University of Toronto, Department of Computer Science
CSC 2501F—Computational Linguistics, Fall 2024

Reading assignment 4

Due date: Electronically by noon, Friday, 15 November 2024.
Late write-ups will not be accepted without documentation of a medical or other emergency.
This assignment is worth 3% of your final grade.

What to read

Buder-Gröndahl, T. (2024) What Does Parameter-free Probing Really Uncover? *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pp. 327–336.

What to write

Write a *brief* summary of this paper, with a critical assessment of its merits.

Some points to consider:

- What is grammar?
- What are some problems with probing as an interpretative tool?
- Are there any alternatives to probing for interpretation?

General requirements: Your write-up should be typed, using 12-point font and 1.5-line spacing; it should fit on one to two sides of a sheet of paper.

Submission is electronic, through Quercus.

What does Parameter-free Probing Really Uncover?

Tommi Buder-Gröndahl
University of Helsinki / Yliopistonkatu 3, 00014 Helsinki, Finland
tommi.grondahl@helsinki.fi

Abstract

Supervised approaches to probing large language models (LLMs) have been criticized of using pre-defined theory-laden target labels. As an alternative, *parameter-free probing* constructs structural representations bottom-up via information derived from the LLM alone. This has been suggested to capture a genuine "LLM-internal grammar". However, its relation to familiar linguistic formalisms remains unclear. I extend prior work on a parameter-free probing technique called *perturbed masking* applied to BERT, by comparing its results to the Universal Dependencies (UD) formalism for English. The results highlight several major discrepancies between BERT and UD, which lack correlates in linguistic theory. This raises the question of whether human grammar is the correct analogy to interpret BERT in the first place.

1 Introduction

Probing large language models (LLMs) consists in mapping their internal states to linguistic classes or relations (Rogers et al., 2020; Belinkov, 2022). Most methods use supervised learning for training a probe to predict pre-determined labels (Hewitt and Manning, 2019; Tenney et al., 2019; Kuznetsov and Gurevych, 2020; Manning et al., 2020; Lasri et al., 2022). However, critics have deemed this insufficient for determining whether LLMs actually *represent* linguistic structures (Kulmizev and Nivre, 2022; Buder-Gröndahl, 2023). For representation proper, the labels should not only be predictable from the LLM; they should somehow capture its internal architecture on a high level of abstraction.

A possible way forward is to use *parameter-free probing*, which shuns separate probing classifiers by extracting structural information directly from the LLM (Clark et al., 2019; Mareček and Rosa, 2019; Wu et al., 2020). As a bottom-up approach, this has been interpreted as uncovering the grammar intrinsic to the LLM without relying on *a priori* presumptions derived from linguistic theory.

In this paper, I focus on a parameter-free probe called *perturbed masking*, originally presented and applied to BERT by Wu et al. (2020). While it has received criticism for underwhelming results compared to gold-standard parses (Niu et al., 2022), this overlooks its main goal of uncovering BERT's inherent syntax – which may well deviate from linguistic theory (Wu et al., 2020, 4173). Such deviations do not call for discarding it; instead, they provide insight into how BERT's architecture can differ from common linguistic assumptions.

I compare dependency graphs derived from BERT to the Universal Dependencies (UD) annotation for English, and uncover major discrepancies related to verbal argument structure, noun phrase structure, modifiers, and prepositions. In particular, BERT treats the *root* (in UD's annotation) as a head far more often than UD. This effect of being "attracted by the root" is especially strong in recursive embeddings, but also extends beyond these.

Moreover, BERT's behavior tends to resist linguistic explanation. For example, despite major disagreements within linguistic theory, argument structure is ubiquitously treated as clause-bound: no feasible analysis assimilates embedded clause arguments to main clause arguments. Yet, the BERT-parse regularly does exactly this. Indeed, the only cases where BERT's deviations from UD have a salient linguistic interpretation concern prepositions and some possessive constructions, where dependent-head relations are flipped.

The results thus point to the same direction as critiques of supervised probing: the assumption that BERT represents grammar in line with familiar linguistic formalisms lacks proper support. When this is not built directly into the experiment design (via pre-determined target labels), probing reveals fundamental disparities between BERT and commonly accepted syntactic principles. We are thus prompted to question whether human grammar is an appropriate analogy for BERT after all.

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 327–336

2 Methodology

I describe the parameter-free probing technique investigated (Section 2.1), the dataset (Section 2.2), and the experiment pipeline (Section 2.3).

2.1 Perturbed masking

Parameter-free probing aims to construct linguistic information directly from the LLM without separate training. Wu et al. (2020) present a prominent techique called *perturbed masking*, with which they aim to find "the 'natural' syntax inherent in BERT" (p. 4173) by utilizing an independently motivated relation of *impact* between tokens. I replicated their original setup,[1] which uses the *bert-base-uncased* model presented in Wolf et al. (2020).

As input, BERT takes a sequence of tokens $\mathbf{x} = [x_1, ..., x_n]$. It maps each token x_i to a contextual representation $H_\theta(\mathbf{x})_i$, where the influence of each token $x_j \in \mathbf{x}$ arises via Transformer attention (Vaswani et al., 2017) based on model parameters θ. For perturbed masking, Wu et al. (2020) first mask token x_i, giving $\mathbf{x}\backslash\{x_i\}$. They then also mask token x_j, giving $\mathbf{x}\backslash\{x_i, x_j\}$. The *impact* of x_j to the representation of x_i is now measured as follows, where d is Euclidean distance:[2]

$$f(x_i, x_j) = d(H_\theta(\mathbf{x}\backslash\{x_i\})_i, H_\theta(\mathbf{x}\backslash\{x_i, x_j\})_i)$$

Impacts between all token pairs are collected into an *impact matrix*, which is given as input to an algorithm that constructs a directed dependency graph using the *Eisner* algorithm (Eisner, 1996).[3] The intuitive idea is that heads have the highest impact on their dependents in the matrix.

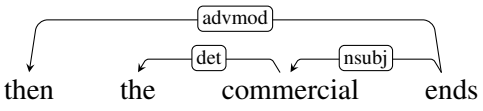
2.2 Data

Following Wu et al. (2020), I used the English Parallel Universal Dependencies (PUD) dataset (Zeman et al., 2017). Consisting of 1000 sentences of which I discarded seven (see Appendix A), it covers 21047 UD-annotated tokens.

2.3 Experiments

UD assigns each word a *head* and a dependency relation type (*deprel*), as exemplified below:[4]

(1)

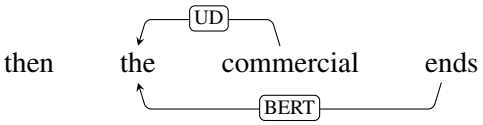


The arrow is read as marking a head-dependent relation (in this direction). The *root* is its own head, and is typically the main verb. The BERT-parse of the same sentence maps all tokens to the root *ends*:

(2)

Here, UD and BERT differ in which head they assign to the determiner *the*. I denote this by marking the UD-assigned head-dependent relation above and the BERT-assigned relation below:

(3)



The challenge in interpreting BERT-parses is that they only give head-dependent relations, not deprels. We thus need external deprels as the theoretical basis of comparing BERT and UD. For this, I use UD-annotations as follows:

$Dep(x)$: deprel assigned to x by UD
$Head_{UD}(x)$: head assigned to x by UD
$Head_{BERT}(x)$: head assigned to x by BERT
$H_U(x) = Dep(Head_{UD}(x))$
$H_B(x) = Dep(Head_{BERT}(x))$

That is, I compare UD- and BERT-assigned heads in terms of their UD-deprels. These values for the determiner in the example above are:

$Dep(\textit{the}) = \textit{det}$
$Head_{UD}(\textit{the}) = \textit{commercial}$
$Head_{BERT}(\textit{the}) = \textit{ends}$
$H_U(\textit{the}) = Dep(\textit{commercial}) = \textit{nsubj}$
$H_B(\textit{the}) = Dep(\textit{ends}) = \textit{root}$

Note that, since Dep is derived from UD, H_B should not be read as directly describing how BERT treats the head. Instead, it describes *how UD would treat the head assigned by BERT*.

By classifying discrepancies between BERT and UD, I assess their prevalence and nature in the PUD data. I focus on four phenomena: argument structure, noun phrase (NP) structure, adjective/adverb modifiers, and prepositional phrases (PPs). Sourcecode for the experiments is openly available.[5]

[1] https://github.com/LividWo/Perturbed-Masking#dependency

[2] Wu et al. (2020) report superior performance to Euclidean distance compared to the difference between probability distributions across targets.

[3] Wu et al. (2020) also experimented with phrase-structures, but the present setup requires dependency graphs to obtain deprel labels (Section 2.3). See Niu et al. (2022) on phrase-structures generated via perturbed masking.

[4] All examples are taken from the PUD dataset (shortened).

[5] https://github.com/tombgro/parameter-free-probing

3 Results

I replicated the original results of Wu et al. (2020) with their best setup on the PUD data,[6] and investigated shifts between BERT and UD in terms of Dep, H_U, and H_B. Section 3.1 presents general findings, Sections 3.2–3.5 cover linguistic details, and Appendix B provides the raw data.

3.1 General findings

Of all 21047 tokens, 58% were subject to a head-dependent shift between UD and BERT. Nearly all Dep-types were involved here, and a clear majority (74%) had a shift ratio over 50%. Clearly the most common H_B was *root*; i.e. shifts typically involved BERT assigning a head which was the root in the UD-parse. This covered 35% of all shifts.

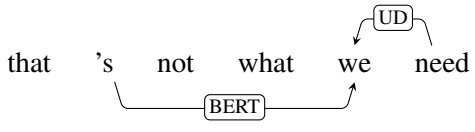
Wu et al. (2020, 4169) suggest that BERT mostly learns local dependencies. To assess this, we calculated dependent-head distances from both parses, and obtained contrasting results: the average is higher in BERT (3.66) than in UD (3.52). Locality thus does not explain the discrepancies. A likely explanation for the increased average dependent-head distance in BERT is its tendency to over-assign the root as a head. As covered in upcoming sections, this can lead to longer dependent-head distances in cases like embedded clauses, where the original UD-head is closer to its dependent than the root.

3.2 Argument structure

Table 1 collects shifts per Dep–H_U pair for active and passive clause subjects (*nsubj*, *nsubj:pass*) and direct objects (*obj*).[7]

In arguments of the root, BERT and UD mostly overlap with shift ratios of $15\% - 29\%$. However, with embedded clauses (*ccomp*, *xcomp*, *conj*, *acl:relcl*), BERT regularly continues to assign arguments to the root, with far higher shift ratios ($64\% - 94\%$) and *root* as the most common H_B. An example is shown below, where BERT assigns the main verb as the head of an embedded subject:

(4)



The BERT-parse thus seems to *shun recursion*, preferring the root even for embedded arguments.

[6] This gives the Unlabeled Attachment Score (UAS) of 41.7, the Undirected UAS (UUAS) of 52.1, and the Neutral Edge Direction (NED) score of 69.6.

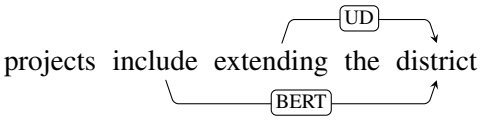
[7] Tables 1–4 contain shifts with the minimum count of 20. "Ratio" denotes the frequency of shifts for each Dep–H_U pair.

Dep	H_U	Ratio	Count
nsubj	root	0.24	198
	acl:relcl	0.81	140
	ccomp	0.92	101
	advcl	0.79	80
	conj	0.83	68
	parataxis	0.64	46
nsubj:pass	root	0.29	38
	acl:relcl	0.94	32
	advcl	0.91	21
obj	advcl	0.66	86
	xcomp	0.75	82
	acl:relcl	0.78	58
	conj	0.66	58
	acl	0.73	52
	root	0.15	47
	ccomp	0.73	29

Table 1: Verbal argument structure: subjects and objects.

The same pattern also repeats for objects:

(5)

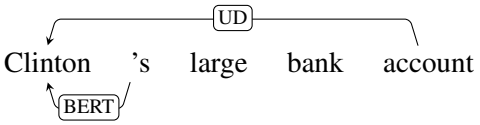


While the explanation of this behavior is not fully clear, in general it shows that the root has an especially high impact for determining the contextual embeddings of other words. One salient possibility is that this arises because the root is usually a main clause verb, which has central influence on both grammatical matters (such as inflection or valency) and semantic matters (such as the possible semantic classes of arguments). Hence, when BERT is pre-trained via masked-token prediction (Devlin et al., 2019), attending to the main clause verb is likely to give useful information pertaining to many masked tokens. A general high impact for the root would follow, in line with these findings.

3.3 Noun phrase structure

Table 2 lists NP-related shifts for three variants of Dep: determiners (*det*), possessors (*nmod:poss*), and numerals (*nummod*). Some of these shifts are grammatically salient: for instance, UD treats the possessor as headed by the possessed noun, but BERT often takes it to be headed by the clitic *'s*:

(6)

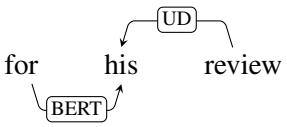


Dep	H_U	Ratio	Count
det	obl	0.52	261
	obj	0.67	253
	nsubj	0.54	208
	nmod	0.49	191
	conj	0.57	44
	nsubj:pass	0.54	43
	nmod:poss	0.64	23
	appos	0.68	21
nmod:poss	obj	0.70	56
	nmod	0.72	55
	obl	0.58	54
	nsubj	0.70	53
nummod	obl	0.69	55
	nmod	0.71	25

Table 2: Determiners, possessors, and numerals.

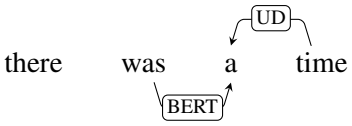
However, many cases are linguistically incoherent. For example, BERT sometimes takes possessors to modify a preposition rather than a noun:

(7)



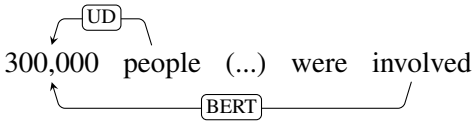
As usual, BERT also regularly assigns the root as the head, as for the determiner (*a*) shown here:

(8)



In principle, the *DP-analysis* in formal linguistics treats determiners as noun phrase heads (Abney, 1987), and might initially justify taking the determiner to head the object (*a time*). However, this would require the noun (*time*) to be headed by the determiner, but instead it is headed by the root as well. BERT thus does not implement the DP-analysis; the determiner is simply attracted by the root. The same occurs for numeral modifiers:

(9)



Since possessors, determiners, and numerals are the *sine qua non* of NP-arguments/modifiers, these results illustrate a drastic shift between BERT and widely shared syntactic assumptions about NPs.

3.4 Adjective and adverb modifiers

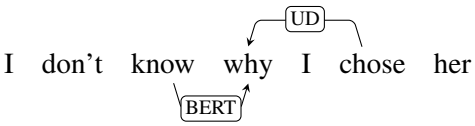
Table 3 shows shifts related to adjectives (*amod*), adverbs (*advmod*), and nominal modifiers (*nmod*).

Dep	H_U	Ratio	Count
amod	obj	0.62	151
	obl	0.52	151
	nmod	0.53	132
	nsubj	0.53	118
	conj	0.63	56
	nsubj:pass	0.52	29
	compound	0.57	21
advmod	root	0.18	57
	conj	0.62	53
	advcl	0.72	51
	acl:relcl	0.73	40
	amod	0.73	36
	advmod	0.71	32
	nummod	0.75	27
	ccomp	0.68	27
	obl	0.72	21
	xcomp	0.72	21
nmod	obl	0.88	243
	obj	0.89	202
	nsubj	0.87	163
	nmod	0.84	127
	conj	0.88	59
	nsubj:pass	0.83	34
	appos	0.85	23
	root	0.38	20

Table 3: Adjectival, adverbial, and nominal modifiers.

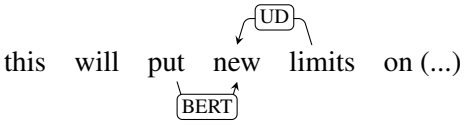
The root is a prominent H_B in embedded clauses as well as nested modifiers, indicating that BERT does not reliably treat modifiers recursively. For example, embedded *wh*-adverbs such as *why* are often assigned as dependents of the main verb:

(10)



However, the lack of recursion is insufficient to explain all modifier-related shifts. In particular, adjectives of even non-embedded noun phrases are regularly treated as dependents of the root:

(11)



This behavior resists interpretation in all prominent syntactic frameworks on adjectives, which ubiquitously treat them as modifiers of nouns or NPs (c.f. Baker 2003; Dixon 2004; Hofherr and Matushansky (ed.) 2010).

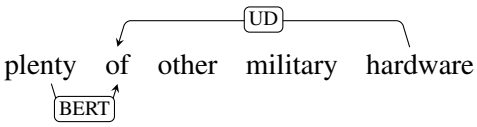
Dep	H_U	Ratio	Count
case	obl	0.72	877
	nmod	0.73	783
	nmod:poss	0.83	85
obl	root	0.47	283
	acl:relcl	0.97	117
	advcl	0.95	92
	conj	0.91	90
	xcomp	0.95	89
	acl	0.93	88
	ccomp	0.96	50
	parataxis	0.96	25

Table 4: Prepositional phrases.

3.5 Prepositional phrases

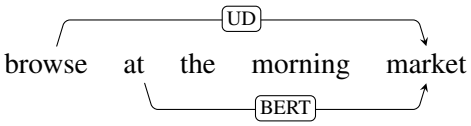
Table 4 collects shifts related to prepositions or clitics (*case*) and their complements (*obl*). BERT regularly treats prepositions as dependents of the token modified by the PP, while UD takes them to be headed by the complement noun:

(12)



BERT also regularly treats the complement as the preposition's dependent, in contrast to UD linking it directly to the token modified by the PP:

(13)



This is especially interesting since here UD prefers the root as opposed to BERT, unlike in our other findings. It thus looks like a genuine syntactic difference. However, the pattern is no longer reliable when the PP modifies a non-root, as shown by the high shift ratios with embedded clauses as H_U. The most prominent H_B here was again *root*.

3.6 Summary

I draw four take-home messages:

1. The root is treated as a head far more by BERT than by UD, even across phrase boundaries.
2. BERT's overlap with UD drastically decreases in embeddings, displaying a lack of recursion.
3. Headedness in PPs is systematically flipped between UD and BERT.
4. Overall, BERT-parses commonly lack a coherent linguistic interpretation.

4 Discussion

The results are not easily explained by some trivial non-linguistic property. Locality does not account for BERT's deviations from UD, since the average head-dependent distance is actually higher in BERT-parses (Section 3.1). Another initial possibility could be that BERT mimics naive right-chain performance.[8] However, most examples in Sections 3.2–3.5 involve BERT assigning the head *leftward* (i.e. the dependent rightward). Sometimes this even goes directly against right-chain-like annotation in UD, as in example (11) (Section 3.4).

It is also worth raising the controversial status of the UD format itself (c.f. Rehbein et al. 2017; Osborne and Gerdes 2019). The central issue here concerns function words, which UD treats as dependents of content words – going against alternative formats such as *Surface-syntactic Universal Dependencies* (SUD) (Gerdes et al., 2018) where these relations are reversed. The corresponding distinction appears in our results as well, with respect to prepositions and NPs (Section 3.5). BERT's performance might thus accord better alternative formats to UD, such as SUD.

That said, most discrepancies discussed in Section 3 are not specific only to UD. All mainstream syntactic frameworks distinguish between arguments/modifiers of main and embedded clauses (Sections 3.2, 3.4), and treat possessors, determiners, numerals, or adjectives as modifying nouns rather than verbs (Sections 3.3, 3.4). With the possible exception of (root-modifying) PPs (Section 3.5), the shifts are not made linguistically coherent by minor changes to the syntactic formalism.

5 Conclusions and future work

This study uncovered several discrepancies between BERT and UD. While some were syntactically interpretable, BERT's prevailing tendency to treat the root as a head across phrase boundaries lacks a clear linguistic analogy. This puts to question the idea that BERT should be interpreted in line with traditional grammatical formalisms. Instead, it highlights the need to explain LLMs in their own terms – avoiding reliance on *a priori* linguistic assumptions not motivated by LLMs themselves.

[8] Wu et al. (2020) report a 35.0 UAS for the naive right-chain baseline in comparison to the 41.7 UAS for BERT. A related issue concerns the comparison between BERT-derived phrase-structures and a naive right-branching baseline, the similarity between which is covered by Niu et al. (2022).

Limitations

This short paper focused on one model architecture (BERT), one parameter-free probing technique (perturbed masking), and one English dataset (PUD). Extending the work to cover multiple variants of each is an important future prospect. I would especially highlight the importance of inter-lingual comparison, as well as more careful attention to assumptions behind the linguistic formalism.

Methodologically, this study combined quantitative and qualitative analysis, both of which have limitations. Numerical information alone (in Tables 1–4) is insufficient for yielding thorough syntactic details on dependent-head shifts. For obtaining such further analyses, specific parse-pairs between BERT and UD need to be assessed, which is how the example cases were attained. But – as manual work – this is bound to have a smaller coverage. Without seeing any easy way out of this trade-off, I emphasize the need for further work extending both quantitative and qualitative coverage of related phenomena. I hope to have provided a fruitful starting-point for this line of research.

Ethics Statement

Prior source code and data used in the experiments is available as open-source, and the link is given in the paper (Section 2.1). No privacy-sensitive or otherwise harmful data was used, and no experiments on humans or non-human animals were conducted. The source code of the experiments is made available as open-source (Section 2.3).

Acknowledgements

I thank Jörg Tiedemann and Timothee Mickus for helpful discussions related to the paper. This project was funded by the Academy of Finland (decision number 350775).

References

Steven Abney. 1987. *The English Noun Phrase in its Sentential Aspect*. PhD thesis, Massachusets Institute of Technology.

Mark Baker. 2003. *Lexical Categories*. Cambridge University Press, Cambridge.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Tommi Buder-Gröndahl. 2023. The ambiguity of BERTology: what do large language models represent? *Synthese*, 203:15.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? An analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 4171–4186.

Robert M. W. Dixon. 2004. Adjective classes in typological perspective. In Robert M. W. Dixon and Alexandra Y. Aikhenvald, editors, *Explorations in Linguistic Typology 1*, pages 1–49. Oxford University Press, New York.

Jason M. Eisner. 1996. Three new probabilistic models for dependency parsing: An exploration. In *Proceedings of the 16th conference on Computational linguistics: Volume 1*, pages 340–345.

Kim Gerdes, Bruno Guillaume, Sylvain Kahane, and Guy Perrier. 2018. SUD or surface-syntactic Universal Dependencies: An annotation scheme near-isomorphic to UD. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 66–74.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138.

Patricia Cabredo Hofherr and Ora Matushansky (ed.). 2010. *Adjectives: Formal Analyses in Syntax and Semantics*. John Benjamins, Amsterdam.

Artur Kulmizev and Joakim Nivre. 2022. Schrödinger's tree—on syntax and neural language models. *Frontiers in Artificial Intelligence*, 5.

Ilia Kuznetsov and Iryna Gurevych. 2020. A matter of framing: The impact of linguistic formalism on probing results. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 171–182.

Karim Lasri, Tiago Pimentel, Alessandro Lenci, Thierry Poibeau, and Ryan Cotterell. 2022. Probing for the usage of grammatical number. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics Volume 1: Long Papers*, pages 8818–8831.

Christopher D. Manning, Kevin Clark, and John Hewitt. 2020. Emergent linguistic structure in artificial neural networks trained by self-supervision. *PNAS*, 117(48):30046–30054.

David Mareček and Rudolf Rosa. 2019. From balustrades to Pierre Vinken: Looking for syntax in transformer self-attentions. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 263–275.

Jingcheng Niu, Wenjie Lu, Eric Corlett, and Gerald Penn. 2022. Using Roark-Hollingshead distance to probe BERT's syntactic competence. In *Proceedings of the Fifth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 325–334.

T. Osborne and K. Gerdes. 2019. The status of function words in dependency grammar: A critique of universal dependencies (UD). *Glossa*, 4(1):17.

I. Rehbein, J. Steen, B. Do, and Anette Frank. 2017. Universal dependencies are hard to parse – or are they? In *Proceedings of the Fourth International Conference on Dependency Linguistics*, pages 218–228.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhins. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing*, pages 6000–6010.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Zhiyong Wu, Yun Chen, Ben Kao, and Qun Liu. 2020. Perturbed masking: Parameter-free probing for analyzing and interpreting BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4166–4176.

Daniel Zeman, Martin Popel, Milan Straka, Jan Hajič, Joakim Nivre, Filip Ginter, Juhani Luotolahti, Sampo Pyysalo, Slav Petrov, Martin Potthast, Francis Tyers, Elena Badmaeva, Memduh Gokirmak, Anna Nedoluzhko, Silvie Cinková, Jan Hajič jr., Jaroslava Hlaváčová, Václava Kettnerová, Zdeňka Urešová, Jenna Kanerva, Stina Ojala, Anna Missilä, Christopher D. Manning, Sebastian Schuster, Siva Reddy, Dima Taji, Nizar Habash, Herman Leung, Marie-Catherine de Marneffe, Manuela Sanguinetti, Maria Simi, Hiroshi Kanayama, Valeria de Paiva, Kira Droganova, Héctor Martínez Alonso, Çağrı Çöltekin, Umut Sulubacak, Hans Uszkoreit, Vivien Macketanz, Aljoscha Burchardt, Kim Harris, Katrin Marheinecke, Georg Rehm, Tolga Kayadelen, Mohammed Attia, Ali Elkahky, Zhuoran Yu, Emily Pitler, Saran Lertpradit, Michael Mandl, Jesse Kirchner, Hector Fernandez Alcalde, Jana Strnadová, Esha Banerjee, Ruli Manurung, Antonio Stella, Atsuko Shimada, Sookyoung Kwak, Gustavo Mendonça, Tatiana Lando, Rattima Nitisaroj, and Josie Li. 2017. Conll 2017 shared task: Multilingual parsing from raw text to universal dependencies. In *CoNLL 2017 Shared Task*, pages 1–19.

A Appendix: Discarded data

The algorithm for generating a dependency graph – obtained from Wu et al. (2020) – assumes that token IDs are unique and match positions in the sentence. However, in some coordinated sentences, the UD parse has the same ID appearing in two consecutive tokens. The BERT-parse, in turn, treats the repeated tokens as having separate IDs, which creates a disparity. Table 5 shows an example:

Token	Dep	ID (UD)	ID (BERT)
Durán	nsubj	1	1
acts	root	2	2
acts	conj	2	3
as	case	3	4
spokesman	obl	4	5
and	cc	5	6
Ángel	conj	6	7
Pintado	flat	7	8
as	case	8	9
treasurer	obl	9	10

Table 5: Mismatch between UD and BERT in token IDs.

Here, the verb (*acts*) is repeated since it serves a double role as the root and a conjunct. UD assigns the same ID (2) to both instances, but BERT uses an increasing counter of IDs. Hence, after the repetition, the respective token IDs between UD and BERT no longer match. Since dependent-head pairs are encoded in terms of IDs, this results in artificial disparities between the parses.

Because the number of such sentences in the PUD data was marginal (7), I discarded them in the experiments to avoid this problem. However, the original UAS, UUAS, and NED scores – obtained via replicating Wu et al. (2020) – are calculated from the full PUD data containing these sentences (see Footnote 6).

B Appendix: complete results

Table 6 displays each Dep that was subject to a dependent-head shift between BERT and UD. Tables 7–8 show the same per H_U and H_B, respectively. Table 9 lists all shifts that appeared at least 20 times in the format Dep–H_U–H_B. This comprises the data discussed in the main paper, from which Tables 1–4 are derived.

Dep	Ratio	Count
case	0.7251	1799
punct	0.5135	1252
det	0.5433	1105
nmod	0.8500	912
obl	0.7082	869
amod	0.5402	719
nsubj	0.4683	650
compound	0.6675	538
conj	0.8176	511
mark	0.7964	442
obj	0.5011	438
cc	0.7615	431
advmod	0.5035	426
nmod:poss	0.6703	244
advcl	0.7158	209
aux	0.4474	183
acl:relcl	0.8483	179
xcomp	0.5815	157
nummod	0.6071	153
nsubj:pass	0.5720	135
acl	0.6895	131
appos	0.8310	118
flat	0.4978	114
cop	0.3270	103
ccomp	0.7259	98
aux:pass	0.2915	79
parataxis	0.5979	58
fixed	0.5243	54
root	0.0363	36
compound:prt	0.4714	33
nmod:tmod	0.6667	26
csubj	0.5926	16
expl	0.2459	15
obl:npmod	0.7000	14
obl:tmod	0.6111	11
nmod:npmod	0.5263	10
det:predet	0.8889	8
cc:preconj	0.5455	6
csubj:pass	1.0000	3
dislocated	1.0000	2
reparandum	1.0000	1
discourse	1.0000	1
iobj	0.1000	1

Table 6: All dependency-head shifts ordered by Dep ("Ratio": ratio of shifts from all tokens with the Dep).

H_U	Ratio	Count
obl	0.6802	2048
root	0.2664	1694
nmod	0.6788	1655
conj	0.7654	1292
obj	0.7283	946
nsubj	0.6651	872
advcl	0.7791	663
acl:relcl	0.8109	579
xcomp	0.8168	495
ccomp	0.8327	458
acl	0.7762	281
appos	0.7301	238
parataxis	0.7409	223
nsubj:pass	0.6494	176
amod	0.7368	140
nmod:poss	0.7707	121
compound	0.6289	100
advmod	0.7810	82
csubj	0.7703	57
nummod	0.8036	45
flat	0.8276	24
cc	0.8750	14
obl:npmod	0.6667	14
obl:tmod	0.5833	14
csubj:pass	0.8667	13
mark	0.6000	9
nmod:tmod	0.2857	8
case	0.1591	7
dislocated	1.0000	6
nmod:npmod	0.8571	6
iobj	0.8333	5
dep	1.0000	2
det	0.6667	2
cc:preconj	1.0000	1

Table 7: All dependency-head shifts ordered by H_U ("Ratio": ratio of shifts from all tokens with the H_U).

H_D	Ratio	Count
root	0.4763	4244
case	0.9684	1135
amod	0.9386	764
compound	0.9107	602
nsubj	0.5525	542
obl	0.3431	503
nmod	0.3771	474
det	0.9978	453
punct	1.0000	404
obj	0.5306	399
advmod	0.9425	377
cc	0.9936	310
conj	0.4107	276
mark	0.9636	159
nummod	0.9341	156
advcl	0.4519	155
cop	1.0000	122
nsubj:pass	0.5622	122
nmod:poss	0.7707	121
aux	1.0000	119
xcomp	0.5174	119
acl	0.5622	104
flat	0.9533	102
aux:pass	1.0000	92
acl:relcl	0.3571	75
parataxis	0.4621	67
ccomp	0.3907	59
appos	0.3931	57
fixed	1.0000	55
compound:prt	1.0000	33
nmod:tmod	0.5455	24
expl	1.0000	14
obl:npmod	0.6316	12
det:predet	1.0000	9
nmod:npmod	0.9000	9
csubj	0.3462	9
cc:preconj	1.0000	4
obl:tmod	0.2308	3
reparandum	0.6667	2
dislocated	1.0000	1
discourse	1.0000	1
vocative	1.0000	1
csubj:pass	0.3333	1

Table 8: All dependency-head shifts ordered by H_B ("Ratio": ratio of shifts from all tokens with the H_B).

Dep–H_U–H_B shift (count)		
case-obl-root (521)	case-nmod-root (231)	cc-conj-root (191)
det-obj-root (141)	det-nsubj-root (134)	case-nmod-obl (122)
punct-root-obl (117)	nmod-obl-root (107)	det-obl-case (101)
det-nmod-case (100)	case-nmod-obj (99)	obl-root-case (97)
mark-xcomp-root (87)	nmod-nsubj-root (85)	mark-advcl-root (84)
nmod-obj-root (83)	punct-root-nsubj (79)	case-nmod-nsubj (79)
case-nmod-nmod (73)	det-obl-amod (66)	nsubj-ccomp-root (66)
amod-obj-root (64)	det-obl-root (62)	amod-obl-root (61)
case-nmod:poss-root (56)	nmod-nmod-root (54)	punct-root-advmod (53)
case-obl-acl (52)	nsubj-acl:relcl-root (52)	amod-nsubj-root (49)
punct-root-punct (45)	compound-nsubj-root (45)	mark-ccomp-root (44)
compound-obl-root (44)	compound-nmod-root (43)	obl-xcomp-root (43)
obl-acl-root (43)	obl-acl:relcl-root (43)	punct-conj-cc (41)
obl-conj-root (41)	amod-obj-det (40)	obl-root-amod (40)
punct-root-nmod (38)	amod-nmod-root (38)	obl-advcl-root (38)
obl-root-compound (38)	nsubj-advcl-root (37)	obj-advcl-root (36)
nummod-obl-root (36)	punct-root-parataxis (35)	nsubj-root-amod (35)
obj-xcomp-root (35)	punct-conj-conj (35)	nmod-obl-case (34)
case-obl-advcl (33)	case-obl-conj (33)	punct-conj-root (32)
nmod-obj-case (32)	det-nmod-amod (31)	amod-nmod-case (31)
nmod-nmod-case (31)	nsubj-root-compound (31)	nmod:poss-obl-case (31)
punct-appos-root (30)	case-obl-acl:relcl (30)	conj-nmod-root (30)
case-nmod-det (29)	det-nsubj-amod (28)	nmod-obj-amod (28)
cc-conj-obl (27)	punct-conj-nmod (26)	case-nmod-conj (26)
det-nmod-root (26)	det-obj-advcl (26)	nmod-obl-compound (26)
det-nmod-compound (25)	nmod-conj-root (25)	compound-obj-root (25)
nsubj-conj-root (25)	obj-acl-root (25)	det-nsubj:pass-root (24)
obl-root-nmod (24)	conj-nsubj-root (24)	amod-obl-det (23)
nmod:poss-nmod-case (23)	nmod:poss-nsubj-root (23)	punct-conj-obl (22)
det-obj-amod (22)	obl-acl:relcl-case (22)	nsubj-root-case (22)
cc-conj-nmod (22)	advmod-advcl-root (22)	conj-nmod-cc (22)
nmod-nsubj-case (21)	obl-root-nummod (21)	flat-nsubj-root (21)
obj-acl:relcl-root (21)	acl-obj-root (21)	punct-root-det (20)
case-obl-xcomp (20)	nmod-obl-amod (20)	compound-obl-det (20)
compound-nmod-case (20)	obl-ccomp-root (20)	

Table 9: Dep–H_U–H_B shifts and their counts (minimum count: 20).