Nested coordination in Universal Dependencies

Adam Przepiórkowski[1,2,3] **Agnieszka Patejuk**[1,4]
[1]Institute of Computer Science, Polish Academy of Sciences
[2]Institute of Philosophy, University of Warsaw
[3]Wolfson College, University of Oxford
[4]Faculty of Linguistics, Philology and Phonetics, University of Oxford
adamp@ipipan.waw.pl aep@ipipan.waw.pl

Abstract

The aim of this paper is to extend the representation of coordination in Universal Dependencies in a way that makes it possible to distinguish between different embeddings in coordinate structures.

1 Introduction

One principled problem with the UD approach to coordination known to the Universal Dependencies (UD; `http://universaldependencies.org/`; Nivre et al., 2016) community concerns nested – i.e. immediately embedded – coordination, as in:[1]

(1) Tom and Jerry and Scooby-Doo

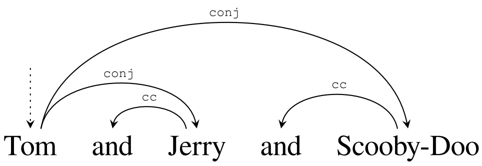
There are three possible ways to structure (1):

(2) Tom and Jerry and Scooby-Doo
(3) [Tom and Jerry] and Scooby-Doo
(4) Tom and [Jerry and Scooby-Doo]

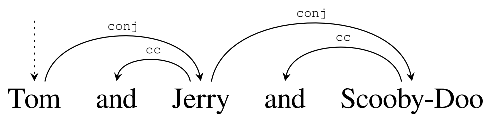
That is, (1) may be construed as flat ternary coordination (cf. (2)), or as binary coordination whose first (cf. (3)) or second (cf. (4)) conjunct is itself a coordinate structure.

UD is not able to distinguish the first two structures, (2)–(3) – it assigns the same representation (5) to both, while representing (4) as (6):

(5)



(6)



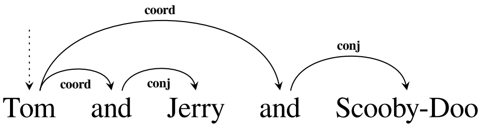
Przepiórkowski and Patejuk, 2019, §4.3 observe that this is not a fundamental problem in practice: in about a dozen cases of nested coordination (out of over 17,000 sentences) in the UD_Polish-LFG treebank of Polish, all or almost all involve a contrastive conjunction such as *ale* 'but', which is strictly binary. Hence, it is often easy to 'disambiguate' representations such as (5) to a truly nested structure: if one of the conjuncts is binary, this cannot be a flat ternary structure. Perhaps for this reason this theoretical flaw has been tolerated in the UD community. Nevertheless, as argued e.g. in Borsley, 2005, 468–469, sequences such as (1) are truly ambiguous between the structures indicated in (2)–(4), so it would be at least theoretically desirable for UD to be able to represent such different nestings.

2 Nested Coordination in Dependency Grammars

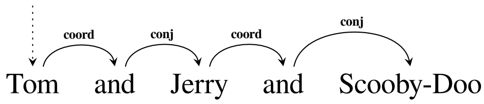
A similar problem occurs in the surface syntactic dependency representation of coordination assumed by Igor Mel'čuk's Meaning–Text Theory (MTT), where conjuncts and conjunctions form a chain; on that approach, the structure (3) is distinguished from the other two, as it gets the basic representation in (7), but the other two structures, (2) and (4), share the basic representation in (8):

[1]See `http://universaldependencies.org/u/dep/conj.html#nested-coordination` (last referenced on 16 June 2019), where this problem is explicitly pointed out.

(7)



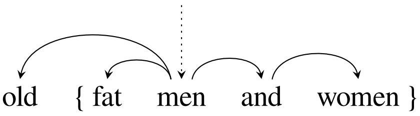
(8)



Contemporary dependency theories deal with various coordination problems by giving up pure dependency representations and introducing additional constituency-like structures: *groupings* in MTT[2] and *word strings* in Richard Hudson's Word Grammar (WG) and related dependency approaches.[3] Groupings are unordered sets containing nodes of contiguous dependency trees. They seem to be used in MTT if and only if the need arises to indicate that a coordinate structure should be treated as a whole. Mel'čuk (2009, 100) illustrates this need with example (9) on the reading where both men and women are described as old, but only men as fat. On this reading, the representation of (9) is (10) (ignoring dependency labels):

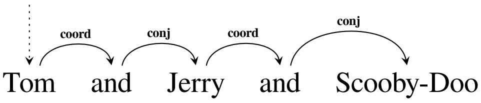
(9) old fat men and women

(10)

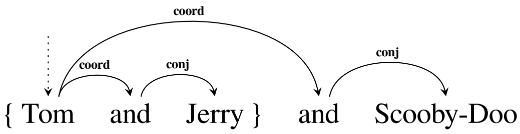


On this view, groupings are superimposed on a well-formed dependency tree, i.e. groupings do not act as nodes in dependency trees. When applied to nested coordination, the three representations of (2)–(4) could be (11)–(13), respectively (Mel'čuk, 2009, 101):

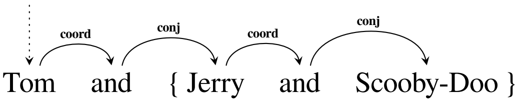
(11)



(12)

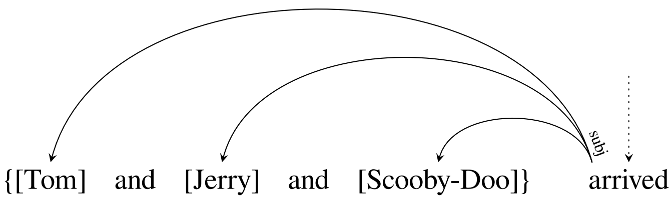


(13)



In Word Grammar, whole coordinate structures and particular conjuncts are analysed as constituents, called 'word strings': they may contain words or other 'word strings', i.e., they may actually have a hierarchical structure. Such 'word strings' are marked with curly brackets in the case of coordinate structures and by square brackets in the case of those conjuncts which are not coordinate structures themselves:

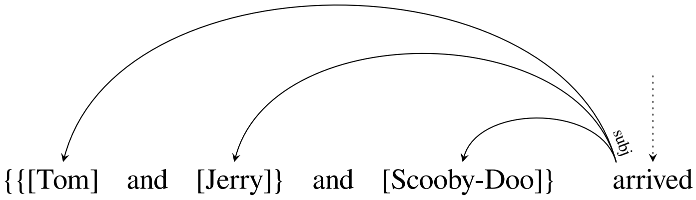
(14)



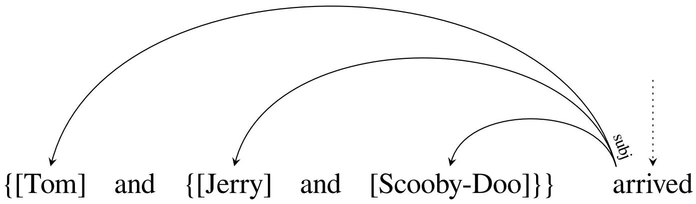
[2]See e.g. Mel'čuk, 1964, 25, Mel'čuk, 1974, 214–216, Mel'čuk and Pertsov, 1987, 74, Mel'čuk, 1988, 28–33, Mel'čuk, 2009, 94, 100–101.

[3]See e.g. Hudson, 1980, 496–499, Hudson, 1984, 211–240, Hudson, 1988, 1989, Hudson, 1990, 404–421 and Hudson, 2018, §4.2, as well as Pickering and Barry, 1993 and Osborne, 2006b,a, Osborne and Groß, 2017. For a more general combination of dependency and constituency relations, with applications to coordination, see Kahane, 1997 and references therein to 1960s work by Aleksej V. Gladkij.

(15)



(16)



The use of constituency to differentiate between different nestings of coordinate structures is suggested e.g. in Hudson, 1988, 318 and in Hudson, 1990, 408. On this approach, coordinate structures are not connected dependency graphs. The status of conjunctions differs in different versions of WG, but in Hudson, 1984, 1988, 1989, 1990 they are not integrated with the rest of the *dependency* structure: they are neither heads nor dependents. On the other hand, conjunctions are integrated in the *constituency* structure of coordinate structures: they are immediate constituents of coordinate structures (in Hudson, 1984, 1988, 1989 and in (14)–(16)) or of the immediately following conjuncts (in Hudson, 1990). Apart from constituency structures used to model coordination, another formal device which goes beyond simple dependency structures is that of split dependency relations. Thus, all three structures (14)–(16) should be understood as containing a single subject dependency originating in *arrived*, which splits into three edges targeting the proper nouns.

3 Nested Coordination in Enhanced UD

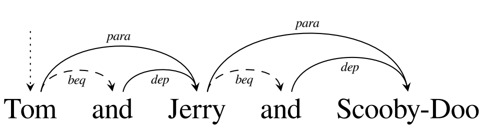
UD does not assume either constituency structure or split relations; basic UD representations are simple dependency trees, and enhanced representations are dependency graphs (with the option of introducing empty nodes). How could then the three different structures of (1) be represented in UD? In the following subsections, we consider various solutions starting from the least UD-conservative (i.e., least conservative from the point of view of UD) and moving to the most UD-conservative.

3.1 Different Topology

A theoretically possible solution would be to change the general UD topology of coordinate structures and represent them as headed by the conjunction. While this was probably the most popular representation of coordination in pre-UD treebanks (Popel et al., 2013), the idea that conjunctions head coordinate structures is widely rejected on theoretical linguistic grounds, both within dependency approaches (e.g. Mel'čuk and Pertsov, 1987, 65, Hudson, 1988, 314–315 and Gerdes and Kahane, 2015, 102–105) and within constituency approaches (Borsley, 2005). Hence, we will not consider this possibility here.

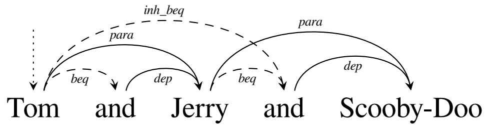
A proposal to distinguish different nestings in dependency graphs which does not assume mechanisms outside of dependency relations and in which coordination is represented as headed by the first conjunct is outlined in Gerdes and Kahane, 2015, 108. According to that proposal, the three representations of different nestings of *Tom and Jerry and Scooby-Doo* would be:[4]

(17)

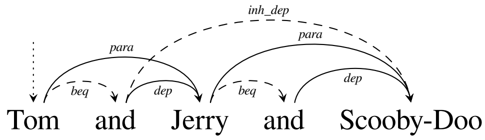


[4]See Gerdes and Kahane, 2015 for the explanation of the labels occurring in these graphs: *para(digmatic link)*, *dep(endent)*, *beq(ueather)*, *inh(erited)_dep(endent)* and *inh(erited)_beq(ueather)*; their precise meaning is not crucial here.

(18)



(19)

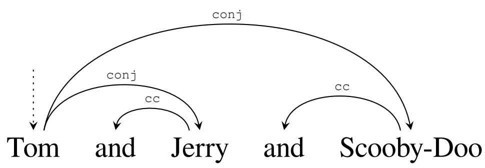


In the above graphs, primary dependencies are shown as solid lines and secondary dependencies are shown as dashed arrows. Note that, even after removing the secondary dependencies, these graphs do not become trees: all non-initial conjuncts have two incoming edges. Moreover, reversing the *dep* dependencies between conjunctions and the following conjuncts does not necessarily produce UD-like trees. The reason is that the "chain" ("Moscow", in the terminology of Popel et al., 2013) topology of (17) (representing no nesting), in which conjuncts form a dependency chain, differs from the "bouquet" ("Stanford") topology adopted in UD, in which all non-initial conjuncts dependent on the first one; and similarly for (18) (representing nesting indicated in (3)). While Gerdes and Kahane (2015) carefully justify such a representation of coordinate (and, more generally, paradigmatic) structures, the solutions presented below are more UD-conservative.

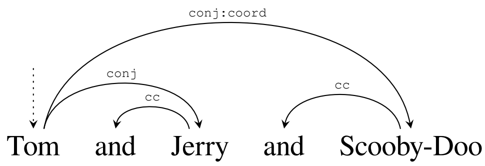
3.2 Enriching Dependency Labels

One possible solution (suggested to us by Nathan Schneider, p.c., August 2018) is to enrich dependency labels via subtyping, e.g.:

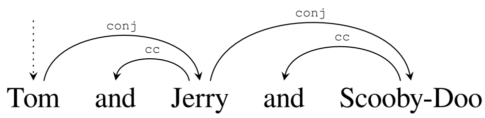
(20)



(21)



(22)

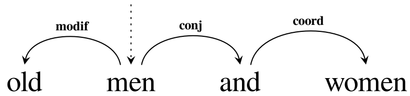


According to this idea, the flat structure is represented as before (see (20), same as (5) above), and so is the structure where the second and third nouns form a nested coordinate structure (see (22), same as (6) above). However, the structure with the first two nouns forming a nested coordinate structure is distinguished from the flat coordination with the use of the subtyped dependency relation `conj:coord`, which signals that the head of the dependency is itself a coordinate structure (see (21)).

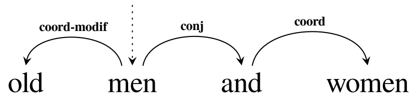
A similar solution to a related problem of distinguishing between the two readings of (23) indicated in (24)–(25) is discussed in Mel'čuk, 2009, 93–94 (and earlier, on the basis of different examples, in Mel'čuk, 1988, 99); the dependency trees corresponding to (24)–(25) are (26)–(27), respectively.

(23) old men and women
(24) [old men] and women
(25) old [men and women]

(26)



(27)



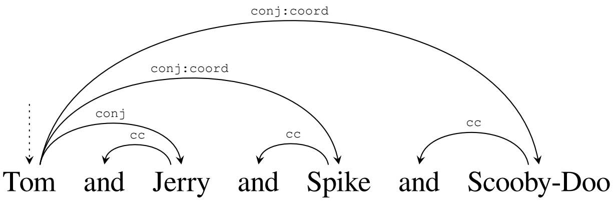
This possibility is rejected as 'highly unnatural' (Mel'čuk, 1988, 30) and as leading to the doubling of dependency labels (Mel'čuk, 2009, 94).[5] In fact, when applied to the problem of nested coordination, the `conj` label would not only have to be doubled by `conj:coord`, but – in order to represent more nested coordination – multiplied indefinitely, insofar as there are no theoretical bounds on the depths of nesting of coordinate structures. To see the problem, consider (28), two of its structures indicated in (29)–(30), and the representation in (31):

(28) Tom and Jerry and Spike and Scooby-Doo

(29) [Tom and Jerry] and Spike and Scooby-Doo

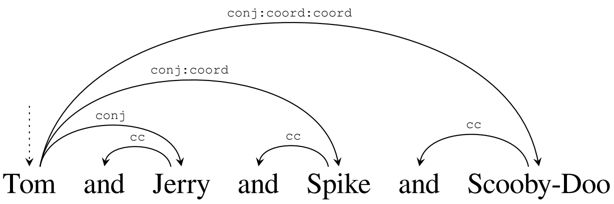
(30) [[Tom and Jerry] and Spike] and Scooby-Doo

(31)



While (31) is a reasonable representation of (29), it would also need to serve as the representation of (30), if `conj` were only allowed to be extended to `conj:coord`. The problem is that `conj:coord` represents the information that the head is embedded inside another coordination, but not the information about the number of coordinate structure boundaries that the dependency crosses. Such information could be represented by repeating the `:coord` subtype an appropriate number of times, as in (32):

(32)

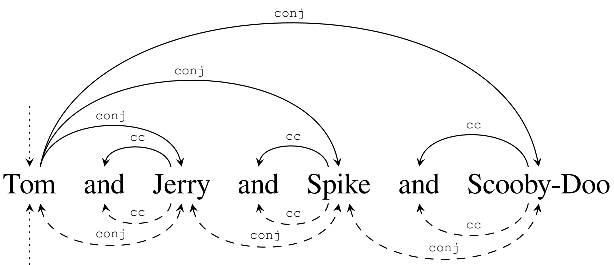


Since adopting this solution would amount to allowing for a theoretically infinite number of possible dependency labels, we reject it here, and instead present two solutions that are free from this problem.[6]

3.3 Co-Headedness of Conjuncts

Another relatively UD-conservative solution is to retain the by now standard basic tree representation of coordination in UD, and distinguish different nesting possibilities in the enhanced representation, by linking co-conjuncts in a bidirectional chain. For example, in the case of *Tom and Jerry and Spike and Scooby-Doo*, the flat structure and the two nestings indicated in (29)–(30) (repeated below) receive the following three UD representations on our proposal, with secondary edges in enhanced representations shown as dashed arrows drawn under the tokens:

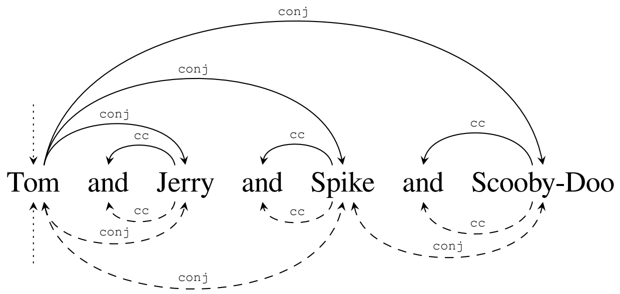
(33) Tom and Jerry and Spike and Scooby-Doo (= (28), i.e., no nesting):



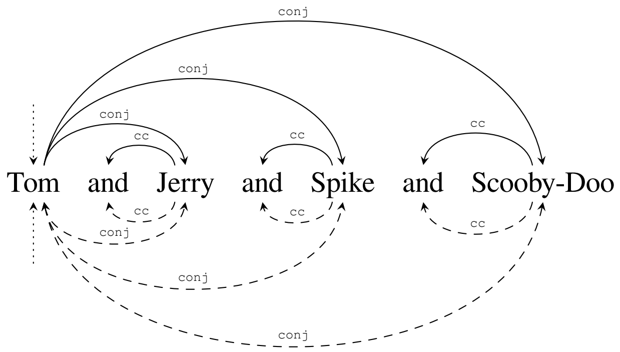
[5] Mel'čuk, 2009, 94 also claims that this solution is 'not sufficient formally' to represent the difference between 'hungry [men and women and children]' and '[hungry [men and women]] and children', but this difference can in fact be represented by the combination of Mel'čuk's suggestion concerning the scope of modifiers illustrated in (26)–(27) and its extension to the case of nested coordination discussed here.

[6] Having a limited number of dependency labels is often perceived as important by the UD community; see, e.g., Schuster et al., 2017, 130–131 for arguments against encoding paths in dependency labels in the context of the UD representation of gapping, the most important of which is that this would introduce an unbounded number of dependency relations.

(34) [Tom and Jerry] and Spike and Scooby-Doo (= (29)):



(35) [[Tom and Jerry] and Spike] and Scooby-Doo (= (30)):



According to this proposal, any two neighbouring conjuncts in the same coordinate structure are connected with a bidirectional `conj` dependency in the enhanced representation. In order to avoid introducing a new mechanism into UD, this bidirectional dependency may be understood as two usual (unidirectional) dependencies going in the opposite direction.

The practical advantage of this proposal is that, for any number of conjuncts in a coordinate structure, any two different nestings will provably differ either in their basic tree representation, or in their enhanced representation, or in both. Hence, different nestings of a coordinate structure may now be distinguished in UD; Table 1 gives all 11 possibilities for the case of 4 conjuncts.[7] This would also be true if we did not insist on the bidirectionality, but instead allowed for chaining, say, from left to right. Moreover, this solution is also compatible with the proposal of Kanayama et al. (2018), who convincingly argue for right-headed basic tree representations of coordination in the case of head-final languages such as Japanese and Korean, i.e., representations symmetric with respect to the strictly left-headed trees currently imposed by UD. In the case of such head-final languages, the unidirectional version of the enhanced representation proposed here would make more sense with chains from right to left.

The theoretical advantage which does, however, rely on bidirectionality is related to the frequently expressed (but rarely implemented) sentiment that all conjuncts are heads of a coordinate structure. This is the hallmark of the Generalized Phrase Structure Grammar (GPSG; Gazdar et al., 1985) analysis of coordination (see e.g. Gazdar et al., 1985, ch.6 and Sag et al., 1985), and it has also been defended within Head-driven Phrase Structure Grammar (HPSG; Pollard and Sag, 1987, 1994), e.g. in Abeillé, 2003. This is also a recurrent theme within dependency (and related) approaches ever since their inception: Tesnière, 1959, 2015 (chapters 136 and 143–146) contains analyses of coordinate structures as effectively multi-headed by (roots of) all conjuncts.[8] Similarly, formalisms combining dependency and constituency are sometimes motivated by the need to represent conjuncts as co-heads, cf. e.g. Kahane, 1997, §5.1 and Kahane and

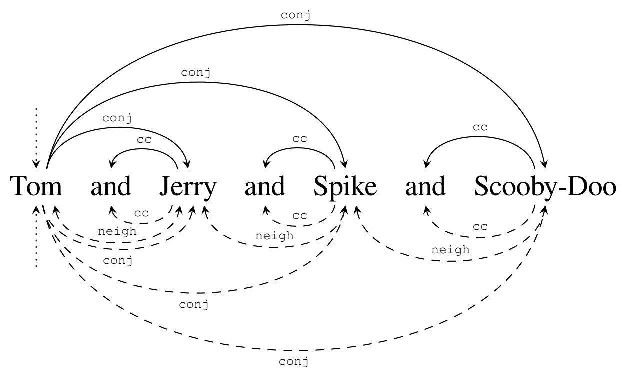
[7] The numbers of different nestings – i.e. 1 (for two conjuncts), 3 (for three conjuncts), 11 (for four conjuncts, as in Table 1), 45 (for five conjuncts), etc. – form a sequence known in combinatorics as (little) Schröder numbers, Schröder–Hipparchus numbers or super-Catalan numbers; see e.g. Stanley, 1997 for the history of these numbers, and their other interpretations. This is an exponential sequence; for example, for ten conjuncts, there are 103,049 possible nestings (as calculated already by Hipparchus of Nicaea, c. 190 – c. 120 BC).

[8] Unlike the representation in chapter 38, which suggests a WG-like analysis.

Mazziotta, 2015, 162–163.[9] Moreover, the reason often given by Hudson for the headless representation of coordination is that – according to Hudson – it is not clear which conjunct should be taken to be the head. The representation proposed here treats all conjuncts (or rather, in Hudson's terminology, conjunct-roots) as heads, even in nested coordination: each conjunct is subordinate to all other conjuncts.[10]

It should be noted that this proposal solves the nested coordination problem on the assumption that the two UD levels of representation – the basic tree and the enhanced graph – are considered simultaneously: neither of the two levels distinguishes all nestings by itself. We do not see a way of modifying the *basic* representation alone in a way that distinguishes all nestings and does not create the problem (discussed above) of an indefinite number of dependency labels. However, the obvious way for the *enhanced* representation to distinguish all nestings alone is to make it contain both: the standard UD representation of coordination and the bidirectional links between neighbouring conjuncts. Assuming such bidirectional links are labelled as `neigh` (in order to distinguish them from the standard `conj` dependencies), the resulting representation of the 'no nesting' case would be as in (36):

(36) Tom and Jerry and Spike and Scooby-Doo (no nesting, all information in enhanced representation):

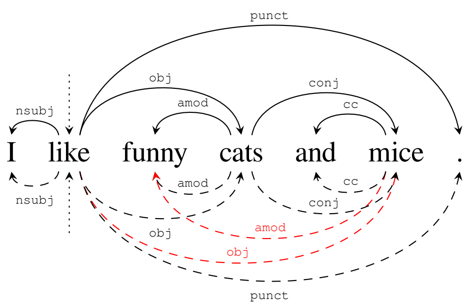


3.4 UD-Conservative Solution

Finally, there is a solution (whose initial version was suggested to us by an anonymous reviewer) that does not have the theoretical advantage of encoding co-headedness of conjuncts, but has the practical advantage of being maximally UD-conservative. As in the previous representation, the basic tree representation of coordination is left intact, but the enhanced representation is a proper extension of this basic tree according to *some* of the general enhanced UD principles of representing coordination. To recall these principles, consider (37) and its UD representation – on the reading where both cats and mice are funny – in (38).

(37) I like funny cats and mice.

(38)



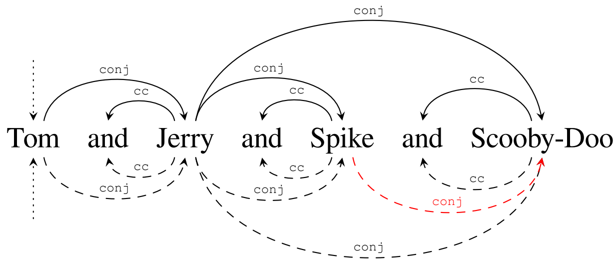
As illustrated in (38), dependencies involving the whole coordinate structure are represented on the head of the coordinate structure in the basic tree, but on all conjuncts in the enhanced graph. This concerns both incoming and outgoing dependencies: the `obj` dependency from the verb targets all conjuncts in the enhanced representation and the `amod` dependency to the adjective originates from all conjuncts.

[9]Unfortunately, a more detailed comparison – requested by a reviewer – of the current proposal to previous work of Sylvain Kahane, Nicolas Mazziotta and Kim Gerdes carried out within such combined approaches is impossible within the limits of this paper.

[10]Recall that the subordination relation is the reflexive transitive closure of the dependency relation.

Now, a representation of different nestings is possible in the enhanced graph if the second – and only the second – way of distributing dependencies among conjuncts is adopted, i.e., if dependents of a coordinate structure are represented as headed by each conjunct. This is illustrated with (39):

(39) Tom and [[Jerry and Spike] and Scooby-Doo]:



Since *Scooby-Doo* is a `conj` dependent of the coordinated *Jerry and Spike*, the enhanced representation adds the `conj` dependency from *Spike* to *Scooby-Doo*. However, while the coordination *Jerry and Spike and Scooby-Doo* is a dependent of *Tom*, there are no additional edges from *Tom* to any non-initial conjuncts within *Jerry and Spike and Scooby-Doo*.

Just as in the case of the proposal of the previous subsection, it may be demonstrated that, for any number of conjuncts, this representation distinguishes between different nestings. Here, we illustrate all possibilities for the case of 4 conjuncts – see Table 2. Unlike in the case of the initial proposal of the previous subsection, different nestings may be distinguished on the basis of a single level of representation – the enhanced graph.

It is important to realise that this solution only works if just one part of the general UD approach to distributing coordination in enhanced representation is adopted: what is distributed among conjuncts are dependencies *from* coordinate structures, not dependencies *to* coordinate structures. As is easy to verify, the other two possibilities fail already in the case of 3 conjuncts: adopting both parts, i.e., also distributing dependencies *to* coordinate structures, fails to distinguish (3) from (4), while adopting only the latter part fails to distinguish (2) from (3). This might be construed as an additional argument for modifying the general UD rules of distribution in coordinate structures: distributing dependencies targeting coordinate structure is sometimes (p.c. to members of UD community) considered potentially confusing for downstream applications relying on enhanced representations; e.g., in the case of (37), looking just at the dependencies headed by the verb *like* in (38) suggests that there are two separate direct objects. Allowing only for the distribution of dependencies *from* coordinate structures would remove this problem and make the current proposal just a straightforward application of general UD principles.

4 Conclusion

While nested coordination is currently a problem for Universal Dependencies, a number of more or less conservative modifications rectifying this situation are in sight, including the possibility to enrich dependency labels in a way that indicates the nestings involved (cf. §3.2). However, we propose two solutions which do not lead to such a – theoretically unbounded – proliferation of labels and which fully preserve the current UD representation of coordination at the level of basic trees. The theoretical advantage of the solution presented in §3.3 is that it encodes the intuition – common in various linguistic approaches – that, in a sense, each conjunct is a head of a coordinate structure, but its practical disadvantage is that the enhanced representation is very different from current UD representations of coordination. The solution in §3.4 is much more UD-conservative, as the enhanced representation is a superset of the basic tree and it is constructed in full compliance with already existing UD mechanisms. Adopting either of these solutions would remove this embarrassing flaw in the current version of Universal Dependencies.

Acknowledgements

For useful comments, which led to numerous improvements of this paper, we wish to thank Tim Osborne, anonymous SyntaxFest 2019 reviewers, and the audience of the 17th International Workshop on Treebanks and Linguistic Theory (University of Oslo, December 2018). Agnieszka Patejuk gratefully acknowledges the *Mobilność Plus* mobility grant awarded to her by the Polish Ministry of Science and Higher Education.

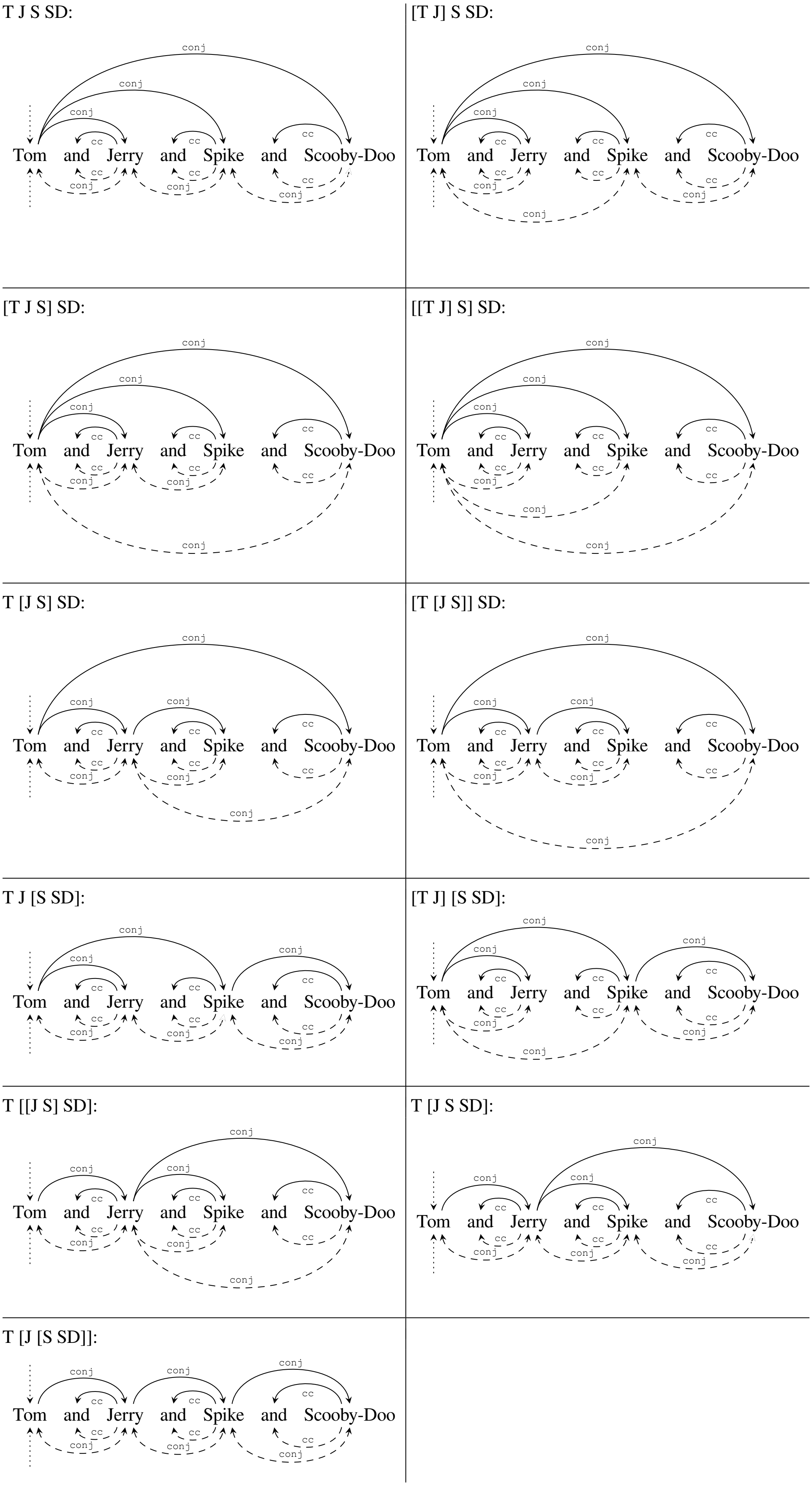


Table 1: Possible nestings of *Tom and Jerry and Spike and Scooby-Doo* according to §3.3

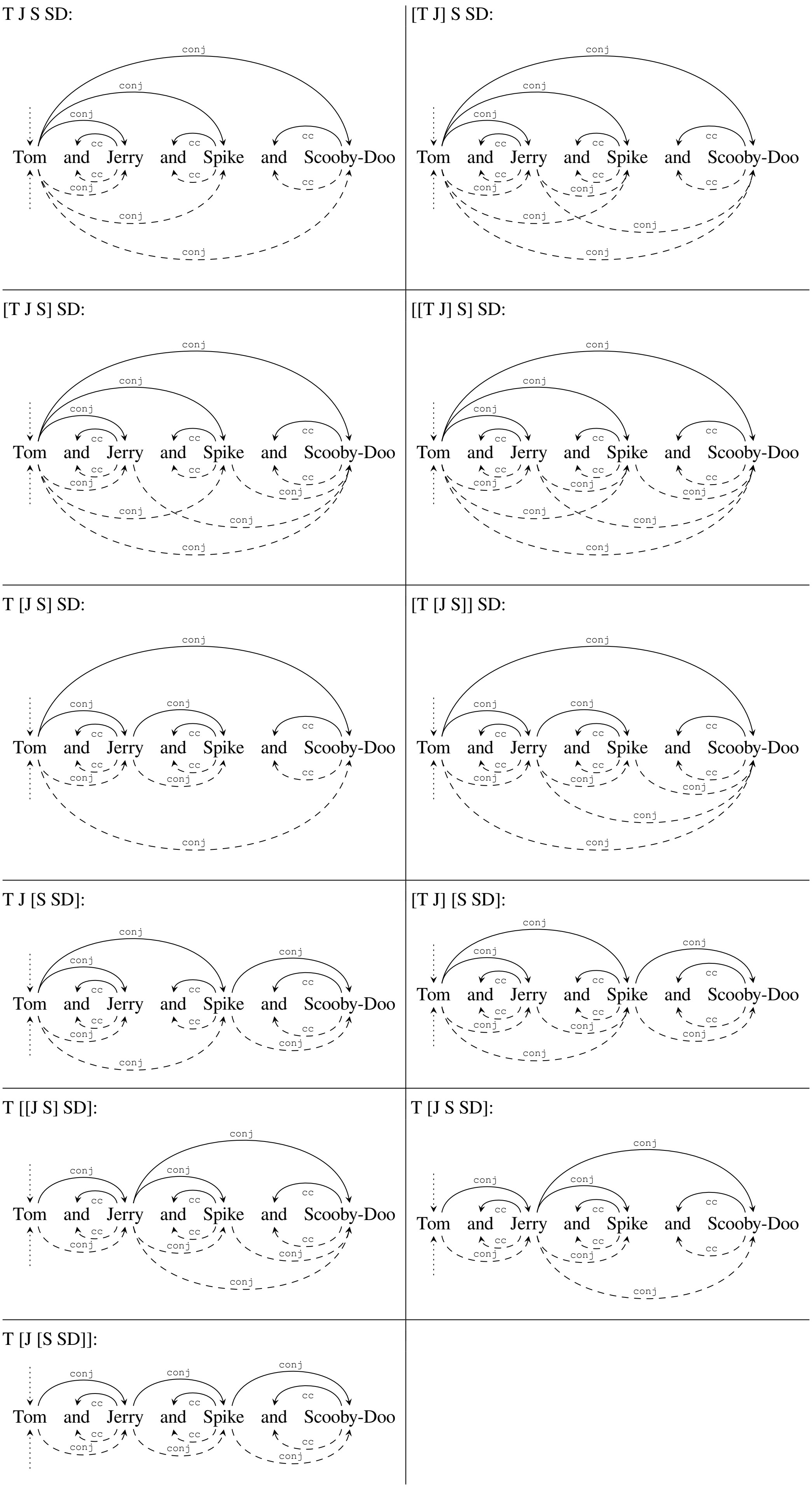


Table 2: Possible nestings of *Tom and Jerry and Spike and Scooby-Doo* according to §3.4

References

Anne Abeillé. 2003. A lexicon- and construction-based approach to coordinations. In Stefan Müller, editor, *Proceedings of the HPSG 2003 Conference*, pages 5–25, Stanford, CA: CSLI Publications.

Robert D. Borsley. 2005. Against ConjP. *Lingua* 115(4):461–482.

Gerald Gazdar, Ewan Klein, Goeffrey K. Pullum, and Ivan A. Sag. 1985. *Generalized Phrase Structure Grammar*. Cambridge / Cambridge, MA: Blackwell / Harvard University Press.

Kim Gerdes and Sylvain Kahane. 2015. Non-constituent coordination and other coordinative constructions as dependency graphs. In Eva Hajičová and Joakim Nivre, editors, *Proceedings of the Third International Conference on Dependency Linguistics (DepLing 2015)*, pages 101–110, Uppsala.

Richard Hudson. 1980. A second attack on constituency: A reply to Dahl. *Linguistics* 18:489–504.

Richard Hudson. 1984. *Word Grammar*. Oxford: Blackwell.

Richard Hudson. 1988. Coordination and grammatical relations. *Journal of Linguistics* 24(2):303–342.

Richard Hudson. 1989. Gapping and grammatical relations. *Journal of Linguistics* 25(1):57–94.

Richard Hudson. 1990. *English Word Grammar*. Oxford: Blackwell.

Richard Hudson. 2018. HPSG and Dependency Grammar, to appear in the HPSG handbook published by the Language Science Press.

Sylvain Kahane. 1997. Bubble trees and syntactic representations. In Tilman Becker and Hans-Ulrich Krieger, editors, *Proceedings of Mathematics of Language 5*, pages 70–76, DFKI, Saarbrücken.

Sylvain Kahane and Nicolas Mazziotta. 2015. Syntactic polygraphs: A formalism extending both constituency and dependency. In Marco Kuhlmann, Makoto Kanazawa, and Gregory M. Kobele, editors, *Proceedings of Mathematics of Language 14*, pages 152–164, Chicago, IL.

Hiroshi Kanayama, Na-Rae Han, Masayuki Asahara, Jena D. Hwang, Yusuke Miyao, Jinho D. Choi, and Yuji Matsumoto. 2018. Coordinate structures in Universal Dependencies for head-final languages. In Marie-Catherine de Marneffe, Teresa Lynn, and Sebastian Schuster, editors, *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 75–84, Association for Computational Linguistics.

Igor Mel'čuk. 1964. *Avtomatičeskij sintaksičeskij analiz. Tom I: Obščie principy. Vnutrisegmentnyj sintaksičeskij analiz*. Novosibirsk: Nauka.

Igor Mel'čuk. 1974. *Opyt teorii lingvističeskix modelej «Smysl ⇔ Tekst»*. Moscow: Nauka.

Igor Mel'čuk. 1988. *Dependency Syntax: Theory and Practice*. Albany, NY: The SUNY Press.

Igor Mel'čuk. 2009. Dependency in natural language. In Alain Polguère and Igor Mel'čuk, editors, *Dependency in Linguistic Description*, pages 1–110, Amsterdam: John Benjamins.

Igor Mel'čuk and Nikolaj Pertsov. 1987. *Surface Syntax of English. A Formal Model within the Meaning–Text Framework*. Amsterdam: John Benjamins.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation, LREC 2016*, pages 1659–1666, ELRA, Portorož, Slovenia: European Language Resources Association (ELRA).

Timothy Osborne. 2006a. Gapping vs. non-gapping coordination. *Linguistische Berichte* 207:307–337.

Timothy Osborne. 2006b. Shared material and grammar: Toward a Dependency Grammar theory of non-gapping coordination for English and German. *Zeitschrift für Sprachwissenschaft* 25:39–93.

Timothy Osborne and Thomas Groß. 2017. Left node blocking. *Journal of Linguistics* 53(3):641–688.

Agnieszka Patejuk and Adam Przepiórkowski. 2018. *From Lexical Functional Grammar to Enhanced Universal Dependencies: Linguistically Informed Treebanks of Polish*. Warsaw: Institute of Computer Science, Polish Academy of Sciences.

Martin Pickering and Guy Barry. 1993. Dependency categorical grammar and coordination. *Linguistics* 31:855–902.

Carl Pollard and Ivan A. Sag. 1987. *Information-Based Syntax and Semantics, Volume 1: Fundamentals*. CSLI Lecture Notes, No. 13, Stanford, CA: CSLI Publications.

Carl Pollard and Ivan A. Sag. 1994. *Head-driven Phrase Structure Grammar*. Chicago, IL: Chicago University Press / CSLI Publications.

Martin Popel, 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 (Volume 1: Long Papers)*, pages 517–527, Sofia, Bulgaria.

Adam Przepiórkowski and Agnieszka Patejuk. 2019. From Lexical Functional Grammar to enhanced Universal Dependencies: The UD-LFG treebank of Polish, to appear in *Language Resources and Evaluation* (published on-line on 4 February 2019).

Ivan A. Sag, Gerald Gazdar, Thomas Wasow, and Steven Weisler. 1985. Coordination and how to distinguish categories. *Natural Language and Linguistic Theory* 3:117–171.

Sebastian Schuster, Matthew Lamm, and Christopher D. Manning. 2017. Gapping constructions in Universal Dependencies v2. In Marie-Catherine de Marneffe, Joakim Nivre, and Sebastian Schuster, editors, *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 123–132, Association for Computational Linguistics, Gothenburg, Sweden.

Richard P. Stanley. 1997. Hipparchus, Plutarch, Schröder, and Hough. *The American Mathematical Monthly* 104(4):344–350.

Lucien Tesnière. 1959. *Éléments de Syntaxe Structurale*. Paris: Klincksieck.

Lucien Tesnière. 2015. *Elements of Structural Syntax*. Amsterdam: John Benjamins.