead), Fr. *plutôt que* "rather than": *mwe*(que,plutôt)), *name* for proper nouns constituted of multiple nominal elements, as in the Finnish and Italian dependency treebanks,[5] and *compound* to label other types of multi-word lexemes. Thus *compound* is used for any kind of X^0 compounding: noun compounds (e.g., *phone book*), but also verb and adjective compounds that are more common in other languages (such as Persian or Japanese light verb constructions); for numbers (e.g., *three thousand books* gives *compound*(thousand,three)); for particles of phrasal verbs (e.g., *put up*: *compound*(put,up)).

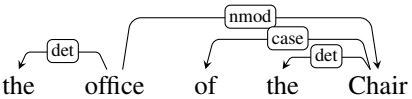
2.5. Treatment of prepositions and case marking

A major proposed change from the extant versions of SD is a new treatment of prepositions to provide a uniform analysis of prepositions and case in morphologically rich languages. The analysis we chose is to push all the way the design principle of having direct links between content words. We abandon treating a preposition as a mediator between a modified word and its object, and, instead, any case-marking element (including prepositions, postpositions, and clitic case markers) will be treated as a dependent of the noun it attaches to or introduces. The proposed analysis is shown in (2): *nmod* labels the relation between the two content words, whereas the preposition is now viewed as a *case* depending on its complement. In general, *nmod* expresses some form of oblique or adjunct relation further specified by the *case*.

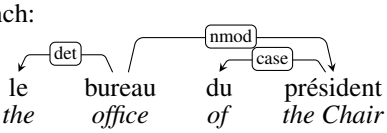
(2) a.



b.



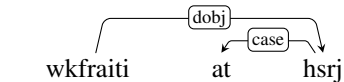
c. French:



[5]That is, *name* would be used between the words of "Hillary Rodham Clinton" but not to replace the usual relations in a phrasal or clausal name like "The Lord of the Rings".

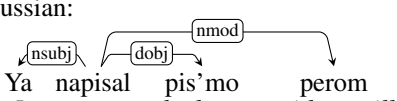
The treatment of case marking is illustrated in (3). In (3a), *at* in Hebrew is a separate token indicating an accusative object: the case marker depends on the object. In (3c), we show the analysis when case markers are morphemes. The case morpheme is not divided off the noun as a separate *case* dependent, but the noun as a whole is analyzed as a *nmod* of the verb. To overtly mark case, we include POS tags in the representation as shown in (3b) and (3d). We use the universal POS tagset from Petrov et al. (2012) to which we append case information.

(3) a. Hebrew:



b. *dobj*(wkfraiti/VERB, hsrj/NOUN)
case(hsrj/NOUN, at/PRT-ACC)

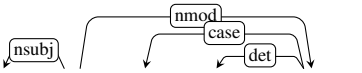
c. Russian:



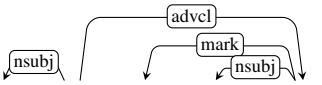
d. *nsubj*(napisal/VERB, Ya/NOUN-NOM)
dobj(napisal/VERB, pis'mo/NOUN-ACC)
nmod(napisal/VERB, perom/NOUN-INSTR)

This treatment provides parallelism between different constructions across and within languages. A good result is that we now have greater parallelism between prepositional phrases and subordinate clauses, which are in practice often introduced by a preposition, as in (4).

(4) a.

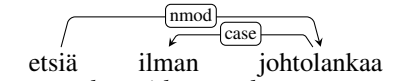


b.

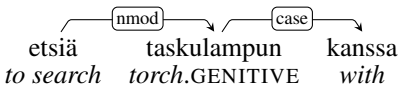


We also obtain parallel constructions for the possessive alternation as shown in (2), for variant forms with case, a preposition or a postposition in Finnish, as shown in (5), and for the dative alternation where the prepositional construction gets a similar analysis to the double object construction, see (6).

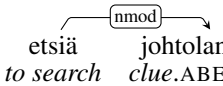
(5) a. Finnish:



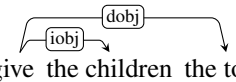
b.

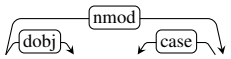


c.



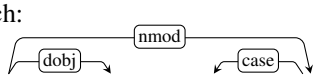
(6) a.





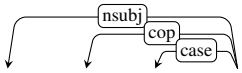
b. give the toys to the children

c. French:



donner les jouets aux enfants
give the toys to the children

Another advantage of this new analysis is that it provides a treatment of prepositional phrases that are predicative complements of "be" as in (7) that is consistent with the treatment of nominal predicative complements, as in (1).

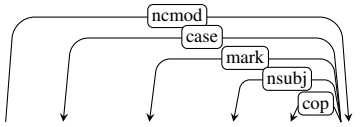


(7) Sue is in shape

SD is a surface syntactic representation, which does not represent semantic roles. The semantic roles of modifiers are hard to categorize and hard to determine. We feel that there is a lot of use for a representation which works solely in terms of the overt role-marking resources of each language. This is supported by many rich language-particular traditions of grammatical analysis, whether via Sanskrit cases or the case particles on Japanese *bunsetsu*.

Prepositions sometimes introduce a clause as their complement, e.g., (8a). Following the principle that dependencies do mark where new clauses are introduced, this relation should have a different name from *nmod*, and we suggest calling it *ncmod* "nominalized clause modifier". Under the proposed new analysis, the head of the modifier of *data* will be *upset*. The result will be the analysis in (8b).

(8) a. We have no data about whether users are upset.



b. data about whether users are upset

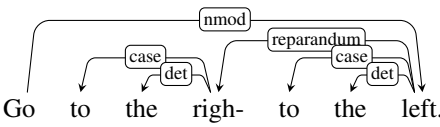
Another issue is what analysis to give to cases of stacked prepositions, such as *out of*. Our proposal is that all such cases should be regarded as some form of *mwe*, as in (9b).

(9) a. Out of all this, something good will come.

b. Out of all this ... come

2.6. Informal text genres

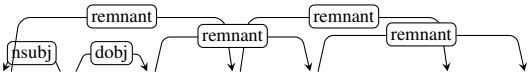
Following the practical approach used in part-of-speech tagging of recent LDC treebanks, we introduce the relation *goeswith* to connect multiple tokens that correspond to a single standard word, as a result of reanalysis of words as compounds ("hand some" for "handsome") or input error ("othe r" for "other"). We use *foreign* to label sequences of foreign words. To indicate disfluencies overridden in a speech repair, we use *reparandum*, as in (10).



(10) Go to the righ- to the left.

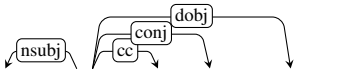
The "loose joining relations" aim at a robust analysis of more informal forms of text, which are now common in NLP applications. Informal written text often contains lists of comparable items, which are parsed as single sentences. Email signatures in particular contain these structures, in the form of contact information. Following de Marneffe et al. (2013), we use the *list*, *parataxis*, (and *appos*) relations to label these kinds of structures. The relation *parataxis* is also used in more formal writing for constructions such as sentences joined with a colon.

The *dislocated* relation captures preposed (topics) and postposed elements. The *remnant* relation is used to provide a treatment of ellipsis (in the case of gapping or stripping, where predicational or verbal material gets elided), something that was lacking in earlier versions of SD. It provides a basis for being able to reconstruct dependencies in the enhanced version of SD. For example, in (11), the *remnant* relations enable us to correctly retrieve the subjects and objects in the clauses with an elided verb.

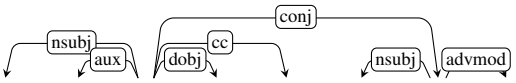


(11) John won bronze, Mary silver, and Sandy gold.

In contrast, in right-node-raising (RNR) (12) and VP-ellipsis (13) constructions in which some kind of predicational or verbal material is still present, the *remnant* relation is not used. In RNR, the verbs are coordinated and the object is a *dobj* of the first verb. In VP-ellipsis, we keep the auxiliary as the head, as shown in (13).



(12) John bought and ate an apple.



(13) John will win gold and Mary will too.

2.7. Language-particular relations

In addition to a universal dependency taxonomy, it is desirable to recognize grammatical relations that are particular to one language or a small group of related languages. Such language-particular relations are necessary to accurately capture the genius of a particular language but will not involve concepts that generalize broadly. The suggestion here is that these relations should always be regarded as a subtype of an existing Universal SD relation. The SD relations have a taxonomic organization (de Marneffe et al., 2006), and some of the universal relations are already subtypes of each other (e.g., *auxpass* is a subtype of *aux*). Language-particular relations that seem useful to distinguish for English are included at the bottom of Table 2: *npmod* for bare nominal modifiers of predicates lacking a preposition, among which in particular there is *tmod* for bare NP temporal modifiers; *poss* for possessives, since the syntax of prenominal possession is very distinct from postnominal modifiers (which may also express possession); *predet* for words such as *all* that precede regular determiners and *preconj* for words preceding a conjunction like *either*; and *prt* for verb particles.

3. Mapping to existing schemes

There has recently been an effort to push towards homogeneity across resources for different languages and to come up with cross-linguistically consistent annotation aimed at multi-lingual technology, for part-of-speech tagset (Petrov et al., 2012) as well as for dependency representation (McDonald et al., 2013). The scheme proposed in McDonald et al. (2013) took SD as a starting point. Annotators for six different languages (German, English, Swedish, Spanish, French and Korean) produced annotation guidelines for the language they were working on, keeping the label and construction set as close as possible to the original English SD representation. They were only allowed to add labels for phenomena that do not exist in English. Given the sets of relations obtained for the different languages, a harmonization step was performed to maximize consistency of annotation across languages. However, this rigid strategy lost some important distinctions, such as the distinction between compounding and phrasal modification, while maintaining some distinctions that are best abandoned, such as a distinction between infinitival and participial modifiers.

McDonald et al. (2013) does not address giving an elegant treatment of morphologically rich languages. In contrast, Tsarfaty (2013) proposes to treat morphology as syntax in her dependency proposal, illustrated with Hebrew. However, this representation choice conflicts with the lexicalist approach of SD. Here we take up her goal of trying to give a uniform treatment of both morphologically rich and morphologically poor languages, but suggest achieving the goal in a different way, which maintains a lexicalist approach (see Section 2.5.). Table 2 shows a comparison between the evolution of the SD scheme (Stanford Dependencies v2.0.0, used in the SANCL shared task, and Stanford Dependencies v3.3.0, the 2013/11/12 version), the proposals in McDonald et al. (2013) (GSD) and in Tsarfaty (2013) (TSD), and the dependency set proposed here (USD).

Existing dependency sets for other languages can be fairly straightforwardly mapped onto our new proposal. Even if the schemes examined here are "SD-centric", they dealt with particular constructions present in each language and posited new relations when necessary. The mapping is less difficult because USD adopts some of the ideas and relations that were first developed for these other treebanks, such as the content word as head analysis of prepositional phrases from the Turku Dependency Treebank (TDT). In table 3, we show how the Finnish (Haverinen et al., 2013), Italian (Bosco et al., 2013), Chinese (Chang et al., 2009) and Persian (Seraji et al., 2013) schemes can be mapped onto the proposed universal taxonomy (USD). The bold labels are language-specific relations, and they are subtypes of the corresponding USD relation in the row. Gaps indicate existing constructions in the language that were not captured in the original scheme (the USD label is applicable there); $*$ indicates constructions that are not present in the language. Since copular verbs are not heads anymore, the *attr* relation is removed, requiring modifications to the existing analyses of copular sentences for Italian and Chinese. We also introduced extra relations for certain constructions, which some schemes had not incorporated yet.

For Finnish, the relation *rel* (for relative marker) will be mapped to whatever syntactic role the relative is playing in the relative clause (*nsubj*, *dobj*, etc.), information which is present in the second annotation layer of the TDT corpus.

ISDT is the conversion of the MIDT Italian dependency scheme to SD. Some of the *clit* uses in ISDT (for reflexive pronouns in pronominal verbs – frequent in Romance languages – such as Fr. *se douter* "to suspect") will be encompassed by *expl*. However, when the reflexive pronoun can truly be a direct or indirect object, it gets assigned the corresponding object relation.

Chinese has serial verb constructions which might now be *compound* (and not *conj*). We treat post-nominal localizers and prepositions as a form of case.

In Persian, there are no relative pronouns and *rel* was used for the fixed relative marker, but it can be mapped to *mark*. UPDT has a *fw* relation between sequences of foreign words, unanalyzed within Persian grammar, which we adopt, naming it *foreign*. We also adopt the UPDT *dep-top* relation used for a fronted element that introduces the topic of a sentence, but we generalize it to *dislocated* to account for postposed elements as well as. Right dislocated elements are frequent in spoken languages: e.g., Fr. *faut pas la manger, la pâte* (literally, "need not it eat, the dough").

Labels of language-specific relations will be harmonized to be shared between languages: Chinese *assmod* will be mapped to *poss*, and Persian *dep-top* to *topic*.

4. Different forms of Stanford Dependencies

The current Stanford converter provides a number of variant forms of SD, of which the most used are the **basic** dependency tree, and the **collapsed, cc-processed** form that adds extra dependency arcs, restructures prepositions to not be heads, and spreads relations across conjunctions. This section suggests some new ideas for how to provide potentially less but different options.

One concern about our proposed taxonomy is that straightforward parsing to USD is likely to be harder for parsers than the current representation (for English). It is now fairly well known that, while dependency representations in which content words are made heads tend to help semantically oriented downstream applications, dependency parsing numbers are higher if you make auxiliary verbs heads, if you analyze long conjunctions by chaining (rather than having a single head of the whole construction), and if you make prepositions the head of prepositional phrases (Schwartz et al., 2012; Elming et al., 2013). The generalization is that dependency parsers, perhaps in particular the efficient shift-reduce-style dependency parsers like MaltParser (Nivre et al., 2007), work best the more the dependency representation looks like a chain of short dependencies along the sentence. Under the proposed USD, SD would then be making the "wrong" choice in each case.

However, it seems wrong-headed to choose a linguistic representation to maximize parser performance, rather than based on the linguistic quality of the representation and its usefulness for applications that build further processing on top of it. Rather, it may be useful to do parsing using a transformation of the target dependency system. In constituency parsing, it is completely usual for the target representation to be transformed so as to improve

SD v2.0.0	SD v3.3.0	GSD	TSD	USD	Notes
nsubj	nsubj	nsubj	nsubj	nsubj	✓
csubj	csubj	csubj	csubj	csubj	✓
nsubjpass	nsubjpass	nsubjpass	nsubjpass	nsubjpass	✓
csubjpass	csubjpass	csubjpass	csubjpass	csubjpass	✓
dobj	dobj	dobj	dobj	dobj	✓
iobj	iobj	iobj	iobj	iobj	✓ (TSD also has *gobj* for genitive object)
ccomp	ccomp	ccomp	ccomp	ccomp	USD & TSD define as clause with internal subject, not finite
xcomp	xcomp	xcomp	xcomp	xcomp	USD & TSD define as clause with external subject, not nonfinite
acomp	acomp	acomp	acomp	–	*acomp* can be generalized into *xcomp*
attr	–	attr	–	–	*attr* removed: *wh-* is head or *xcomp* (with copula head option)
advmod	advmod	advmod	advmod	advmod	✓
advcl	advcl	advcl	–	advcl	TSD omits but needed to preserve clause boundaries
purpcl	–	–	–	–	Folded into *advcl*
neg	neg	neg	neg	neg	As well as adverbial *not, never*, USD extends to negative *det* like *no*
det	det	det	det	det	✓ (TSD defines *dem* and *def* as subtypes of *det*)
amod	amod	amod	amod	amod	✓
appos	appos	appos	appos	appos	✓
abbrev	–	–	abbrev	–	Parenthetical abbreviations become cases of *appos*
num	num	num	nummod	nummod	Renamed for clarity
rcmod	rcmod	rcmod	rcmod	relcl	✓
partmod	partmod	partmod	?	nfincl	Make *partmod*, *infmod* into *nfincl*; use (rich) POS to distinguish
infmod	infmod	infmod	infmod	nfincl	Make *partmod*, *infmod* into *nfincl*; use (rich) POS to distinguish
quantmod	quantmod	advmod	?	–	Generally folded into *advmod*
root	root	ROOT	root	root	
punct	punct	p	punct	punct	
aux	aux	aux	aux	aux	✓ (TSD adds *qaux* for question auxiliary. Infinitive is now *mark*.)
auxpass	auxpass	auxpass	auxpass	auxpass	✓
cop	cop	cop	cop	cop	GSD has *cop* only in content-head version
expl	expl	expl	expl	expl	Subject and object expletives, frozen reflexives (Fr. *se douter*)
mark	mark	mark	mark	mark	✓ *to* introducing an infinitive will now be *mark* (instead of *aux*)
complm	–	–	complm	–	Remove and use *mark* more broadly
–	discourse	–	parataxis?	discourse	A gap in original and other typologies
–	–	–	–	vocative	A gap in original and other typologies
dep	dep	dep	dep	dep	GSD uses for *vocative* and *discourse*
rel	–	rel	rel	–	Converter's unresolved ones now *dep*; TSD *rel* is really *mark*
prep	prep	adpmod	prepmod	case	USD *case* is dependent of NP modifier not of thing modified
–	–	nmod	–	–	Equivalent to *nmod* below
pobj	pobj	adpobj	pobj	nmod	*nmod* now goes from thing modified to NP of PP
pcomp	pcomp	adpcomp	pcomp	ncmod	*ncmod* goes from thing modified to clause
–	–	adp	prep/case	case	TSD has N/A/G/D subtypes, but can't keep adding for all cases
possessive	possessive	adp	gen	–	View as a manifestation of *case*
nn	nn	compmod	nn	compound	Generalize *nn* to light verbs, etc.; X^0 compounding not modification
–	–	mwe	–	name	Multi-word proper nouns (e.g., *John Smith*) as in TDT and ISDT
number	number	num	nummod?	–	Regarded as type of compound; using *nummod* is wrong
mwe	mwe	mwe	mwe	mwe	Fixed expressions with function words (*so that, all but, due to,* ...)
–	goeswith	–	–	goeswith	For orthographic errors: *othe r*
–	–	–	–	foreign	Linear analysis of foreign words (head is left-most) as in UPDT
–	–	–	–	reparandum	For disfluencies overridden in speech repairs
conj	conj	conj	conj	conj	✓
cc	cc	cc	cc	cc	✓
parataxis	parataxis	parataxis	parataxis	parataxis	✓
–	–	–	–	list	Used for informal list structures, signature blocks, etc.
–	–	–	–	remnant	Used to give a treatment of ellipsis without empty nodes
–	–	–	–	dislocated	Preposed topics and dislocated elements as in UPDT
English particular					
npadvmod	npadvmod	nmod	advmod?	**npmod**	A subtype of *nmod*
tmod	tmod	advmod	tmod	**tmod**	A subtype of *npmod*
predet	predet	–	predet	**predet**	A subtype of *det*
preconj	preconj	cc	preconj	**preconj**	A subtype of *cc*
prt	prt	prt	?	**prt**	A subtype of *compound*
poss	poss	poss	possmod	**poss**	A subtype of *case*

Table 2: Comparison of proposals on English: SD, McDonald et al. (2013) (GSD), Tsarfaty (2013) (TSD) and ours (USD).

TDT	ISDT	Chinese	UPDT	USD
nsubj	nsubj	nsubj, top	nsubj	nsubj
csubj	csubj		∗	csubj
∗	nsubjpass	nsubjpass	nsubjpass	nsubjpass
∗	csubjpass		∗	csubjpass
dobj	dobj , clit	dobj	dobj	dobj
∗	iobj , clit	iobj	∗	iobj
ccomp, **iccomp**	ccomp	ccomp, rcomp	ccomp	ccomp
xcomp, acomp	xcomp, acomp		xcomp, acomp	xcomp
∗	attr	attr	∗	∗
advmod, quantmod	advmod	advmod, dvpmod	advmod, quantmod	advmod
advcl	advcl	some advmod	advcl	advcl
neg	neg	neg	neg	neg
det	det, **predet**	det	det, **predet**	det
amod	amod	amod	amod	amod
appos	appos	prnmod	appos	appos
num	num	nummod, ordmod	num	nummod
rcmod	rcmod	rcmod	rcmod	relcl
partmod, infmod	partmod	vmod	∗	nfincl
root	root	root	root	root
punct	punct	punct	punct	punct
aux	aux	asp, mmod	aux	aux
auxpass	auxpass	pass	auxpass	auxpass
cop	cop	cop	cop	cop
∗	expl, clit	∗	∗	expl
complm, mark	mark	cpm	complm, mark, **rel**	mark
intj	discourse			discourse
voc			dep-voc	vocative
dep	comp, mod	dep	dep	dep
poss, **gobj**, **gsubj**, nommod	pobj, **poss**, npadvmod, **tmod**	pobj, **lobj**, **assmod**, clf, range, **tmod**	pobj, **poss**, **cpobj** npadvmod, **tmod**	nmod
∗	pcomp	pccomp, **lccomp**	∗	ncmod
adpos	possessive, prep	assm, prep, **ba**, **dvpm**, **loc**	**acc**, prep, **cprep**	case
number, nn, **prt**	number, nn	nn, some conj	number, nn, **prt**, {**nsubj**|**dobj**|**acomp**|**prep**}**-lvc**	compound
name	nnp			name
some dep	mwe	prtmod	mwe	mwe
	goeswith			goeswith
			fw	foreign
				reparandum
conj	conj	conj, etc, comod	conj	conj
cc, **preconj**	cc, **preconj**	cc	cc, **preconjunct**	cc
parataxis	parataxis		parataxis	parataxis
				list
ellipsis				remnant
			dep-top	dislocated

Table 3: Mappings of the Finnish (TDT), Italian (ISDT), Chinese and Persian (UPDT) schemes to USD.

parsing numbers, such as by head-lexicalization (Collins, 2003), by manual or automatic subcategorization of categories (Klein and Manning, 2003; Petrov et al., 2006), and even by other methods such as unary chain contraction (Finkel et al., 2008). After parsing, a detransformation process reconstructs trees in the target representation. This kind of transform-detransform architecture is at present less common in dependency parsing, although Nilsson, Nivre & Hall (2006; 2007) do this for coordination and verb groups, and pseudo-projective parsing (Nivre and Nilsson, 2005) can also be seen as an instance of this architecture. A transform-detransform architecture should become more standard in dependency parsing. We therefore propose a **parsing** representation that changes some of the dependency head choices to maximize parsing performance. This requires developing tools to convert seamlessly both ways between the **basic** and **parsing** representations.[6]

Since the new treatment of prepositional phrases basically does what the **collapsed** representation was designed to do (putting a direct link between the noun complement of a preposition and what it modifies), except for not renaming

[6] A small part of this is in place in the Stanford converter, in the ability to generate copula- and content-head versions from trees.

the dependency relation, the **collapsed** representation on its own has less utility. However, the ideas of having extra dependencies to mark external subjects and the external role in relative clauses is useful, the renaming of dependencies to include case or preposition information helps in many applications, and spreading relations over conjunctions is definitely useful for relation extraction. These transformations can be provided in an **enhanced** representation.
We thus suggest providing three versions of Stanford Dependencies: **basic**, **enhanced**, and **parsing**.

5. Conclusion

We proposed a taxonomy of grammatical relations applicable to a variety of languages, developing the implications of a lexicalist approach for the treatment of morphology and compounding. Some of the decisions made on linguistic grounds are at odds with what works best for processing tools. We suggested that the transform-detransform architecture standardly used in constituency parsing is the solution to adopt for dependency parsing. We worked out the mapping of existing dependency resources for different languages to the taxonomy proposed here. We hope this work will enhance consistency in annotation between languages and further facilitate cross-lingual applications.

6. References

Aronoff, M. (2007). In the beginning was the word. *Language*, 83:803–830.

Bosco, C., Montemagni, S., and Simi, M. (2013). Converting Italian treebanks: Towards an Italian Stanford dependency treebank. In *Seventh Linguistic Annotation Workshop & Interoperability with Discourse*.

Bresnan, J. and Mchombo, S. A. (1995). The lexical integrity principle: Evidence from Bantu. *Natural Language and Linguistic Theory*, 13:181–254.

Bresnan, J. (2001). *Lexical-Functional Syntax*. Blackwell, Oxford.

Chang, P.-C., Tseng, H., Jurafsky, D., and Manning, C. D. (2009). Discriminative reordering with Chinese grammatical relations features. In *Third Workshop on Syntax and Structure in Statistical Translation*, pages 51–59.

Chomsky, N. (1970). Remarks on nominalization. In Jacobs, R. A. and Rosenbaum, P. S., editors, *Readings in English transformational grammar*, pages 184–221. Ginn, Waltham, MA.

Chomsky, N. (1981). *Lectures on Government and Binding*. Foris, Dordrecht.

Collins, M. (2003). Head-driven statistical models for natural language parsing. *Computational Linguistics*, 29:589–637.

de Marneffe, M.-C. and Manning, C. D. (2008). The Stanford typed dependencies representation. In *Workshop on Cross-framework and Cross-domain Parser Evaluation*.

de Marneffe, M.-C., MacCartney, B., and Manning, C. D. (2006). Generating typed dependency parses from phrase structure parses. In *LREC*.

de Marneffe, M.-C., Connor, M., Silveira, N., Bowman, S. R., Dozat, T., and Manning, C. D. (2013). More constructions, more genres: Extending Stanford dependencies. In *DepLing 2013*.

Elming, J., Johannsen, A., Klerke, S., Lapponi, E., Martinez, H., and Søgaard, A. (2013). Down-stream effects of tree-to-dependency conversions. In *NAACL HLT 2013*.

Finkel, J. R., Kleeman, A., and Manning, C. D. (2008). Efficient, feature-based, conditional random field parsing. In *ACL 46*, pages 959–967.

Haverinen, K., Nyblom, J., Viljanen, T., Laippala, V., Kohonen, S., Missilä, A., Ojala, S., Salakoski, T., and Ginter, F. (2013). Building the essential resources for Finnish: the Turku dependency treebank. *Language Resources and Evaluation*. In press. Available online.

Klein, D. and Manning, C. D. (2003). Accurate unlexicalized parsing. In *ACL 41*, pages 423–430.

McDonald, R., Nivre, J., Quirmbach-Brundage, Y., Goldberg, Y., Das, D., Ganchev, K., Hall, K., Petrov, S., Zhang, H., Täckström, O., Bedini, C., Bertomeu Castelló, N., and Lee, J. (2013). Universal dependency annotation for multilingual parsing. In *ACL 51*.

Nilsson, J., Nivre, J., and Hall, J. (2006). Graph transformations in data-driven dependency parsing. In *COLING 21 and ACL 44*, pages 257–264.

Nilsson, J., Nivre, J., and Hall, J. (2007). Tree transformations for inductive dependency parsing. In *ACL 45*.

Nivre, J. and Nilsson, J. (2005). Pseudo-projective dependency parsing. In *ACL 43*, pages 99–106.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., Marinov, S., and Marsi, E. (2007). MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13:95–135.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31:71–105.

Petrov, S., Barrett, L., Thibaux, R., and Klein, D. (2006). Learning accurate, compact, and interpretable tree annotation. In *COLING 21 and ACL 44*, pages 433–440.

Petrov, S., Das, D., and McDonald, R. (2012). A universal part-of-speech tagset. In *LREC*.

Radford, A. (1988). *Transformational Grammar*. Cambridge University Press, Cambridge.

Schwartz, R., Abend, O., and Rappoport, A. (2012). Learnability-based syntactic annotation design. In *COLING 24*, pages 2405–2422.

Seraji, M., Jahani, C., Megyesi, B., and Nivre, J. (2013). Uppsala Persian dependency treebank annotation guidelines. Technical report, Uppsala University.

Taylor, A., Marcus, M., and Santorini, B. (2003). The Penn treebank: An overview. In Abeillé, A., editor, *Building and Using Parsed Corpora*, volume 20 of *Text, Speech and Language Technology*. Springer.

Tsarfaty, R. (2013). A unified morpho-syntactic scheme of Stanford dependencies. In *ACL 51*.

Xue, N., Xia, F., Chiou, F.-D., and Palmer, M. (2005). The Penn Chinese treebank: Phrase structure annotation of a large corpus. *Natural Language Engineering*, 11:207–238.

Zwicky, A. M. and Pullum, G. K. (1983). Cliticization vs. inflection: English *n't*. *Language*, 59:502–513.