Universal Dependencies According to BERT: Both More Specific and More General

Tomasz Limisiewicz and **David Mareček** and **Rudolf Rosa**
Institute of Formal and Applied Linguistics, Faculty of Mathematics and Physics
Charles University, Prague, Czech Republic
{limisiewicz,rosa,marecek}@ufal.mff.cuni.cz

Abstract

This work focuses on analyzing the form and extent of syntactic abstraction captured by BERT by extracting labeled dependency trees from self-attentions.

Previous work showed that individual BERT heads tend to encode particular dependency relation types. We extend these findings by explicitly comparing BERT relations to Universal Dependencies (UD) annotations, showing that they often do not match one-to-one. We suggest a method for relation identification and syntactic tree construction. Our approach produces significantly more consistent dependency trees than previous work, showing that it better explains the syntactic abstractions in BERT.

At the same time, it can be successfully applied with only a minimal amount of supervision and generalizes well across languages.

1 Introduction and Related Work

In recent years, systems based on Transformer architecture achieved state-of-the-art results in language modeling (Devlin et al., 2019) and machine translation (Vaswani et al., 2017). Additionally, the contextual embeddings obtained from the intermediate representation of the model brought improvements in various NLP tasks. Multiple recent works try to analyze such latent representations (Linzen et al., 2019), observe syntactic properties in some Transformer self-attention heads, and extract syntactic trees from the attentions matrices (Raganato and Tiedemann, 2018; Mareček and Rosa, 2019; Clark et al., 2019; Jawahar et al., 2019).

In our work, we focus on the comparative analysis of the syntactic structure, examining how the BERT self-attention weights correspond to Universal Dependencies (UD) syntax (Nivre et al., 2016). We confirm the findings of Vig and Belinkov (2019) and Voita et al. (2019) that in Transformer based systems particular heads tend to capture specific dependency relation types (e.g. in one head the attention at the predicate is usually focused on the nominal subject).

We extend understanding of syntax in BERT by examining the ways in which it systematically diverges from standard annotation (UD). We attempt to bridge the gap between them in three ways:

- We modify the UD annotation of three linguistic phenomena to better match the BERT syntax (§3)
- We introduce a head ensemble method, combining multiple heads which capture the same dependency relation label (§4)
- We observe and analyze multipurpose heads, containing multiple syntactic functions (§7)

Finally, we apply our observations to improve the method of extracting dependency trees from attention (§5), and analyze the results both in a monolingual and a multilingual setting (§6).

Our method crucially differs from probing (Belinkov et al., 2017; Hewitt and Manning, 2019; Chi et al., 2020; Kulmizev et al., 2020). We do not use treebank data to train a parser; rather, we extract dependency relations directly from selected attention heads. We only employ syntactically annotated data to select the heads; however, this means estimating relatively few parameters, and only a small amount of data is sufficient for that purpose (§6.1).

2 Models and Data

We analyze the uncased base BERT model for English, which we will refer to as **enBERT**, and the uncased multilingual BERT model, **mBERT**, for English, German, French, Czech, Finnish, Indonesian, Turkish, Korean, and Japanese [1]. The code

[1] Pretrained models are available at https://github.com/google-research/bert

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 2710–2722
November 16 - 20, 2020. ©2020 Association for Computational Linguistics

shared by Clark et al. (2019) [2] substantially helped us in extracting attention weights from BERT.

To find syntactic heads, we use: 1000 EuroParl multi parallel sentences (Koehn, 2004) for five European languages, automatically annotated with UDPipe UD 2.0 models (Straka and Straková, 2017); Google Universal Dependency Treebanks (GSD) for Indonesian, Korean, and Japanese (McDonald et al., 2013); the UD Turkish Treebank (IMST-UD) (Sulubacak et al., 2016).

We use another PUD treebanks from the CoNLL 2017 Shared Task for evaluation of mBERT in all languages (Nivre et al., 2017)[3].

3 Adapting UD to BERT

Since the explicit dependency structure is not used in BERT training, syntactic dependencies captured in latent layers are expected to diverge from annotation guidelines. After initial experiments, we have observed that some of the differences are systematic (see Table 1).

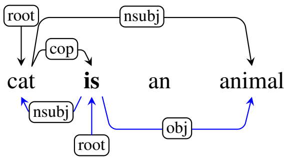
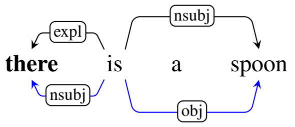
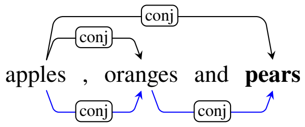
UD	Modified	Example
Copula attaches to a noun	Copula is a root. [4]	 root nsubj cop cat is an animal nsubj obj root
Expletive is not a subject	Expletive is treated as a subject	
In multiple coordination, all conjuncts attach to the first conjunct	Conjunct attaches to a previous one	 conj conj apples , oranges and pears conj conj

Table 1: Comparison of original Universal Dependencies annotations (**edges above**) and our modification (edges below).

Based on these observations, we modify the UD annotations in our experiments to better fit the BERT syntax, using UDApi[5] (Popel et al., 2017).

The main motivation of our approach is to get trees similar to structures emerging from BERT, which we have observed in qualitative analysis of attention weights. We note that for copulas and coordinations, BERT syntax resembles Surface-syntactic UD (SUD) (Gerdes et al., 2018). Nevertheless, we decided to use our custom modification, since some systematic divergences between SUD and the latent representation occur as well. It is not our intention to compare two annotation guidelines. A comprehensive comparison between extracting UD and extracting SUD trees from BERT was performed by (Kulmizev et al., 2020). However, they used a probing approach, which is noticeably different from our setting.

4 Head Ensemble

In line with Clark et al. (2019) and other studies Voita et al. (2019); Vig and Belinkov (2019), we have noticed that a specific syntactic relation type can often be found in a specific head. Additionally, we observe that a single head often captures only a specific aspect or subtype of one UD relation type, motivating us to combine multiple heads to cover the full relation.

Figure 1 shows attention weights of two syntactic heads (right columns) and their average (left column). In the top row (purple), both heads identify the parent noun for an adjectival modifier: Head 9 in Layer 3 if their distance is two positions or less, Head 10 in Layer 7 if they are further away (as in "a stable , green economy").

Similarly, for an object to predicate relation (blue bottom row), Head 9 in Layer 7 and Head 8 in Layer 3 capture pairs with shorter and longer positional distances, respectively.

4.1 Dependency Accuracy of Heads

To quantify the amount of syntactic information conveyed by a self-attention head A for a dependency relation label l in a specific direction d (for instance predicate → subject), we compute:

$$DepAcc_{l,d,A} = \frac{|\{(i,j) \in E_{l,d} : j = \arg\max A[i]\}|}{|E_{l,d}|}$$

where $E_{l,d}$ is a set of all dependency tree edges with the label l and with direction d, i.e., in dependent to parent direction (abbreviated to **p2d**)

[2] https://github.com/clarkkev/attention-analysis

[3] Mentioned treebanks are available at the Universal Dependencies web page https://universaldependencies.org

[4] Certain dependents of the original root (e.g., subject, auxiliaries) are rehanged and attached to the new root – copula verb.

[5] https://udapi.github.io

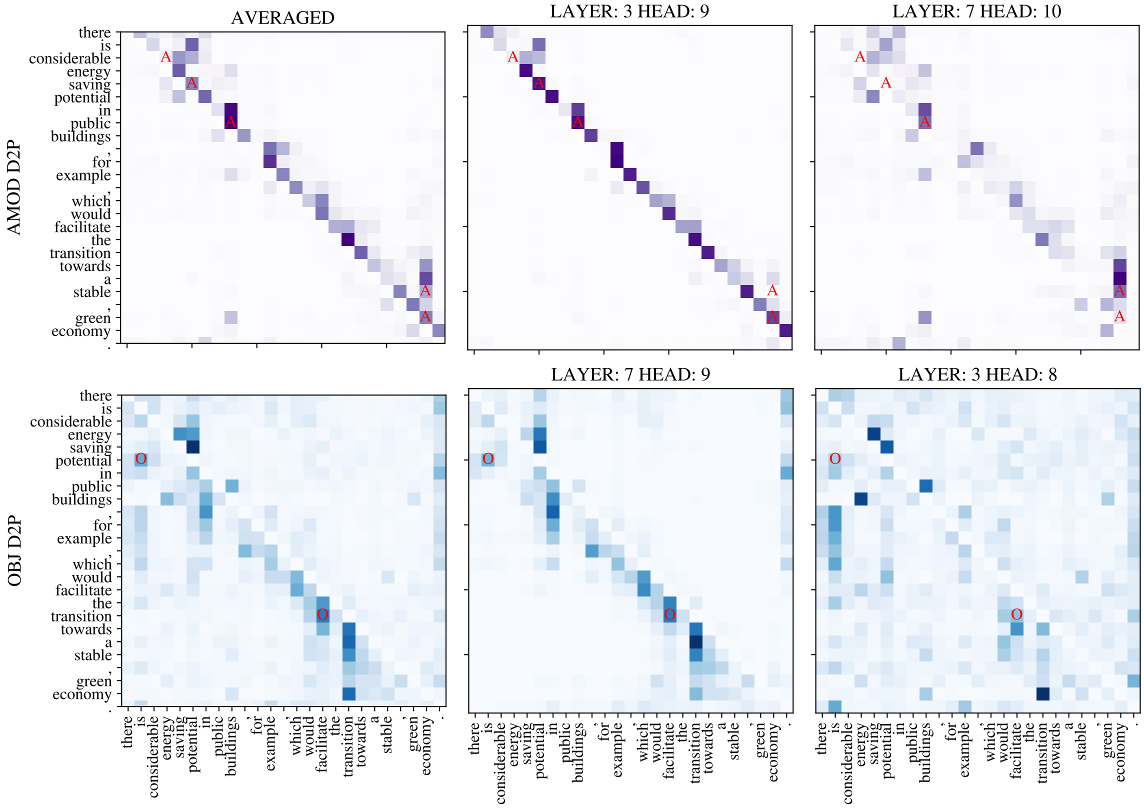


Figure 1: Examples of two enBERT's attention heads covering the same relation label and their average. Gold relations are marked by red letters.

the first element of the tuple i is dependent of the relation and the second element j is the governor; $A[i]$ is the i^{th} row of the attention matrix A.

In this article, when we say that head with attention matrix A is syntactic for a relation type l, we mean that its $DepAcc_{l,d,A}$ is high in one of the directions (parent to dependent **p2d** or dependent to parent **d2p**).

4.2 Method

Having observed that some heads convey only partial information about a UD relation, we propose a method to connect knowledge of multiple heads.

Our objective is to find a set of heads for each directed relation so that their attention weights after averaging have a high dependency accuracy. The algorithm is straightforward: we define the maximum number N of heads in the subset; sort the heads based on their $DepAcc$ on development set; starting from the most syntactic one we check whether including head's attention matrix in the average would increase $DepAcc$; if it does the head is added to the ensemble. When there are already N heads in the ensemble, the newly added head may substitute another added before, so to maximize $DepAcc$ of the averaged attention matrices.[6]

We set N to be 4, as allowing larger ensembles does not improve the results significantly.

5 Dependency Tree Construction

To extract dependency trees from self-attention weights, we use a method similar to Raganato and Tiedemann (2018), which employs a maximum spanning tree algorithm (Edmonds, 1966) and uses gold information about the root of the syntax tree.

We use the following steps to construct a labeled dependency tree:

1. For each non-clausal UD relation label, syntactic heads ensembles are selected as described in Section 4. Attention matrices in the ensembles are averaged. Hence, we obtain two matrices for each label (one for each direction: "dependent to parent" and "parent to dependent")

2. The "dependent to parent" matrix is transposed and averaged with "parent to dependent" matrix. We use a weighted geometric

[6] The code is available at GitHub: https://github.com/Tom556/BERTHeadEnsembles

average with weights corresponding to dependency accuracy values for each direction.

3. We compute the final dependency matrix by max-pooling over all individual relation-label matrices from step 2. At the same time, we save the syntactic-relation label that was used for each position in the final matrix.

4. In the final matrix, we set the row corresponding to the gold root to zero, to assure it will be the root in the final tree as well.

5. We use the Chu-Liu-Edmond's algorithm (Edmonds, 1966) to find the maximum spanning tree. For each edge, we assign the label saved in step 3.

It is important to note that the total number of heads used for tree construction can be at most $4 * 12 * 2 = 96$, (number of heads per ensemble $*$ number of considered labels $*$ two directions). However, the number of used heads is typically much lower (see Table 3). That means our method uses at most 96 integer parameters (indices of the selected heads), considerably less than projection layers in fine-tuning or structural probing, consisting of thousands of real parameters.

As far as we know, we are first to construct labeled dependency trees from attention matrices in Transformer. Moreover, we have extended the previous approach by using an ensemble of heads instead of a single head.

6 Results

6.1 Dependency Accuracy

In Table 2, we present results for the dependency accuracy (Section 4.1) of a single head, four heads ensemble, and the positional baseline.[10]

Noticeably, a single attention head surpasses the baseline for every relation label in at least one direction. The average of 4 heads surpasses the baseline by more than 10% for every relation.

Ensembling brings the most considerable improvement for nominal subjects (p2d: +13.3 pp) and noun modifiers (p2d: +13.2 pp). The relative change of accuracy is more evident for clausal relations than non-clausal. Dependent to parent direction has higher accuracy for modifiers (except adverbial modifiers), functional relations, and objects, whereas parent to dependent favors other nominal relations (nominal subject and nominal modifiers).

Relation label	Baseline	1 Head		4 Heads	
		d2p	p2d	d2p	p2d
amod	78.3	90.6	77.5	**93.8**	79.5
advmod	48.7	53.3	62.0	62.1	**63.6**
aux	69.2	90.9	86.9	**94.5**	88.0
case	36.4	83.0	67.1	**88.4**	68.9
compound	75.8	83.2	75.8	**87.0**	79.1
conjunct	31.7	47.4	41.6	**58.8**	51.3
det	56.5	95.2	62.3	**97.2**	69.4
nmod	25.4	34.3	41.5	49.1	**54.7**
nummod	57.9	75.9	64.6	**79.3**	72.6
mark	53.7	66.2	54.7	**73.5**	65.9
obj[7]	39.2	84.9	68.6	**89.3**	78.5
nsubj	45.8	56.2	62.7	57.8	**76.0**
⇑ AVG. NON-CLAUSAL	52.8	67.8		**74.1**	
acl	27.9	41.5	36.5	**50.5**	43.8
advcl	9.3	26.3	26.7	**40.7**	26.3
csubj	20.0	20.7	**31.0**	24.1	**31.0**
x/ccomp[8]	34.8	60.4	47.9	**66.9**	52.1
parataxis	10.4	17.6	12.1	23.1	**24.2**
⇑ AVG. CLAUSAL	20.5	32.1		**38.3**	
punct	9.4	21.1	40.3	28.4	**44.0**
dep[9]	18.8	21.6	33.1	25.1	**37.0**

Table 2: Dependency accuracy for single heads, 4 heads ensembles, and positional baselines. The evaluation was done using the pretrained model enBERT and modified UD as described in Section 3.

Introducing the UD modifications (Section 3) had a significant effect for nominal subject. Without such modifications, the accuracy for parent to dependent direction would drop from 76.0% to 70.1%

Selection Supervision The selection of syntactic heads requires annotated data for accuracy evaluation. In Figure 2, we examine what number of annotated sentences is sufficient, using 1, 10, 20, 50, 100 or 1000 sentences.

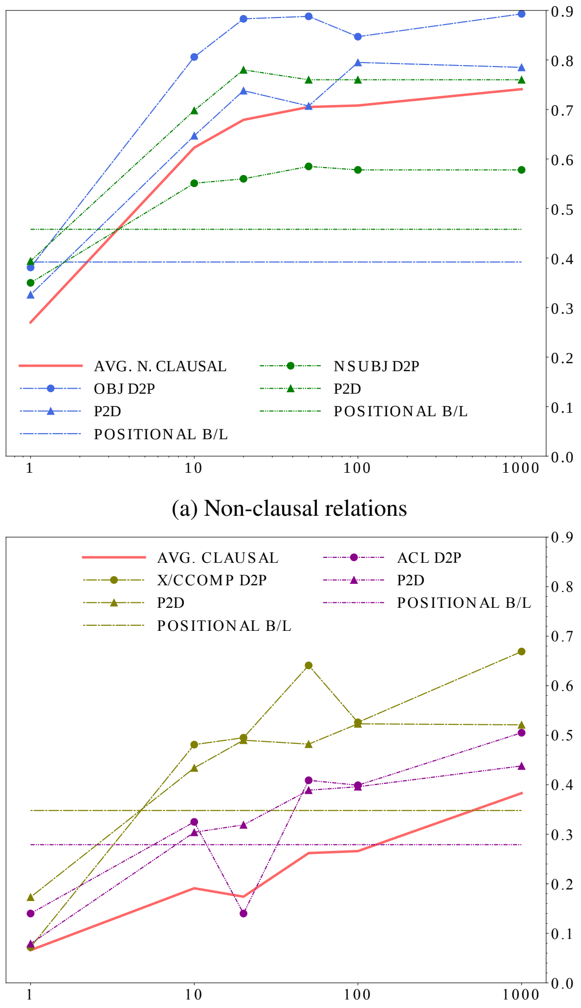
For non-clausal relations (Figure 2a), head selection on just 10 annotated sentences allows us to surpass the positional baseline. Using over 20 examples brings only a minor improvement. For clausal relations (Figure 2b), the score improves steadily with more data. However, even for the full corpus, it is relatively low, since the clausal relations are less frequent in the corpus and harder to

[7] Objects also include indirect objects (*iobj*).

[8] Open clausal complements and clausal complements.

[9] *Dep* relations and all relations not included in this table.

[10] The positional baseline looks at the most frequent relative position for each dependency label (Voita et al., 2019).



(a) Non-clausal relations

(b) Clausal relations

Figure 2: Dependency accuracy against the number of sentences used for selection.

identify due to longer distances between dependent and parent.

6.2 Dependency Tree Construction

In Table 3, we report the evaluation results on the English PUD treebank (Nivre et al., 2017) using unlabeled and labeled attachment scores (UAS and LAS). For comparison, we also include the left- and right-branching baseline with gold root information, and the highest score obtained by Raganato and Tiedemann (2018) who used the neural machine translation Transformer model and extracted whole trees from a single attention head. Also, they did not perform direction averaging. The results show that ensembling multiple attention heads for each relation label allows us to construct much better trees than the single-head approach.[11]

The number of unique heads used in the process turned out to be two times lower than the maximal possible number (96). This is because many heads appear in multiple ensembles. We examine it further in Section 7.

Furthermore, to the best of our knowledge, we are the first to produce labeled trees and report both UAS and LAS.

Just for reference, the recent unsupervised parser (Han et al., 2019) obtains 61.4% UAS. However, the results are not comparable since the parser uses information about gold POS tags, and the results were measured on different evaluation data (WSJ Treebank).

Ablation We analyze how much the particular steps described in Section 5 influenced the quality of constructed trees. We also repeat the experimental setting proposed by Raganato and Tiedemann (2018) on enBERT model to see whether a language model is better suited to capture syntax than a translation system. Additionally, we alter the procedure described in Section 5 to analyze which decision influenced our results the most, i.e., we change:

- Size of head ensembles
- Number of sentences used for head selection
- Use the same head ensemble for all relation labels in each direction. Hence we do not conduct max-pooling described in section 5, point 3.

In Table 3, we see that the method by Raganato and Tiedemann (2018) applied to enBERT produces slightly worse trees than the same method applied to neural machine translation. If we do not use ensembles and only one head per each relation label and direction is used, our pipeline from Section 5 offers only 0.2 pp rise in UAS and poor LAS. The analysis shows that the introduction of head ensembles of size four has brought the most significant improvement in our method of tree construction, which is roughly +15 pp for both the variants (with and without labels).

Together with the findings in Section 6.1 this supports our claim that syntactic information is spread across many Transformer's heads. Interestingly, max-pooling over labeled matrices improve

[11] To assure comparability we do not modify the UD annotation for the results in this table.

Setting	Use labels	Model	Selection sentences	Heads per ensemble	Heads used	UAS	LAS
Left branching baseline	—	—	—	—	—	11.0	—
Right branching baseline	—	—	—	—	—	35.5	—
Raganato+ (paper)	no	NMT	1000*	—	1	38.9	—
Raganato+	no	enBERT	1000*	—	1	37.2	—
Our method	no	enBERT	1000	1	2	36.0	—
	yes	enBERT	1000	1	15	37.4	9.5
	yes	enBERT	20	4	36	43.6	14.5
	no	enBERT	1000	4	8	51.2	—
Our method	yes	enBERT	1000	4	48	**52.0**	**21.7**

Table 3: Evaluation results for different settings of dependency trees extraction. UD modifications were not applied here. (*In Raganato+ experimens, the trees were induced from each encoder head, but we report only the results for the head with the highest UAS on 1000 test sentences.)

Language	Features	DepAcc b-line	DepAcc Our	UAS b-line	UAS Our	LAS Our
EN	SVO, AN	52.8	**73.2**	35.5	**51.0**	21.8
DE	—[12], AN	42.3	**72.9**	32.9	**45.5**	19.5
FR	SVO, NA	50.6	**72.8**	34.7	**48.3**	18.0
CS	SVO, AN	44.3	**69.7**	34.0	**40.1**	17.1
FI	SVO, AN	55.6	**77.0**	35.5	**45.8**	15.9
ID	SVO, NA	47.0	**64.2**	29.7	**36.9**	14.6
TR	SOV, AN	60.0	**68.0**	**38.8**	29.3	7.9
KO	SOV, AN	**41.8**	32.4	**49.3**	28.8	8.0
JA	SOV, AN	56.9	**69.5**	35.9	**39.0**	14.3
Mean SVO		50.1	**71.4**	33.9	**44.4**	17.5
Mean SOV		52.8	**56.7**	**34.1**	32.4	13.9
Mean AN		50.6	**66.1**	34.3	**39.9**	16.6
Mean NA		48.8	**68.5**	32.2	**42.6**	16.3

Table 4: Average dependency accuracy for non-clausal relations (with UD modification) compared with positional baseline. UAS, LAS of constructed trees (w/o UD modification) compared with UAS of left or right branching tree with gold root, whichever is higher. mBERT was used for all languages.

UAS only by 0.8 pp. Nevertheless, this step is necessary to construct labeled trees. The performance is competitive, even with as little as 20 sentences used for head selection, which is in line with our findings from Section 6.1.

Multilingual Setting In table 4 we present the results of our methods applied to mBERT and evaluated on Parallel Universal Dependencies in nine languages. Comparison of the results for English with table 3 shows that the dependency accuracy and UAS decreased only slightly by changing the model from enBERT to mBERT, while LAS saw 0.1 pp increase. The model captures syntax comparably well in German, French, and Finnish.

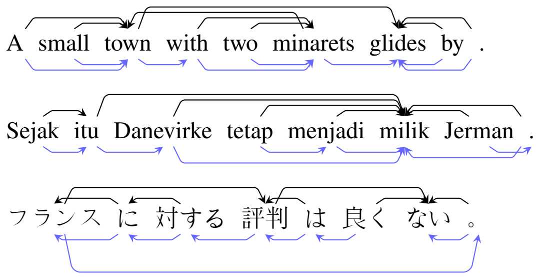

A small town with two minarets glides by .
Sejak itu Danevirke tetap menjadi milik Jerman .
フランス に 対する 評判 は 良く ない 。

Figure 3: English, Indonesian, and Japanese examples of mBERT extracted trees edges below compared with the correct trees **edges above**. For Japanese sentence predicted structure is a left branching chain, which is a strong baseline for this language. English translation of the sentences: from Indonesian: *"The Danevirke has remained in German possession ever since."*; from Japanese: *"France doesn't have a good reputation."*

We observe that results for languages following Subject-Object-Verb (SOV) order (Turkish, Korean, Japanese) are significantly lower than for SVO languages (English, French, Czech, Finnish, Indonesian) in both Dependency Accuracy (14.7 pp) and the UAS (10.5 pp). Our methods outperform the baselines in the latter group by 17.2 pp to 25.4 pp for Dependency Accuracy and from 6.1 pp to 15.5 pp for UAS. The influence of Adjective and Noun order is less apparent. On average, the NA languages results are higher than for the AN languages by 2.4 pp in Dependency Accuracy and 2.7 pp in UAS.

[12] No dominant order

The disparity in the results for SVO and SOV languages was previously observed by (Pires et al., 2019), who fine-tuned mBERT for part of speech tagging and evaluated zero-shot accuracy across typologically diverse languages. We hypothesize that worse performance for SOV languages may be due to their lower presence in mBERT's pre-training corpus.

7 Multipurpose Heads

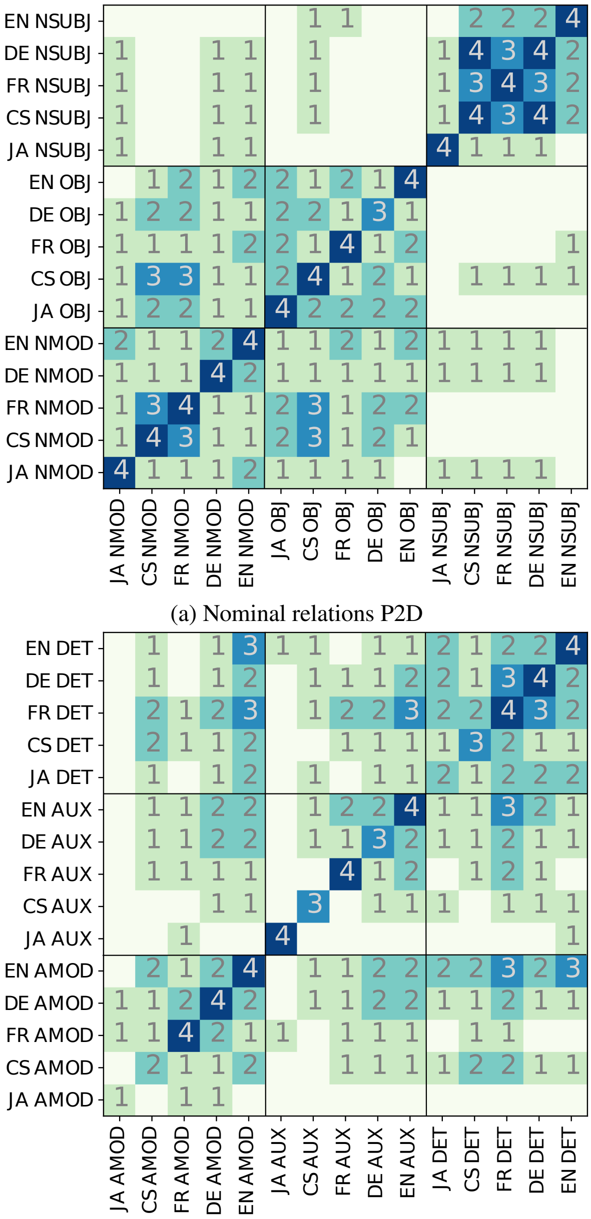
In this experiment, we examine whether a single mBERT's head can perform multiple syntactic functions in a multilingual setting. We choose an ensemble for each syntactic relation for each language. Figure 4 presents the sizes of intersections between head sets for different languages and dependency labels.

Except from Japanese, we observe an overlap of the heads pointing to the governor of adjective modifiers, auxiliaries, and determiners. Shared heads tend to find the root of the syntactic phrase. Interestingly, common heads occur even for relations typically belonging to a verb and noun phrases, such as auxiliaries and adjective modifiers. In our other experiments, we have noticed that these heads do not focus their attention on any particular part of speech. Similarly, objects and noun modifiers share at least one head for all languages. They have a similar function in a sentence; however, they connect with the verb and noun, respectively. Such behavior was also observed in a monolingual model. Figure 5 presents attention weights of two heads that belong to the intersection of the adjective modifier, auxiliary, and determiner dependent to parent ensembles.

7.1 Cross-lingual intersections

Representation of mBERT is language independent to some extent (Pires et al., 2019; Libovickỳ et al., 2019). Thus, a natural question is whether the same mBERT heads encode the same syntactic relations for different languages. In particular, subject relations tend to be encoded by similar heads in different languages, which rarely belong to an ensemble for other dependency labels. Again Japanese is an exception here, possibly due to different Object-Verb order.

For adjective modifiers, the French ensemble has two heads in common with the German and one with other considered languages, although the preferred order of adjective and noun is different.



	JA NMOD	CS NMOD	FR NMOD	DE NMOD	EN NMOD	JA OBJ	CS OBJ	FR OBJ	DE OBJ	EN OBJ	JA NSUBJ	CS NSUBJ	FR NSUBJ	DE NSUBJ	EN NSUBJ
EN NSUBJ							1	1				2	2	2	4
DE NSUBJ	1			1	1		1				1	4	3	4	2
FR NSUBJ	1			1	1		1				1	3	4	3	2
CS NSUBJ	1			1	1		1				1	4	3	4	2
JA NSUBJ	1			1	1						4	1	1	1	
EN OBJ		1	2	1	2	2	1	2	1	4					
DE OBJ	1	2	2	1	1	2	2	1	3	1					
FR OBJ	1	1	1	1	2	2	1	4	1	2					1
CS OBJ	1	3	3	1	1	2	4	1	2	1		1	1	1	1
JA OBJ	1	2	2	1	1	4	2	2	2	2					
EN NMOD	2	1	1	2	4	1	1	2	1	2	1	1	1	1	
DE NMOD	1	1	1	4	2	1	1	1	1	1	1	1	1	1	
FR NMOD	1	3	4	1	1	2	3	1	2	2					
CS NMOD	1	4	3	1	1	2	3	1	2	1					
JA NMOD	4	1	1	1	2	1	1	1	1		1	1	1	1	

(a) Nominal relations P2D

	JA AMOD	CS AMOD	FR AMOD	DE AMOD	EN AMOD	JA AUX	CS AUX	FR AUX	DE AUX	EN AUX	JA DET	CS DET	FR DET	DE DET	EN DET
EN DET		1		1	3	1	1		1	1	2	1	2	2	4
DE DET		1		1	2		1	1	1	2	2	1	3	4	2
FR DET		2	1	2	3		1	2	2	3	2	2	4	3	2
CS DET		2	1	1	2			1	1	1	1	3	2	1	1
JA DET		1		1	2		1		1	1	2	1	2	2	2
EN AUX		1	1	2	2		1	2	2	4	1	1	3	2	1
DE AUX		1	1	2	2		1	1	3	2	1	1	2	1	1
FR AUX		1	1	1	1			4	1	2		1	2	1	
CS AUX				1	1		3		1	1	1		1	1	1
JA AUX			1			4									1
EN AMOD		2	1	2	4		1	1	2	2	2	2	3	2	3
DE AMOD	1	1	2	4	2		1	1	2	2	1	1	2	1	1
FR AMOD	1	1	4	2	1	1		1	1	1		1	1		
CS AMOD		2	1	1	2			1	1	1	1	2	2	1	1
JA AMOD	1		1	1											

(b) Adjective modifiers, auxiliaries, determiners D2P

Figure 4: Number of mBERT's heads shared between relations, both within and across languages.

This phenomenon could be explained by the fact that only a few frequent French adjectives precede modified nouns (e.g. "bon", "petit", "grand"). Attention weights of a head capturing adjective modifiers in French, German, English, and Czech are presented in Figure 6.

8 Conclusion

We have expanded the knowledge about the representation of syntax in self-attention heads of the Transformer architecture. We modified the UD annotation to fit the BERT syntax better. We analyzed the phenomenon of information about one depen-

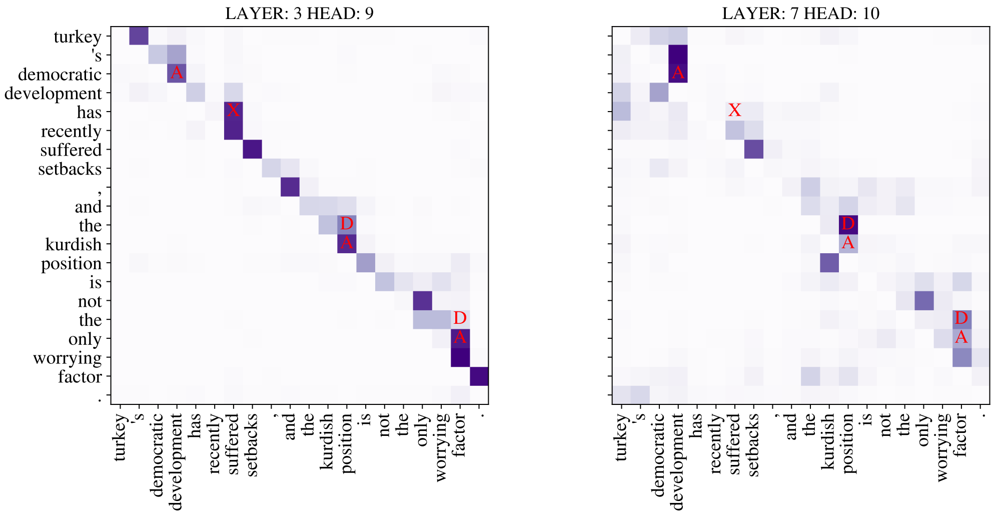
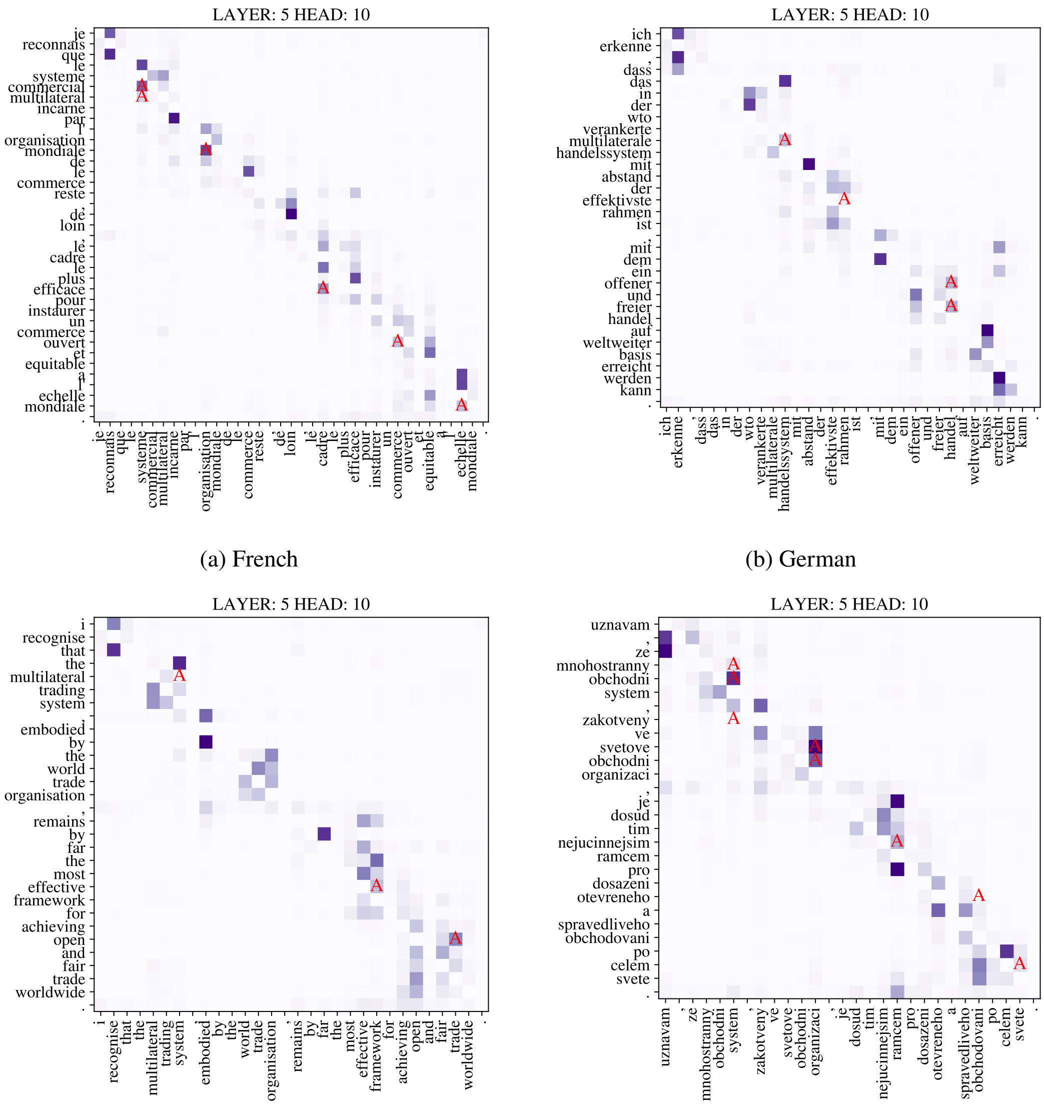


Figure 5: Syntactic enBERT heads retrieving the parent for three relation labels: **A**djective modifiers, Au**X**iliaries, **D**eterminers. UD relations are marked by A, X, and D respectively.



(a) French

(b) German

(c) English

(d) Czech

Figure 6: A single mBERT head which identifies noun heads of French adjective modifiers. It also partially captures the relation in German, English, and Czech, although these languages, unlike French, follow "Adjective Noun" order.

dency relation type being split among many heads and the opposite situation where one head has multiple syntactic functions.

Our method of head ensembling improved the previous results for dependency relation retrieval and extraction of syntactic trees from self-attention matrices. As far as we know, this is the first work that conducted a similar analysis for languages other than English. We have shown that the method generalizes well across languages, especially those following Subject Verb Object order.

We also hypothesize that the proposed method could improve dependency parsing in a low supervision setting.

Acknowledgments

This work has been supported by grant 18-02196S of the Czech Science Foundation. It has been using language resources and tools developed, stored and distributed by the LINDAT/CLARIAH-CZ project of the Ministry of Education, Youth and Sports of the Czech Republic (project LM2018101)

References

Yonatan Belinkov, Lluís Màrquez, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2017. Evaluating layers of representation in neural machine translation on part-of-speech and semantic tagging tasks. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1–10, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Ethan A. Chi, John Hewitt, and Christopher D. Manning. 2020. Finding universal grammatical relations in multilingual BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5564–5577, Online. Association for Computational Linguistics.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *BlackBoxNLP@ACL*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Jack Edmonds. 1966. Optimums branchings. 71B.

Kim Gerdes, Bruno Guillaume, Sylvain Kahane, and Guy Perrier. 2018. SUD or Surface-Syntactic Universal Dependencies: An annotation scheme near-isomorphic to UD. In *Universal Dependencies Workshop 2018*, Brussels, Belgium.

Wenjuan Han, Yong Jiang, and Kewei Tu. 2019. Enhancing unsupervised generative dependency parser with contextual information. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5315–5325, Florence, Italy. Association for Computational Linguistics.

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.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Philipp Koehn. 2004. Europarl: A parallel corpus for statistical machine translation. 5.

Artur Kulmizev, Vinit Ravishankar, Mostafa Abdou, and Joakim Nivre. 2020. Do neural language models show preferences for syntactic formalisms? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4077–4091, Online. Association for Computational Linguistics.

Jindřich Libovickỳ, Rudolf Rosa, and Alexander Fraser. 2019. How language-neutral is multilingual bert? *arXiv preprint arXiv:1911.03310*.

Tal Linzen, Grzegorz Chrupała, Yonatan Belinkov, and Dieuwke Hupkes, editors. 2019. *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Association for Computational Linguistics, Florence, Italy.

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, Florence, Italy. Association for Computational Linguistics.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97, Sofia, Bulgaria. Association for Computational Linguistics.

Joakim Nivre, Željko Agić, Lars Ahrenberg, Lene Antonsen, Maria Jesus Aranzabe, Masayuki Asahara, Luma Ateyah, Mohammed Attia, Aitziber

Atutxa, Elena Badmaeva, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, John Bauer, Kepa Bengoetxea, Riyaz Ahmad Bhat, Eckhard Bick, Cristina Bosco, Gosse Bouma, Sam Bowman, Aljoscha Burchardt, Marie Candito, Gauthier Caron, Gülşen Cebiroğlu Eryiğit, Giuseppe G. A. Celano, Savas Cetin, Fabricio Chalub, Jinho Choi, Yongseok Cho, Silvie Cinková, Çağrı Çöltekin, Miriam Connor, Marie-Catherine de Marneffe, Valeria de Paiva, Arantza Diaz de Ilarraza, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Marhaba Eli, Ali Elkahky, Tomaž Erjavec, Richárd Farkas, Hector Fernandez Alcalde, Jennifer Foster, Cláudia Freitas, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta Gonzáles Saavedra, Matias Grioni, Normunds Grūzītis, Bruno Guillaume, Nizar Habash, Jan Hajič, Jan Hajič jr., Linh Hà Mỹ, Kim Harris, Dag Haug, Barbora Hladká, Jaroslava Hlaváčová, Petter Hohle, Radu Ion, Elena Irimia, Anders Johannsen, Fredrik Jørgensen, Hüner Kaşıkara, Hiroshi Kanayama, Jenna Kanerva, Tolga Kayadelen, Václava Kettnerová, Jesse Kirchner, Natalia Kotsyba, Simon Krek, Sookyoung Kwak, Veronika Laippala, Lorenzo Lambertino, Tatiana Lando, Phương Lê Hồng, Alessandro Lenci, Saran Lertpradit, Herman Leung, Cheuk Ying Li, Josie Li, Nikola Ljubešić, Olga Loginova, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Cătălina Mărănduc, David Mareček, Katrin Marheinecke, Héctor Martínez Alonso, André Martins, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Gustavo Mendonça, Anna Missilä, Verginica Mititelu, Yusuke Miyao, Simonetta Montemagni, Amir More, Laura Moreno Romero, Shunsuke Mori, Bohdan Moskalevskyi, Kadri Muischnek, Nina Mustafina, Kaili Müürisep, Pinkey Nainwani, Anna Nedoluzhko, Lương Nguyễn Thị, Huyền Nguyễn Thị Minh, Vitaly Nikolaev, Rattima Nitisaroj, Hanna Nurmi, Stina Ojala, Petya Osenova, Lilja Øvrelid, Elena Pascual, Marco Passarotti, Cenel-Augusto Perez, Guy Perrier, Slav Petrov, Jussi Piitulainen, Emily Pitler, Barbara Plank, Martin Popel, Lauma Pretkalniņa, Prokopis Prokopidis, Tiina Puolakainen, Sampo Pyysalo, Alexandre Rademaker, Livy Real, Siva Reddy, Georg Rehm, Larissa Rinaldi, Laura Rituma, Rudolf Rosa, Davide Rovati, Shadi Saleh, Manuela Sanguinetti, Baiba Saulīte, Yanin Sawanakunanon, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Lena Shakurova, Mo Shen, Atsuko Shimada, Muh Shohibussirri, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Aaron Smith, Antonio Stella, Jana Strnadová, Alane Suhr, Umut Sulubacak, Zsolt Szántó, Dima Taji, Takaaki Tanaka, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Francis Tyers, Sumire Uematsu, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Gertjan van Noord, Viktor Varga, Veronika Vincze, Jonathan North Washington, Zhuoran Yu, Zdeněk Žabokrtský, Daniel Zeman, and Hanzhi Zhu. 2017. Universal dependencies 2.0 – CoNLL 2017 shared task development and test data. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Joakim Nivre, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Martin Popel, Zdeněk Žabokrtský, and Martin Vojtek. 2017. Udapi: Universal API for universal dependencies. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 96–101, Gothenburg, Sweden. Association for Computational Linguistics.

Alessandro Raganato and Jörg Tiedemann. 2018. An analysis of encoder representations in transformer-based machine translation. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 287–297, Brussels, Belgium. Association for Computational Linguistics.

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

Umut Sulubacak, Gülşen Eryiğit, and Tuğba Pamay. 2016. Imst: A revisited turkish dependency treebank. In *Proceedings of TurCLing 2016, the 1st International Conference on Turkic Computational Linguistics*, pages 1–6, Turkey. EGE UNIVERSITY PRESS.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.

Jesse Vig and Yonatan Belinkov. 2019. Analyzing the structure of attention in a transformer language model. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 63–76, Florence, Italy. Association for Computational Linguistics.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.

A Technical Details

A.1 Computing Infrastructure

We have used one CPU core *Intel(R) Xeon(R) CPU E5-2630 v3* for both head ensemble selection and dependency tree construction. The attention matrices were computed on one GPU core *GeForce GTX 1080 Ti*.

A.2 Components and Runtimes

Our pipeline consists of four steps. We provide the average runtime of processing a file of 1000 sentence file for each of them:

- **Attention matrices computation** is conducted on GPU for both selection and evaluation sets. 144 self-attention matrices are computed for each sentence and saved in npz file. This step takes approximately 25 minutes.
- **Modification of Universal Dependencies** is applied on heads selection and evaluation test (in the latter case only for evaluation of $DepAcc$). We use UDApi with our custom extension (`https://udapi.github.io`). The conversion of a CoNLL-U file takes a few seconds.
- **Head selection** is done on head selection set. The approximate runtime is 3 minutes.
- **Tree extraction** is performed on evaluation set. The approximate runtime is 10 minutes.

The code is available at GitHub: `https://github.com/Tom556/BERTHeadEnsembles`. For details, please refer to the README.

A.3 Data

Our pipeline requires CoNLL-U files as input. EuroParl parsed sentences used for head selection in English, German, French, Czech, and Finnish are provided in a zip file.

All other treebanks mentioned in this paper are available at Universal Dependencies webpage `https://universaldependencies.org`.

We perform head selection on the development part of data for Indonesian, Turkish, on train part for Korean and Japanese, due to small amount of development sentences for these two languages.

B Original UD Results

Dependency Accuracy results for English PUD treebank without our modification are presented in the table 5.

Relation label	Base-line	Orginal d2p	p2d	Modified d2p	p2d
amod	78.3	93.8	79.5	93.8	79.5
advmod	48.6	62.1	62.6	62.1	63.6
aux	65.2	93.4	83.1	94.5	88.0
case	36.2	88.4	68.9	88.4	68.9
compound	75.8	87.0	79.1	87.0	79.1
conjunct	27.8	59.0	47.1	58.8	51.3
det	56.5	97.2	69.4	97.2	69.4
nmod	25.7	49.1	54.7	49.1	54.7
nummod	57.5	79.3	72.6	79.3	72.6
mark	53.7	73.5	65.9	73.5	65.9
obj	39.2	90.8	80.7	89.3	78.5
nsubj	24.6	56.9	70.1	57.8	76.0
⇑ AVG. NON-CLAUSAL	49.1	73.4		74.1	
acl	29.7	50.5	49.0	50.5	43.8
advcl	8.2	40.4	27.7	40.7	26.3
csubj	23.3	58.6	34.5	24.1	31.0
x/ccomp	35.0	64.6	54.9	66.9	52.1
parataxis	4.1	16.5	13.2	23.1	24.2
⇑ AVG. CLAUSAL	24.7	41.0		38.3	
punct	9.3	27.7	41.6	28.4	44.0
dep	14.2	31.7	28.1	25.1	37.0

Table 5: Comparison of dependency accuracy for original and modified UD. Positional baseline was calculated on original UD. The evaluation was done using enBERT's head ensembles of size 4.

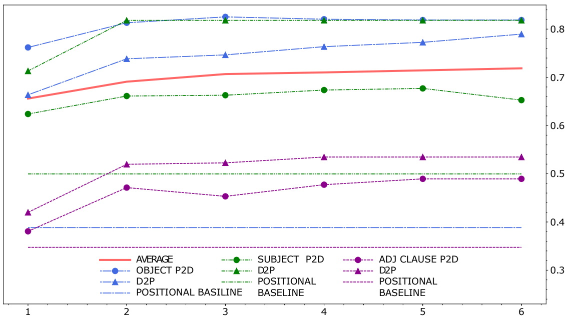


Figure 7: Dependency accuracy on the test set for different sizes of ensembles.

C Head Ensemble Size

In the Figure 7, we see that ensembles of just two heads have significantly higher dependency accuracy than single heads. For the most relation labels adding more heads does not affect the score, while for a few (object dependent to parent), it grows only slightly. As mentioned in the article, we set the number of heads in ensemble N to 4.

D Heads Visualization

This appendix contains an extended version of the Figure 1 from the article.

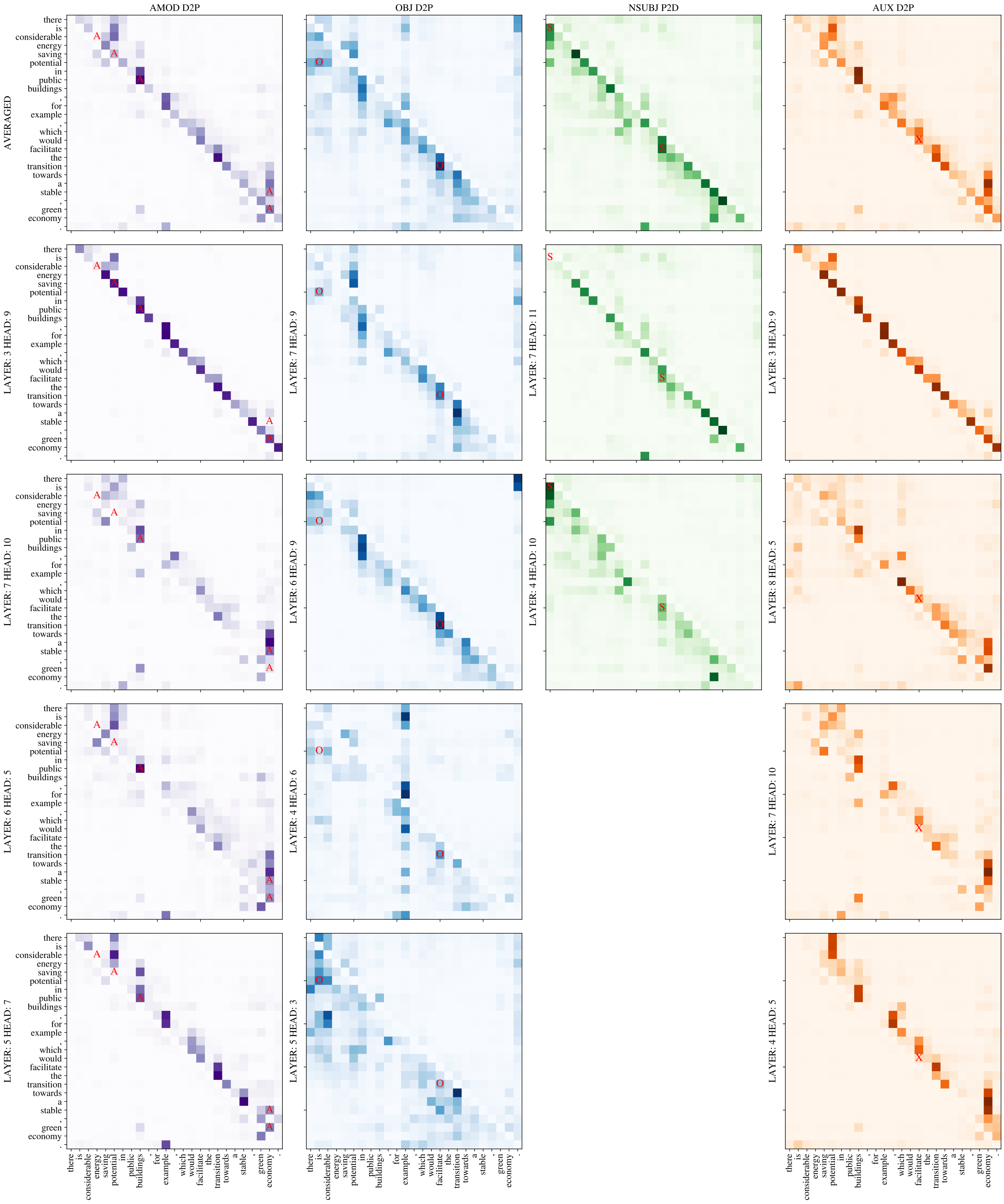


Figure 8: enBERT head ensembles for four dependency types: adjective modifier (d2p); object (d2p); nominal subject (p2d); auxiliary (d2p). The top row presents averaged attention. UD relations are marked by red crosses. The sentence: ”There is considerable energy saving potential inpublic buildings, for example, which would facilitatethe transition towards a stable, green economy.”