Probing for Labeled Dependency Trees

Max Müller-Eberstein and Rob van der Goot and Barbara Plank
Department of Computer Science
IT University of Copenhagen, Denmark
mamy@itu.dk, robv@itu.dk, bapl@itu.dk

Abstract

Probing has become an important tool for analyzing representations in Natural Language Processing (NLP). For graphical NLP tasks such as dependency parsing, linear probes are currently limited to extracting undirected or unlabeled parse trees which do not capture the full task. This work introduces DEPPROBE, a linear probe which can extract labeled and directed dependency parse trees from embeddings while using fewer parameters and compute than prior methods. Leveraging its full task coverage and lightweight parametrization, we investigate its predictive power for selecting the best transfer language for training a full bi-affine attention parser. Across 13 languages, our proposed method identifies the best source treebank 94% of the time, outperforming competitive baselines and prior work. Finally, we analyze the informativeness of task-specific subspaces in contextual embeddings as well as which benefits a full parser’s non-linear parametrization provides.

1 Introduction

Pre-trained, contextualized embeddings have been found to encapsulate information relevant to various syntactic and semantic tasks out-of-the-box (Tenney et al., 2019; Hewitt and Manning, 2019). Quantifying this latent information has become the task of probes — models which take frozen embeddings as input and are parametrized as lightly as possible (e.g. linear transformations). Recent proposals for edge probing (Tenney et al., 2019) and structural probing (Hewitt and Manning, 2019) have enabled analyses beyond classification tasks, including graphical tasks such as dependency parsing. They are able to extract dependency graphs from embeddings, however these are either undirected (Hewitt and Manning, 2019; Hall Maudslay et al., 2020) or unlabeled (Kulmizev et al., 2020), thereby capturing only a subset of the full task.

In this work, we investigate whether this gap can be filled and ask: Can we construct a lightweight probe which can produce fully directed and labeled dependency trees? Using these trees, we further aim to study the less examined problem of transferability estimation for graphical tasks, extending recent work targeting classification and regression tasks (Nguyen et al., 2020; You et al., 2021). Specifically: How well do our probe’s predictions correlate with the transfer performance of a full parser across a diverse set of languages?

To answer these questions, we contribute DEPPROBE (Figure 1), the first linear probe to extract directed and labeled dependency trees while using fewer parameters than prior work and three orders of magnitude fewer trainable parameters than a full parser (Section 3). As this allows us to measure labeled attachment scores (LAS), we investigate the degree to which our probe is predictive of cross-lingual transfer performance of a full parser across 13 typologically diverse languages, finding that our approach chooses the best transfer language 94% of the time, outperforming competitive baselines and prior work (Section 4). Finally, we perform an in-depth analysis of which latent information is most relevant for dependency parsing as well as which edges and relations benefit most from the expressivity of the full parser (Section 5).

1Code available at https://personads.me/x/acl-2022-code.
2 Related Work

Given the ubiquitous use of contextualized embeddings (Devlin et al., 2019; Conneau et al., 2020; Xue et al., 2021), practitioners have turned to various methods for analyzing their linguistic features (Rogers et al., 2020). Hewitt and Manning (2019) examine these intrinsic properties in greater detail for English dependency parsing using a structural probe, finding that undirected dependency graphs are recoverable from BERT by learning a linear transformation on its embeddings (Section 3.1).

Extending the structural probe of Hewitt and Manning (2019) to 12 languages, Chi et al. (2020) extract undirected dependency graphs from mBERT (Devlin et al., 2019), further showing that head-to-child difference vectors in the learned subspace cluster into relations from the Universal Dependencies taxonomy (de Marneffe et al., 2014).

Building on both the structural and tree depth probes (Hewitt and Manning, 2019), Kulmizev et al. (2020) extract directed dependency graphs from mBERT (Devlin et al., 2019), further showing that head-to-child difference vectors in the learned subspace cluster into relations from the Universal Dependencies taxonomy (de Marneffe et al., 2014).

Further variations to structural probing include regularization of the linear transformation (Limisiewicz and Mareček, 2021) as well as alternative objective functions (Hall Maudslay et al., 2020).

None of the proposed linear probing approaches so far are able to produce full dependency parse trees (i.e. directed and labeled), however the closer a probe approximates the full task, the better it quantifies relevant information (Hall Maudslay et al., 2020). It would for example be desirable to estimate LAS for parsing a target treebank with a model trained on a different source without having to train a resource-intensive parser (e.g. Dozat and Manning, 2017) on each source candidate. Although performance prediction methods for such scenarios exist, they typically do not cover graph prediction (Nguyen et al., 2020; You et al., 2021).

In order to bridge the gap between full parsers and unlabeled probes, in addition to the gap between full fine-tuning and lightweight performance prediction, this work proposes a linear probe which can extract labeled and directed dependency parse trees while using less compute than prior methods (Section 3). We use our probe’s LAS to evaluate its predictive power for full parser performance and leverage its linear nature to investigate how dependencies are represented in subspaces of contextual embeddings (Section 5).

3 Probing for Dependencies

In order to construct a directed and labeled dependency parse tree for a sentence $s$ consisting of the words $\{w_0, \ldots, w_N\}$, we require information on the presence or absence of edges between words, the directionality of these edges $(w_i, w_j)$, and the relationships $\{r_0, \ldots, r_N\}$ which they represent. Using the contextualized embeddings $\{h_0, \ldots, h_N\}$ with $h_i \in \mathbb{R}^e$, prior probing work has focused on the first step of identifying edges (Section 3.1) and later directionality (Section 3.2). In this work, we propose a probe which completes the final relational step (Section 3.3) and simultaneously provides a more efficient method for identifying directionality (Section 3.4).

3.1 Undirected Probing

The structural probe introduced by Hewitt and Manning (2019) recovers the first piece of information (i.e. the undirected graph) remarkably well. Here, the probe is a linear transformation $B \in \mathbb{R}^{e \times b}$ with $b < e$ which maps contextual embeddings into a subspace in which the distance measure

$$d_B(h_i, h_j) = \sqrt{(Bh_i - Bh_j)^T(Bh_i - Bh_j)}$$  \hspace{1cm} (1)$$

between $h_i$ and $h_j$ is optimized towards the distance between two words in the dependency graph $d_P(w_i, w_j)$, i.e. the number of edges between the words. For each sentence, the loss is defined as the mean absolute difference across all word pairs:

$$L_B(s) = \frac{1}{N^2} \sum_{i=0}^{N} \sum_{j=0}^{N} \left| d_P(w_i, w_j) - d_B(h_i, h_j) \right|.$$  \hspace{1cm} (2)$$

In order to extract an undirected dependency graph, one computes the distances for a sentence’s word pairs using $d_B$ and extracts the minimum spanning tree (Jarník, 1930; Prim, 1957; MST).

3.2 Directed Probing

Apart from the structural probe $B$, Hewitt and Manning (2019) also probe for tree depth. Using another matrix $C \in \mathbb{R}^{e \times c}$, a subspace is learned in which the squared $L_2$ norm of a transformed embedding $\|Ch_i\|_2^2$ corresponds to a word’s depth in the tree, i.e. the number of edges from the root.

Kulmizev et al. (2020) combine the structural and tree depth probe to extract directed graphs.
This directed probe (DIRPROBE) constructs a score matrix \( M \in \mathbb{R}^{N \times N} \) for which each entry corresponds to a word pair’s negative structural distance \(-dB(h_i, h_j)\). The shallowest node in the depth subspace \( C \) is set as root. Entries in \( M \) which correspond to an edge between \( w_i \) and \( w_j \) for which the word depths follow \(|Ch_i|^2 > |Ch_j|^2\) are set to \(-\infty\). A word’s depth in subspace \( C \) therefore corresponds to edge directionality. The directed graph is built from \( M \) using Chu-Liu-Edmonds decoding (Chu and Liu, 1965; Edmonds, 1967).

DIRPROBE extracts directed dependency parse trees, however it would require additional complexity to label each edge with a relation (e.g. using an additional probe). In the following, we propose a probe which can extract both directionality and relations while using fewer parameters and no dynamic programming-based graph-decoding algorithm.

### 3.3 Relational Probing

The incoming edge of each word \( w_i \) is governed by a single relation. As such the task of dependency relation classification with \( l \) relations can be simplified to a labeling task using a linear transformation \( L \in \mathbb{R}^{x \times l} \) for which the probability of a word’s relation \( r_i \) being of class \( l_k \) is given by:

\[
p(r_i = l_k | w_i) = \text{softmax}(Lh_i)_k
\]

and optimization uses standard cross-entropy loss given the gold label \( r^*_i \) for each word \( w_i \):

\[
\mathcal{L}_L(s) = -\frac{1}{N} \sum_{i=0}^{N} \ln p(r^*_i | w_i).
\]

Should dependency relations be encoded in contextualized embeddings, each dimension of the subspace \( L \) will correspond to the prevalence of information relevant to each relation, quantifiable using relation classification accuracy (RelAcc).

### 3.4 Constructing Dependency Parse Trees

Combining structural probing (Section 3.1) and dependency relation probing (Section 3.3), we propose a new probe for extracting fully directed and labeled dependency trees (DEPPROBE). It combines undirected graphs and relational information in a computationally efficient manner, adding labels while requiring less parameters than prior unlabeled or multi-layer-perceptron-based approaches.

As outlined in Algorithm 1 and illustrated in Figure 1, DEPPROBE uses the distance matrix \( D_B \) derived from the structural probe \( B \) in conjunction with the relation probabilities of the relational probe \( L \) (line 1). The graph is first rooted using the word \( w_r \) for which \( p(\text{root} | w_r) \) is highest (line 2). Iterating over the remaining words until all \( w_j \) are covered in \( T_w \), an edge is drawn to each word \( w_j \) from its head \( w_i \) based on the minimum distance in \( D_B \). The relation \( r_j \) for an edge \((w_j, w_i, r_j)\) is determined by taking the relation label \( l_k \) which maximizes \( p(r_j = l_k | w_j) \) with \( l_k \neq \text{root} \) (line 6). The edge is then added to the set of labeled tree edges \( T_e \). With edge directionality being inferred as simply pointing away from the root, this procedure produces a dependency graph that is both directed and labeled without the need for additional complexity, running in \( O(n^2) \) while dynamic programming-based decoding such as DIRPROBE have runtimes of up to \( O(n^3) \) (Stanojević and Cohen, 2021).

Constructing dependency trees from untuned embeddings requires the matrices \( B \) and \( L \), totaling \( e \cdot b + e \cdot l \) trainable parameters. Optimization can be performed using gradient descent on the sum of losses \( \mathcal{L}_B + \mathcal{L}_L \). With \( l = 37 \) relations in UD, this constitutes a substantially reduced training effort compared to prior probing approaches (with sub-space dimensionalities \( b \) and \( e \) typically set to 128) and multiple magnitudes fewer fine-tuned parameters than for a full biaffine attention parser.

### 4 Experiments

#### 4.1 Setup

**Parsers** In our experiments, we use the deep biaffine attention parser (BAP) by Dozat and Manning (2017) as implemented in van der Goot et al. (2021) as an upper bound for MLM-based pars-
ing performance. As it is closest to our work, we further reimplement DIRPROBE (Kulmizev et al., 2020) with \( b = 128 \) and \( c = 128 \). Note that this approach produces directed, but unlabeled dependency graphs. Finally, we compare both methods to our directed and labeled probing approach, DEP-PROBE with \( b = 128 \) and \( l = 37 \).

All methods use mBERT (Devlin et al., 2019) as their encoder \((c = 768)\). For BAP, training the model includes fine-tuning the encoder’s parameters, while for both probes they remain fixed and only the linear transformations are adjusted. This results in 183M tuned parameters for BAP, 197k for DIRPROBE and 127k for DEP-PROBE. Hyperparameters are set to the values reported by the authors,\(^2\) while for DEP-PROBE we perform an initial tuning step in Section 4.2.

**Target Treebanks**  As targets, we use the set of 13 treebanks proposed by Kulmizev et al. (2019), using versions from Universal Dependencies v2.8 (Zeman et al., 2021). They are diverse with respect to language family, morphological complexity and script (Appendix A). This set further includes EN-EWT (Silveira et al., 2014) which has been used in prior probing work for hyperparameter tuning, allowing us to tune DEP-PROBE on the same data.

**Metrics**  We report labeled attachment scores (LAS) wherever possible (BAP, DEP-PROBE) and unlabeled attachment scores (UAS) for all methods. For DEP-PROBE’s hyperparameters, we evaluate undirected, unlabeled attachment scores (UUAS) as well as relation classification accuracy (RelAcc). One notable difference to prior work is that we include punctuation both during training and evaluation — contrary to prior probing work which excludes all punctuation (Hewitt and Manning, 2019; Kulmizev et al., 2020; Hall Maudslay et al., 2020) — since we are interested in the full parsing task.

**Training**  Each method is trained on each target treebank’s training split and is evaluated on the test split. For cross-lingual transfer, models trained on one language are evaluated on the test splits of all other languages without any further tuning. For DEP-PROBE tuning (Section 4.2) we use the development split of EN-EWT.

BAP uses the training schedule implemented in van der Goot et al. (2021) while DIRPROBE and

\[ \text{DEP-PROBE} \] use AdamW (Loshchilov and Hutter, 2019) with a learning rate of \( 10^{-3} \) which is reduced by a factor of 10 each time the loss plateaus (see also Hewitt and Manning, 2019).

Both probing methods are implemented using PyTorch (Paszke et al., 2019) and use mBERT as implemented in the Transformers library (Wolf et al., 2020). Each model is trained with three random initializations of which we report the mean.

**4.2 DEP-PROBE Tuning**

As prior work has repeatedly found that MLM layers encode different linguistic information, the layers which are most relevant for a probe’s task are typically first identified (Tenney et al., 2019; Hewitt and Manning, 2019). Following this paradigm, we train DEP-PROBE on embeddings from each layer of mBERT. Layer 0 is equivalent to the first, non-contextualized embeddings while layer 12 is the output of the last attention heads. The probe is trained on EN-EWT and evaluated on its development split using UUAS for the structural transformation \( B \) (akin to Hewitt and Manning, 2019) as well as RelAcc for the relational transformation \( L \).

Figure 2 shows that structure is most prevalent around layer 6 at 78 UUAS, corroborating the 6–8 range identified by prior work (Tenney et al., 2019; Hewitt and Manning, 2019; Chi et al., 2020). Dependency relations are easiest to retrieve at around layer 7 with an accuracy of 86%. The standard deviation across initializations is around 0.1 in both cases. Based on these tuning results, we use layer 6 for structural probing and layer 7 for relational probing in the following experiments.

---

\(^2\)For better comparability, we use the best single layer reported by Kulmizev et al. (2020) instead of the weighted sum over all layers.

---

**Figure 2:** Layer-wise Performance on EWT (Dev) for DEP-PROBE as measured by UUAS for the structural probe \( B \) and RelAcc for the relational probe \( L \).
4.3 Parsing Performance

Figure 3 lists UAS for all methods and LAS for BAP and DEPProBE both on target-language test data (=L) and zero-shot transfer targets (~L). Table 3c further shows the mean results for each setting.

Unsurprisingly, the full parametrization of BAP performs best, with in-language scores of 88 LAS and 91 UAS. For zero-shot transfer, these scores drop to 35 LAS and 52 UAS, with some language pairs seeing differences of up to 85 points: e.g. JA → JA (93 LAS) versus AR → JA (8 LAS) in Figure 3a. This again confirms the importance of selecting appropriate source data for any given target.

Both probes, with their limited parametrization, fall short of the full parser’s performance, but still reach up to 73 LAS and 79 UAS. DIRProBE has a mean in-language UAS which is 3 points higher than for DEPProBE, attributable to the more complex decoder. Due to DIRProBE’s output structures being unlabeled, we cannot compare LAS.

DEPProBE reaches a competitive 67 UAS despite its much simpler decoding procedure and appears to be more stable for zero-shot transfer as it outperforms DIRProBE by around 2 UAS while maintaining a lower standard deviation. Most importantly, it produces directed and labeled parses such that we can fully compare it to BAP. Considering that DEPProBE has more than three orders of magnitude fewer tunable parameters, a mean in-language LAS of 60 is considerable and highlights the large degree of latent dependency information in untuned, contextual embeddings. For zero-shot transfer, the performance gap to BAP narrows to 13 LAS and 14 UAS.

4.4 Transfer Prediction

Given that DEPProBE provides a highly parameter-efficient method for producing directed, labeled parse trees, we next investigate whether its performance patterns are indicative of the full parser’s performance and could aid in selecting an appropriate source treebank for a given target without having to train the 183 million parameters of BAP.

Setup  Comparing UAS and LAS of BAP with respective scores of DEPProBE and DIRProBE, we compute the Pearson correlation coefficient ρ and the weighted Kendall’s τw (Vigna, 2015). The latter can be interpreted as corresponding to a co-

Figure 3: In-language and Cross-lingual Transfer Performance for 13 target treebanks (train → test) in UAS for BAP (fully tuned parser), DEPProBE, DIRProBE and LAS for BAP, DEPProBE (DIRProBE is unlabeled).
relation in $[-1, 1]$, and that given a probe ranking one source treebank over another, the probability of this higher rank corresponding to higher performance in the full parser is $\tau_{w}^{2}$. All reported correlations are significant at $p < 0.001$. Similarly, differences between correlation coefficients are also significant at $p < 0.001$ as measured using a standard Z-test. In addition to the probes, we also compare against a method commonly employed by practitioners by using the cosine similarity of typological features from the URIEL database as represented in lang2vec (Littell et al., 2017; L2V) between our 13 targets (details in Appendix A).

Results Table 1 shows that the L2V baseline correlates with final parser performance, but that actual dependency parses yield significantly higher correlation and predictive power. For UAS, we find that despite having similar attachment scores, DEPROBE performance correlates higher with BAP than that of DIRPROBE, both with respect to predicting the ability to parse any particular language as well as ranking the best source to transfer from. Using the labeled parse trees of DEPROBE results in almost perfect correlation with BAP’s LAS at $\rho = .97$ as well as a $\tau_{w}$ of .88, highlighting the importance of modeling the full task and including dependency relation information. Using Kendall’s $\tau_{w}$ with respect to LAS, we can estimate that selecting the highest performing source treebank from DEPROBE to train the full parser will be the best choice 94% of the time for any treebank pair.

5 Analysis

5.1 Tree Depth versus Relations

Why does DEPROBE predict transfer performance more accurately than DIRPROBE despite its simpler architecture? As each probe consists only of two matrices optimized to extract tree structural, depth or relational information, we can directly compare the similarity of all task-relevant parameters across languages against the full BAP’s cross-lingual performance.

In order to measure the similarity of probe matrices from different languages, we use mean subspace angles (Knyazev and Argentati, 2002; SSA), similarly to prior probing work (Chi et al., 2020). Intuitively, SSA quantifies the energy required to transform one matrix to another by converting the singular values of the transformation into angles between 0° and 90°. SSAs are computed for the structural probe (SSA-STRUCT) which is equivalent in both methods, DIRPROBE’s depth probe (SSA-DEPTH) and DEPROBE’s relational probe (SSA-REL). We use Pearson $\rho$ and the weighted Kendall’s $\tau_{w}$ to measure the correlation between cross-lingual probe SSAs and BAP performance. This allows us to investigate which type of information is most important for final parsing performance.

From Table 2, we can observe that SSAs between probes of different languages correlate less with transfer performance than UAS or LAS (Table 1), underlining the importance of extracting full parses. Among the different types of dependency information, we observe that SSAs between the relational probes used by DEPROBE correlate highest with final performance at .73 for LAS and .65 for UAS. Structural probing correlates significantly both with BAP’s LAS and UAS at .68 and .60 respectively, but to a lesser degree. Probes for tree depth have the lowest correlation at .62 for LAS and .53 for UAS. Despite tree depth being a distinctive syntactic feature for language pairs such as the agglutinative Turkish and the more function word-based English, depth is either not as relevant for BAP or may be represented less consistently in embeddings across languages, leading to lower correlation between SSAs and final performance.

| MODEL    | LAS   | UAS   |
|----------|-------|-------|
|          | $\rho$ | $\tau_{w}$ | $\rho$ | $\tau_{w}$ |
| L2V      | .86   | .72   | .80   | .70   |
| DIRPROBE | —     | —     | .91   | .81   |
| DEPROBE  | .97   | .88   | .94   | .85   |

Table 1: Transfer Correlation with BAP. Pearson $\rho$ and weighted Kendall’s $\tau_{w}$ for BAP’s LAS and UAS with respect to DIRPROBE’s UAS, DEPROBE’s UAS and LAS as well as lang2vec cosine similarity (L2V).

| MODEL     | LAS   | UAS   |
|-----------|-------|-------|
|           | $\rho$ | $\tau_{w}$ | $\rho$ | $\tau_{w}$ |
| SSA-STRUCT | .68   | .42   | .60   | .43   |
| SSA-DEPTH  | .62   | .34   | .53   | .35   |
| SSA-REL    | .73   | .55   | .65   | .53   |

Table 2: SSA Correlation with BAP. Pearson $\rho$ and weighted Kendall’s $\tau_{w}$ for BAP’s LAS and UAS with respect to subspace angles between structural (STRUCT), depth (DEPTH) and relation probes (REL).
5.2 Full Parser versus Probe

In the following analysis we investigate performance differences between the full BAP and DEP-PROBE across all 13 targets in order to identify finer-grained limitations of the linear approach and also which kinds of dependencies benefit from full parameter tuning and non-linear decoding.

Edge Length  Figure 5 shows offsets between gold and predicted head positions. The majority of heads are predicted correctly with a ratio of 92.1% for BAP and 69.7% for DEP-PROBE. Both methods are less accurate in predicting long-distance edges with length 150–250, resulting in offsets of ca. 100 (aggregated into < and > in Figure 5). Most likely, this is due to these edges’ overall sparsity in the data (only 6.7% of edges cover a distance of more than 10 tokens) as well as their higher overall subjective difficulty. Nonetheless, BAP is able to capture such dependencies more accurately as shown by its lower error rates for long edges compared to those of DEP-PROBE.

In addition to very distant head nodes, BAP also seems to recover more of the nuanced edges in the [-5, 5] interval. This range is particularly impactful for downstream performance as the edges in our target treebanks have a median length of 2 (mean length 3.62 with $\sigma = 5.70$). The structural probing loss (Equation 2) and the simple linear parametrization of the probe are able to capture a large number of these edges as evidenced by overall low error rates, but lack the necessary expressivity in order to accurately capture all cases.

Relations  Looking at RelAcc for each category in the UD taxonomy (de Marneffe et al., 2014) in Figure 4 allows us to identify where higher parametrization and more complex decoding are required for high parsing performance. While we again observe that performance on all relations is higher for BAP than for DEP-PROBE, a large subset of the relations is characterized by comparable or equivalent performance. These include simple punctuation (punct), but also the majority of function word relations such as aux, case, clf, det and mark as well as coordination (e.g. cc, conj). We attribute the high performance of DEP-PROBE on these relations to the fact that the words used to express them typically stem from closed classes and consequently similar embeddings: e.g., determiners “the/a/an” (EN), case markers “di/da” (IT).

Interestingly, some relations expressed through open class words are also captured by the linear probe. These include the modifiers advmod, amod and discourse as well as some nominal relations such as expl, nmod, nsubj and nummod. As prior work has identified PoS information in untuned embeddings (Tenney et al., 2019), the modifiers are likely benefiting from the same embedding features. The fact that DEP-PROBE nonetheless identifies syntax-specific relations such as nsubj, and to a lesser degree obj and obl, indicates the presence of context-dependent syntactic information in addition to PoS.

The larger the set of possible words for a relation, the more difficult it is to capture with the probe. The functional cop (copula) relation provides an informative example: In English (and related languages), it is almost exclusively assigned to the verb “be” resulting in 85% RelAcc, while in non-European languages such as Japanese it can be ascribed to a larger set which often overlaps with other relations (e.g. aux) resulting in 65% RelAcc. BAP adapts to each language by tuning all parameters while DEP-PROBE, using fixed embeddings, reaches competitive scores on European languages, but performs worse in non-European settings (details in Appendix B).

Besides capturing larger variation in surface forms, BAP also appears to benefit from higher expressivity when labeling clausal relations such as ccomp, csubj. These relations are often characterized not only by surface form variation, but also by PoS variation of head/child words and overlap with other relation types (e.g. clausal subjects stem from verbs or adjectives), making them difficult to distinguish in untuned embeddings. Simultaneously, they often span longer edges compared to determiners or other function words.

Another relation of particular importance is root as it determines the direction of all edges predicted by DEP-PROBE. An analysis of the 14% RelAcc difference to BAP reveals that both methods most frequently confuse root with relations that fit the word’s PoS, e.g. NOUN roots with nsubj or amod. For the majority PoS VERB (70% of all root), we further observe that DEP-PROBE predicts twice as many xcomp and parataxis confusions compared to BAP, likely attributable to their root-similar function in subclauses. Since their distinction hinges on context, the full parser, which also tunes the contextual encoder, is better equipped to differentiate between them.
With **DEP**ROBE, we have introduced a novel probing procedure to extract fully labeled and directed dependency trees from untuned, contextualized embeddings. Compared to prior approaches which only capture structures lacking labels, edge directionality or both, our method retains a simple linear parametrization which is in fact more lightweight and does not require complex decoders (Section 3).

To the best of our knowledge, this is the first linear probe which can be used to estimate LAS from untuned embeddings. Using this property, we evaluated the predictive power of DEP**ROBE** on cross-lingual parsing with respect to the transfer performance of a fully fine-tuned biaffine attention parser. Across the considered 169 language pairs, DEP**ROBE** is surprisingly effective: Its LAS correlates significantly ($p < 0.001$) and most highly compared with unlabeled probes or competitive language feature baselines, choosing the best source treebank in 94% of all cases (Section 4).

Leveraging the linearity of the probe to analyze structural and relational subspaces in mBERT embeddings, we find that dependency **relation** information is particularly important for parsing performance and cross-lingual transferability, compared to both tree depth and structure. DEP**ROBE**, which models structure and relations, is able to recover many functional and syntactic relations with competitive accuracy to the full BAP (Section 5).

Finally, the substantially higher efficiency of DEP**ROBE** with respect to time and compute make it suitable for accurate parsing performance prediction. As contemporary performance prediction methods lack formulations for graphical tasks and handcrafted features such as lang2vec are not available in all transfer settings (e.g. document domains, MLM encoder choice), we see linear approaches such as DEP**ROBE** as a valuable alternative.

---

**6 Conclusion**

The last category in which BAP outperforms DEP**ROBE** includes rare, treebank-specific relations such as **reparandum** (reference from a corrected word to an erroneous one). Again, the larger number of tunable parameters in addition to the non-linear decoding procedure of the full parser enable it to capture more edge cases while DEP**ROBE**’s linear approach can only approximate a local optimum for any relations which are represented non-linearly.

**Efficiency** When using a probe for performance prediction, it is important to consider its computational efficiency over the full parser’s fine-tuning procedure. In terms of tunable parameters, **DEP**ROBE** has 36% fewer parameters than DIR**PROBE** and three orders of magnitude fewer parameters than BAP. In practice, this translates to training times in the order of minutes instead of hours.

Despite its simple $O(n^2)$ decoding procedure compared to dynamic programming-based graph-decoding algorithms ($O(n^3)$), **DEP**ROBE** is able to extract full dependency trees which correlate highly with downstream performance while maintaining high efficiency (Section 4.4).
Acknowledgements

We would like to thank the NLPnorth group for insightful discussions on this work — in particular Elisa Bassignana and Mike Zhang. Additional thanks to ITU’s High-performance Computing Cluster team. Finally, we thank the anonymous reviewers for their helpful feedback. This research is supported by the Independent Research Fund Denmark (Danmarks Frie Forskningsfond; DFF) grant number 9063-00077B.

References

Maria Jesus Aranzabe, Aitziber Atutxa, Kepa Bengoetxea, Arantza Díaz de Ibarraza, lakes Goenaga, Koldo Gojenola, and Larraitz Uria. 2015. Automatic conversion of the Basque dependency treebank to universal dependencies. In Proceedings of the fourteenth international workshop on treebanks and linguistic theories (TLT14), pages 233–241.

Masayuki Asahara, Hiroshi Kanayama, Takaaki Tanaka, Yusuke Miyao, Sumire Uematsu, Shin suke Mori, Yuji Matsumoto, Mai Omura, and Yugo Murawaki. 2018. Universal Dependencies version 2 for Japanese. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan. European Language Resources Association (ELRA).

Cristina Bosco, Felice Dell’Orletta, Simonetta Montemagni, Manuela Sanguinetti, and Maria Simi. 2014. The EVALITA 2014 dependency parsing task. In EVALITA 2014 Evaluation of NLP and Speech Tools for Italian, pages 1–8. Pisa University Press.

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.

Yoeng-Jin Chu and Tseng-Hong Liu. 1965. On the shortest arborescence of a directed graph. Scientia Sinica, 14:1396–1400.

Jayeol Chun, Na-Rae Han, Jena D. Hwang, and Jinho D. Choi. 2018. Building Universal Dependency treebanks in Korean. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan. European Language Resources Association (ELRA).

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 8440–8451, Online. Association for Computational Linguistics.

Marie-Catherine de Marneffe, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, and Christopher D. Manning. 2014. Universal Stanford dependencies: A cross-linguistic typology. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14), pages 4585–4592, Reykjavik, Iceland. European Language Resources Association (ELRA).

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: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017. Conference Track Proceedings. OpenReview.net.

Kira Droganova, Olga Lyashevskaya, and Daniel Zeman. 2018. Data conversion and consistency of monolingual corpora: Russian UD treebanks. In Proceedings of the 17th international workshop on treebanks and linguistic theories (ltt 2018), 155, pages 53–66.

Jack Edmonds. 1967. Optimum branchings. Journal of Research of the National Bureau of Standards, B, 71:233–240.

Jan Hajíč, Otakar Smrž, Petr Zemánek, Petr Pajás, Jan Šnaidauf, Emanuel Beška, Jakub Kráčmar, and Kamila Hassanová. 2009. Prague Arabic dependency treebank 1.0.

Rowan Hall Maudslay, Josef Valvoda, Tiago Pimentel, Adina Williams, and Ryan Cotterell. 2020. A tale of a probe and a parser. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 7389–7395, Online. Association for Computational Linguistics.

Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérad-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. 2020. Array programming with NumPy. Nature, 585(7825):357–362.
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, Minneapolis, Minnesota. Association for Computational Linguistics.

J. D. Hunter. 2007. Matplotlib: A 2d graphics environment. Computing in Science & Engineering, 9(3):90–95.

Vojtěch Jarník. 1930. O jistém problému minimálním.(z dopisu panu o. boruvkovi). Práce moravské přírodovědecké společnosti, 6(4):57–63.

Andrew V Knyazev and Merico E Argentati. 2002. Principal angles between subspaces in an a-based scalar product: algorithms and perturbation estimates. SIAM Journal on Scientific Computing, 23(6):2008–2040.

Artur Kulmizev, Miryam de Lhoneux, Johannes Gontrum, Elena Fano, and Joakim Nivre. 2019. Deep contextualized word embeddings in transition-based and graph-based dependency parsing - a tale of two parsers revisited. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 2755–2768, Hong Kong, China. Association for Computational Linguistics.

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.

Tomasz Limisiewicz and David Mareček. 2021. Introducing orthogonal constraint in structural probes. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 428–442, Online. Association for Computational Linguistics.

Patrick Littell, David R. Mortensen, Ke Lin, Katherine Kairis, Carlisle Turner, and Lori Levin. 2017. URIEL and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pages 8–14, Valencia, Spain. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net.

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.

Cuong Nguyen, Tal Hassner, Matthias W. Seeger, and Cédric Archambeau. 2020. LEEP: A new measure to evaluate transferability of learned representations. In Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, volume 119 of Proceedings of Machine Learning Research, pages 7294–7305. PMLR.

Martha Palmer, Rajesh Bhattacharjee, Bhuvana Narasimhan, Owen Rambow, Dipi Misra Sharma, and Fei Xia. 2009. Hindi syntax: Annotating dependency, lexical predicate-argument structure, and phrase structure. In The 7th International Conference on Natural Language Processing, pages 14–17.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An imperative style, high-performance deep learning library. In Advances in Neural Information Processing Systems 32, pages 8024–8035. Curran Associates, Inc.

Robert Clay Prim. 1957. Shortest connection networks and some generalizations. The Bell System Technical Journal, 36(6):1389–1401.

Sampo Pyysalo, Jenna Kanerva, Anna Missilä, Veronika Laippala, and Filip Ginter. 2015. Universal Dependencies for Finnish. In Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015), pages 163–172, Vilnius, Lithuania. Linköping University Electronic Press, Sweden.

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.

Mo Shen, Ryan McDonald, Daniel Zeman, and Peng Qi. 2016. UD_Chinese-GSD. https://github.com/UniversalDependencies/UD_Chinese-GSD.

Natalia Silveira, Timothy Dozat, Marie-Catherine de Marneffe, Samuel Bowman, Miriam Connor, John Bauer, and Chris Manning. 2014. A gold standard dependency corpus for English. In Proceedings of the Ninth International Conference on Language
Sidney Facundes, Richard Farkas, Marlı́a Fernan-
da, Hector Fernandez Alcalde, Jennifer Fos-
ter, Cláudia Freitas, Kazunori Fujita, Katarína Gaj-
došová, Daniel Galbraith, Marcos Garcia, Moa
Gardenfors, Sebastian Garza, Fabrício Ferraz Ger-
ardi, Kim Gerdes, Filip Ginter, Gustavo Godoy,
Iakes Goenaga, Koldo Gojenola, Mendiš Gökkr-
mak, Yvoa Goldberg, Xavier Gómez Guinovart,
Berta González Saavedra, Bermudeta Gruciút, Ma-
tias Grioni, Loïc Grobol, Normunds Grūzitis, Bruno
Guillaume, Céline Guillot-Barbance, Tunga Günsôr,
Nizar Habash, Hinrik Hafsteinsson, Jan Hajič, Jan
Hajič jr., Mika Hääläinen, Linh Hà Mỳ, Na-
Rae Han, Muhammad Yudistira Hanifmuti, Sam
Hardwick, Kim Harris, Dag Haug, Johannes Hei-
necke, Oliver Hellwig, Felix Hennig, Barbor
Hałdká, Jaroslava Hlaváčová, Florinel Hociung,
Pet-
ter Höhle, Eva Huber, Jena Hwang, Takumi Ikeda,
Anton Karl Ingason, Radu Ion, Elena Iríma, Olájó-
đe Ishola, Kaoru Ito, Tomáš Jelinek, Apoorva Jha,
Anders Johannsen, Hildur Jónsdóttir, Fredrik Jør-
gensen, Markus Juutinen, Sarveswaran K, Hún
Kaškara, Andre Kaasen, Nadezhda Kabaea, Sy
Vain Kahané, Hiroshi Kanayama, Jenny Kanerva,
Neslihan Kara, Boris Katz, Tolga Kayadelen, Jes-
tica Kenney, Václava Kettnerová, Jesse Kirchner,
Elena Klementieva, Arne Kühn, Abdullatif Kök-
sal, Kamil Kopacewicz, Timo Korkiašangas, Na-
talija Kotsyba, Jolanta Kovalevskaitė, Simon Krek,
Parameswari Krishnamurthy, Özgün Kuyruç, As
Kuzgun, Sooyookyung Kwak, Veronika Laip-
na, Ana Lam, Lorenz Lambertino, Tatiana Lan-
do, Septina Dian Larasati, Alexei Lavrentiev,
John Lee, Phuong Lê Hông, Alessandro Lenci,
Saran Lertpradit, Herman Leung, Maria Levina,
Cheuk Ying Li, Josie Li, Keying Li, Yuan Li, Kyung-
Tae Lim, Bruna Lima Padovaní, Niko Miekka,
Karina Mischenkova, Margaretta Misrhapayeha-
vík, Anna Missilä, Cătălin Mititelu, Niko Mieka,
Karina Mischenkova, Margarita Misrhapayeha-
vík, Anna Missilä, Cătălin Mititelu, Maria Mitrofan,
Yusuke Miyao, Amir Hossein Mojiri, Hana Murak-
thur Mortensen, Bohdan Moskalévský, Károly Mus-
chnik, Robert Munro, Yugo Murawaki, Kaili
Miürisep, Pinky Nainwani, Mariam Nakhlé, Juan
Ignacio Narvarro Horúaček, Anna Nedoluzhko,
Gunta Nešpore-Bēržkalne, Manuela Nevaci, Luong
Nguyễn Thi, Huyễn Nguyên Thi Minh, Yoshihiro
Nikaido, Vitaly Nikolaev, Rattima Nitisaraj, Alireza
Nozad, Marat Niyazov, Shihori Nozumi, Sinta
Ojana Atul Kr. Ojha, Adédayo Olóókún, Mai
Omura, Emeka Onuegbu-
uzia, Şaziye Betül Özateş, Merve Özçelik, Arzu-
can Özgür, Balkız Öztürk Başaran, Hyunjai Hay-
ley Park, Niko Partanen, Elena Pascual, Marco
Passarotti, Agnieszka Patejuk, Guíherme Paulino-
Passos, Angelika Pełjak-Lapińska, Siyao Peng,
Cenel-Augusto Perez, Natalia Perkova, Guy Per-
rier, Slav Petrov, Daria Petrova, Jason Phelan, Jussi
Piitulainen, Tommi A Pirinen, Emily Pitler, Bar-
bara Plank, Thierry Poibeau, Larisa Ponomareva,
Martin Popel, Lauma Pretkalinina, Sophie Prévost,
Prokopis Prokopidas, Adam Przepiórkowski, Tiina
Puolakainen, Sampo Pyysalo, Peng Qi, Andriela
Rääabis, Alexandre Rademaker, Taraka Rama, Lo-
ganathan Ramasamy, Carlos Ramisch, Fam Rashed,
Mohammad Sadegh Rasooli, Vinit Ravishankar,
Livy Real, Petru Rebeja, Siva Reddy, Georg Rehm,
Ivan Riabov, Michael Rießler, Erika Rimkutė,
Larissa Rinaldi, Laura Rituma, Luisa Rocha, Eirikur
Rögnvaldsson, Mykhailo Romanenko, Rudolf Rosa,
Valentin Rosca, Davide Rovati, Olga Rudina, Jack
Rueter, Valðrín Rúnarsson, Shoval Saúde, Pegah
Safari, Benoit Sagot, Alexei Sahala, Shadi Saleh,
Alessio Salomoni, Tanja Samardžić, Stephanie Sam-
son, Manuela Sanguinetti, Ezgi Sanyurt, Dage Särg,
Baiba Saulite, Yann Sawanakunanon, Shefali Sax-
ena, Kevin Scannell, Salvatore Scarlata, Nathan
Schneider, Sebastian Schuster, Lane Schwartz,
Djamé Saddah, Wolfgang Secker, Mojgan Serají,
Mo Shen, Atsuko Shimagi, Hiroyuki Shirasu, Yana
Shishkina, Mun Shohibussirri, Dmitriy Sichinava,
Jaine Siewert, Einay Freyr Sigurðsson, Aline Silveira,
Natalia Silveira, Maria Simi, Rud Simionescu,
Katalin Simkö, Mária Šimková, Kiril Simov, Maria
Skachedubova, Aaron Smith, Isabela Soares-Bastos,
Carolyn Spadine, Rachele Sprungnoli, Steinhöfler Stei-
grímsdóttir, Antonio Stella, Milan Straka, Emmett
Strickland, Jana Strnadová, Alane Suhr, Yogi Les-
mana Sulestio, Umit Sulubacak, Shingo Suzuki,
Zsolt Szántó, Dimi Tuji, Yuta Takahashi, Fabio Tam-
burini, Mary Ann C. Tan, Takaaki Tanaka, Sam-
son Tellä, Isabelle Tellier, Mariella Testori, Gaul-
laume Thomas, Liisi Torga, Märsida Toska, Trond
Trosterud, Anna Trukhina, Reut Tsarfaty, Utku Türk,
Francis Tyers, Sumire Uematsu, Roman Untilov,
Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, An-
drius Utka, Somnya Vajjala, Rob van der Goot,
Martine Vanhove, Daniel van Niekerk, Gertjan van
Noord, Víctor Varga, Eric Villemonte de la Clerg-
erie, Veronika Vizecze, Natalia Vlasova, Aya Wakasa,
Joel C. Wallenberg, Lars Wallin, Abigail Walsh,
Jing Xian Wang, Jonathan North Washington, Max-
Joel C. Wallenberg, Lars Wallin, Abigail Walsh,
Jing Xian Wang, Jonathan North Washington, Max-
imilian Wendt, Paul Widner, Semy Williams, Matis
Wirén, Christian Wittern, Tsegay Woldemariam,
Alessio Salomoni, Tanja Samardžić, Stephanie Sam-
son, Manuela Sanguinetti, Ezgi Sanyurt, Dage Särg,
Baiba Saulite, Yann Sawanakunanon, Shefali Sax-
ena, Kevin Scannell, Salvatore Scarlata, Nathan
Schneider, Sebastian Schuster, Lane Schwartz,
Djamé Saddah, Wolfgang Secker, Mojgan Serají,
Mo Shen, Atsuko Shimagi, Hiroyuki Shirasu, Yana
Shishkina, Mun Shohibussirri, Dmitriy Sichinava,
Jaine Siewert, Einay Freyr Sigurðsson, Aline Silveira,
Natalia Silveira, Maria Simi, Rud Simionescu,
Katalin Simkö, Mária Šimková, Kiril Simov, Maria
Skachedubova, Aaron Smith, Isabela Soares-Bastos,
Carolyn Spadine, Rachele Sprungnoli, Steinhöfler Stei-
grímsdóttir, Antonio Stella, Milan Straka, Emmett
Strickland, Jana Strnadová, Alane Suhr, Yogi Les-
mana Sulestio, Umit Sulubacak, Shingo Suzuki,
Zsolt Szántó, Dimi Tuji, Yuta Takahashi, Fabio Tam-
burini, Mary Ann C. Tan, Takaaki Tanaka, Sam-
son Tellä, Isabelle Tellier, Mariella Testori, Gaul-
laume Thomas, Liisi Torga, Märsida Toska, Trond
Trosterud, Anna Trukhina, Reut Tsarfaty, Utku Türk,
Francis Tyers, Sumire Uematsu, Roman Untilov,
Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, An-
drius Utka, Somnya Vajjala, Rob van der Goot,
Martine Vanhove, Daniel van Niekerk, Gertjan van
Noord, Víctor Varga, Eric Villemonte de la Clerg-
erie, Veronika Vizecze, Natalia Vlasova, Aya Wakasa,
Joel C. Wallenberg, Lars Wallin, Abigail Walsh,
Jing Xian Wang, Jonathan North Washington, Max-
imilian Wendt, Paul Widner, Semy Williams, Matis
Wirén, Christian Wittern, Tsegay Woldemariam,
Alessio Salomoni, Tanja Samardžić, Stephanie Sam-
son, Manuela Sanguinetti, Ezgi Sanyurt, Dage Särg,
Baiba Saulite, Yann Sawanakunanon, Shefali Sax-
ena, Kevin Scannell, Salvatore Scarlata, Nathan
Schneider, Sebastian Schuster, Lane Schwartz,
Djamé Saddah, Wolfgang Secker, Mojgan Serají,
Mo Shen, Atsuko Shimagi, Hiroyuki Shirasu, Yana
Shishkina, Mun Shohibussirri, Dmitriy Sichinava,
Jaine Siewert, Einay Freyr Sigurðsson, Aline Silveira,
Natalia Silveira, Maria Simi, Rud Simionescu,
Katalin Simkö, Mária Šimková, Kiril Simov, Maria
Skachedubova, Aaron Smith, Isabela Soares-Bastos,
Appendix

A Experimental Setup

### Target Treebanks

Table 3: Target Treebanks based on Kulmizev et al. (2019) with language family (FAMILY) and total number of sentences (SIZE).

| TARGET | LANG | FAMILY | SIZE |
|--------|------|--------|------|
| AR-PADT | Arabic | Afro-Asiatic | 7.6k |
| EN-EWT | English | Indo-European | 16.6k |
| EU-BDT | Basque | Basque | 9.0k |
| FI-TDT | Finnish | Uralic | 15.1k |
| HE-HTB | Hebrew | Afro-Asiatic | 6.2k |
| HI-HDTB | Hindi | Indo-European | 16.6k |
| IT-ISDT | Italian | Indo-European | 14.1k |
| JA-GSD | Japanese | Japanese | 8.1k |
| KO-GSD | Korean | Korean | 6.3k |
| RU-SynTagRus | Russian | Indo-European | 61.9k |
| SV-Talbanken | Swedish | Indo-European | 6.0k |
| TR-IMST | Turkish | Turkic | 5.6k |
| ZH-GSD | Chinese | Sino-Tibetan | 5.0k |

Table 3: Target Treebanks

In order to ensure reproducibility for future work, we release the code for our methods and reimplementations in addition to token-level predictions (e.g. for significance testing) at https://personads.me/x/acl-2022-code.

B Additional Results

### Subspace Angles (SSA)

SSA values are converted from radians to degrees $\in [0, 90]$ for improved readability. Correlation in Table 2 is calculated based on negative SSA (Chi et al., 2020).

### Relation Accuracy (RelAcc)

RelAcc (Wolf et al., 2020) is used in Section 5.2 to analyze dependency relations which benefit from the full parametrization of BAP compared to the linear DEPROBE. Figures 7–19 show RelAcc per language in addition to the aggregated scores in Figure 4. As noted in Section 5.2, some relations such as cop differ substantially across languages with respect to their realization (e.g. surface form variation). Furthermore, the set of relations represented in each target treebank may differ, especially for specialized categories.
Figure 6: SSA of Probe Transformations in degrees across 13 target treebanks for the structural (SSA-STRUCT), depth (SSA-DEPTH) and relational probes (SSA-REL).

Figure 7: RelAcc of BAP and DEPPROBE on AR-PADT (Test) grouped according to UD taxonomy.

Figure 8: RelAcc of BAP and DEPPROBE on EN-EWT (Test) grouped according to UD taxonomy.

Figure 9: RelAcc of BAP and DEPPROBE on EU-BDT (Test) grouped according to UD taxonomy.

Figure 10: RelAcc of BAP and DEPPROBE on FI-TDT (Test) grouped according to UD taxonomy.
Figure 11: RelAcc of BAP and DEPPROBE on HE-HTB (Test) grouped according to UD taxonomy.

Figure 12: RelAcc of BAP and DEPPROBE on HI-HDTB (Test) grouped according to UD taxonomy.

Figure 13: RelAcc of BAP and DEPPROBE on IT-ISDT (Test) grouped according to UD taxonomy.

Figure 14: RelAcc of BAP and DEPPROBE on JA-GSD (Test) grouped according to UD taxonomy.

Figure 15: RelAcc of BAP and DEPPROBE on KO-GSD (Test) grouped according to UD taxonomy.
Figure 16: RelAcc of BAP and DEPProBE on RU-SynTagRus (Test) grouped according to UD taxonomy.

Figure 17: RelAcc of BAP and DEPProBE on SV-Talbanken (Test) grouped according to UD taxonomy.

Figure 18: RelAcc of BAP and DEPProBE on TR-IMST (Test) grouped according to UD taxonomy.

Figure 19: RelAcc of BAP and DEPProBE on ZH-GSD (Test) grouped according to UD taxonomy.