Pareto Probing: Trading Off Accuracy for Complexity

Tiago Pimentel[*,🔭] **Naomi Saphra**[*,👽] **Adina Williams**[🚀] **Ryan Cotterell**[🔭,🛸]

🔭University of Cambridge 👽University of Edinburgh 🚀Facebook AI Research 🛸ETH Zürich
tp472@cam.ac.uk, n.saphra@ed.ac.uk,
adinawilliams@fb.com, ryan.cotterell@inf.ethz.ch

Abstract

The question of how to probe contextual word representations for linguistic structure in a way that is both principled and useful has seen significant attention recently in the NLP literature. In our contribution to this discussion, we argue for a probe metric that reflects the fundamental trade-off between probe complexity and performance: the Pareto hypervolume. To measure complexity, we present a number of parametric and non-parametric metrics. Our experiments using Pareto hypervolume as an evaluation metric show that probes often do not conform to our expectations—e.g., why should the non-contextual fastText representations encode more morpho-syntactic information than the contextual BERT representations? These results suggest that common, simplistic probing tasks, such as part-of-speech labeling and dependency arc labeling, are inadequate to evaluate the linguistic structure encoded in contextual word representations. This leads us to propose full dependency parsing as a probing task. In support of our suggestion that harder probing tasks are necessary, our experiments with dependency parsing reveal a wide gap in syntactic knowledge between contextual and non-contextual representations. Our code can be found at https://github.com/rycolab/pareto-probing.

1 Introduction

Neural networks are a pillar of modern NLP systems. However, their inner workings are poorly understood; indeed, for this reason, they are often referred to as black-box systems (Psichogios and Ungar, 1992; Orphanos et al., 1999; Cauer et al., 2000). This lack of understanding, coupled with the rising adoption of neural NLP systems in both industry and academia, has fomented a rapidly growing literature devoted to "cracking open the black box," as it were (Alishahi et al., 2019; Linzen
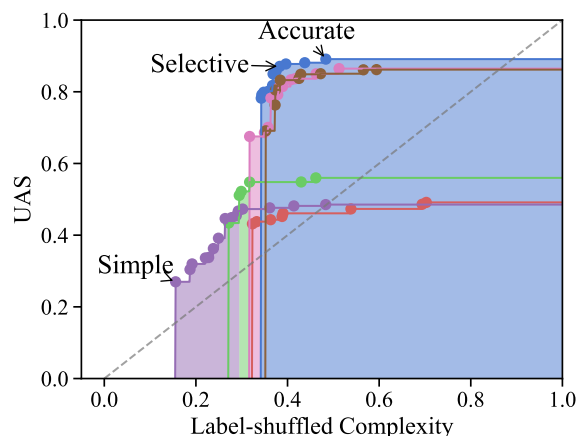
---

*Equal contribution

Figure 1: Probe results on dependency parsing in English. The x-axis corresponds to complexity and measures a probe's ability to memorize the training data. The y-axis measures the probes performance on the task. Probing the representations: ALBERT, BERT, RoBERTa, fastText, one-hot, and random.

et al., 2019). One popular method for studying the linguistic content of neural networks is **probing**, which we define in this work as training a supervised classifier (known as a **probe**) on top of pre-trained models' frozen representations (Alain and Bengio, 2017). By analyzing the classifier's performance, one can assess how much 'knowledge' the representations contain about language.

Much work in probing advocates for the need for simple probes (Hewitt and Manning, 2019; Hall Maudslay et al., 2020). Indeed, on this point, Alain and Bengio (2017) write:

> "The task of a deep neural network classifier is to come up with a representation for the final layer that can be easily fed to a linear classifier (i.e., the most elementary form of useful classifier)."

as a justification for their operationalization of **complexity** as the restriction of the probe to linear models (as opposed to deep neural networks). Most

saliently, Hewitt and Liang (2019) attempts to operationalize complexity in terms of control tasks, which constrain a probe's capacity for memorization.[1] Voita and Titov (2020) follow in this vein with an information-theoretic estimate of complexity: a model's minimum description length.

In opposition to the complexity of a probe is its **accuracy**, i.e., its ability to perform the target probing task. From an information-theoretic perspective, Pimentel et al. (2020) argues for the use of more complex probes, since they better estimate the amount of mutual information between a representation and the target linguistic property. From a different perspective, Saphra and Lopez (2019) also criticize the indiscriminate use of simple probes, because most neural representations are not estimated with the explicit aim of making information linearly separable; thus, it is unlikely that they will *naturally* do so, and foolish, perhaps, to expect them to.

This paper proposes to directly acknowledge the existence of a trade-off between the two when considering the development of probes. We argue—in part based on experimental evidence—that naïvely selecting a family of probes either for its complexity or its performance leads to degenerate edge-cases; see Fig. 1. We conclude that the nuanced trade-off between accuracy and complexity in probing should thus be treated as a bi-objective optimization problem: One objective encourages low complexity and another encourages high accuracy. We then propose a novel evaluation paradigm for probes. We advocate for **Pareto optimal probes**, i.e., probes that are both simpler and more accurate than all others. The set of such optimal probes can then be taken in aggregate to form a **Pareto frontier**, which allows for broader analysis and easier comparison between representations.

We run a battery of probing experiments for part-of-speech labeling and dependency-arc labeling, using both parametric and non-parametric complexity metrics. Our experiments show that if we desire simple probes, then we are forced to conclude that one-hot encoding representations and randomly generated ones almost always encode more linguistic structure than those representations derived from BERT—a nonsensical result. On the

other hand, seeking the most accurate probes is equivalent to performing NLP task-based research (e.g., part-of-speech tagging) in the classic way. We contend our Pareto curve–based measurements strike a reasonable balance.

To wrap up our paper, we levy a criticism at the probing tasks themselves; we argue that "toyish" probing tasks are not very useful for revealing how much more linguistic information BERT manages to capture than standard baseline representations. With this in mind, we advocate for more challenging probing tasks, e.g., dependency *parsing* instead of its toyish cousin dependency *arc labeling*. We find that using actual NLP tasks as probing tasks reveals much more about the advantages BERT provides over non-contextual representations.

2 Performance and Complexity

We argue in favor of treating probing for linguistic structure in neural representations as a two part optimization problem. On the one hand, we must optimize our probe for *high accuracy* on our chosen probing task: If we do not directly train the probe to accurately extract the linguistic features from the representation, how else can we determine whether they are implicitly encoded? On the other hand, the received wisdom in the probing community is that probes should be simple (Alain and Bengio, 2017; Hewitt and Manning, 2019): If the probe is an overly *complex* model, we might ascribe high accuracy on the probing task to the probe itself, meaning the probe has "learned the task" to a large extent. In this section, we argue that a probing framework that does not *explicitly* take into account the accuracy–complexity trade-off may be easily gamed. Indeed, we demonstrate how to game both accuracy and complexity respectively below.

2.1 The Nature of Probing Tasks

Most **probing tasks** are relatively "toy" in nature (Hupkes et al., 2018).[2] For instance, two of the most common probing tasks are **part-of-speech labeling** (POSL; Hewitt and Liang, 2019; Belinkov et al., 2017) and **dependency arc labeling** (DAL; Tenney et al., 2019a,b; Voita and Titov, 2020). Both tasks are treated as multi-way classification problems. POSL requires a model to assign a part-of-speech tag to a word in context *without* modeling the entire sequence of part-of-speech

---

[1] Hewitt and Liang (2019) define **selectivity** as the difference between a model's accuracy on a task versus its accuracy on a control version of that task. The control version of the tasks are built by randomly shuffling labels across word types and measures a probe's capacity for memorization. Our non-parametric measures of complexity differ from control tasks; we describe these differences in § 5.

[2] Not all though, several people have looked into e.g., parse tree reconstruction tasks (Jawahar et al., 2019; Hewitt and Manning, 2019; Vilares et al., 2020)

tags. Likewise, DAL requires a model to assign a dependency-arc label to an arc independently of the larger dependency tree. These word-oriented probing approaches force models to rely on information about context indirectly encoded in the feature vectors generated by the probed model. Importantly, both are simplified versions of their structured prediction cousins—part-of-speech tagging and dependency parsing—which require the modeling of entire sentences. Accuracy on POSL and DAL is then considered indicative of probed representations' "knowledge" of the linguistic structure encoded in the probing task. Limiting *explicit* access to context therefore allows an analysis constrained to how context is *implicitly* encoded in a particular representation. Furthermore, because POSL and DAL do not require complex structured prediction models, their simplicity is seen as a virtue to the mindset of disfavoring complexity (discussed further in § 2.3).

2.2 Optimizing for Performance

We first will argue that it is problematic to judge a probe either only by its performance on the probing task or by its complexity. Pimentel et al. (2020) showed that, under a weak assumption, any contextualized representation contains as much information about a linguistic task as the original sentence. They write:

> "under our operationalization, the endeavour of finding syntax in contextualized embeddings sentences is nonsensical. This is because, under Assumption 1, we know the answer *a priori*."

We agree that under their operationalization probing is nonsensical—purely optimizing for performance does not tell us anything about the representations, but only about the sentence itself.

Researchers, of course, have realized that choosing the most accurate probe is not wise for analysis; see Hewitt and Manning (2019) and the references therein for a good articulation of this point. To compensate for this tension, researchers have imposed explicit restrictions on the complexity of the probe, resulting in wider differences between contextual and non-contextual representations. Indeed, this is the logic behind the study of Hewitt and Liang (2019) who argue that **selective** probes should be chosen to judge whether the target linguistic property is well encoded in the representations. Relatedly, other researchers have explicitly focused on

linear classifiers as probes with the explicit reasoning that linear models are simpler than non-linear ones (Alain and Bengio, 2017; Hewitt and Manning, 2019; Hall Maudslay et al., 2020).

2.3 Reducing a Probe's Complexity

In § 2.2, we argued that solely optimizing for accuracy does not lead to a reasonable probing framework. Less commonly discussed, however, is that we also cannot directly optimize for simplicity. Let us consider the POSL probing task and the case where we are using a linear model as our probabilistic probe:

$$p(t \mid \mathbf{h}) = \mathrm{softmax}\left(\mathbf{Wh}\right) \qquad (1)$$

where $t \in \mathcal{T}$ is the target, e.g., a universal part-of-speech tag (Petrov et al., 2012), $\mathbf{h} \in \mathbb{R}^d$ is a contextual embedding and $\mathbf{W} \in \mathbb{R}^{|\mathcal{T}| \times d}$ is a linear projection matrix.

A natural measure of probe complexity in this framework is the **rank** of the projection matrix: $\mathrm{rank}(\mathbf{W})$. Indeed, this complexity metric was considered in one of the experiments in Hewitt and Manning (2019) to show that BERT representations strictly dominate ELMo representations for all ranks in their analyzed task. That experiment, though, left out some important baselines—the simplest of which is the encoding of words as one-hot representations. We take inspiration from those experiments and expand upon them (but rely instead on the nuclear norm as a convex relaxation of the matrix rank § 4) to produce the more complete plots in Figs. 2 and 3. These results are quite stunning; they show that, if we only cared about representations that simple probes could extract linguistic properties from, then a one-hot encoding of the word types is the best choice.

It is easy to see why the one-hot encoding does so well. For many of the toy probing tasks, the identity of the word is the single most important factor. It seems natural to expect that a low-complexity probe will be unable to exploit much more than a word's identity, so a one-hot embedding is really the best you can do—the word's identity is trivially encoded. Our point here is that both accuracy and complexity matter and neither can be sensibly optimized without the other.

3 An Invitation to Pareto Probing

We now advocate for a probing evaluation metric that combines both accuracy *and* complexity. We

argued in § 2 that probe accuracy and complexity exist in a trade-off. Because of this trade-off, we should search for models that are Pareto optimal. A probe is considered **Pareto optimal** (with respect to a family of probes) if there is no competing probe where both the accuracy is higher *and* the complexity is lower on the task. The set of Pareto optimal points may be called the **Pareto frontier** and is generally connected, as is shown in Fig. 2. As can also be seen in Fig. 2, we can compare different representations according to their Pareto frontiers. The set of representations that appear on the Pareto frontier should be sufficient–the other representations are **Pareto dominated**, since you can improve in one aspect (complexity or accuracy) without sacrificing the other. We call the set of representations which are on the frontier **Pareto dominant**.

We can also analyze each representation's frontier individually. This notion leads us to a very natural metric for evaluating probes: **Pareto hypervolume** (PH; Auger et al., 2012).[3] One important technical caveat involving evaluating the hypervolume is that it is undefined when the metric of model complexity for the experiment is unbounded. Thus, it is necessary to restrict model complexity to a bounded interval so that the PH is always finite.

4 Parametric Metrics of Complexity

We consider two types of probe complexity metrics. We term the first **parametric complexity**, which we discuss in this section. The second type is **non-parametric complexity**, which we discuss in § 5. For the parametric one, we first require a family of probes, e.g., the family of linear probes—which are all those that take the form of eq. (1), without restriction on the representation's dimension d.

4.1 Parametric Complexity for Linear Probes

In the case of linear probes, we explore two metrics of parametric complexity: the nuclear norm and rank. The **nuclear norm** is defined as

$$||\mathbf{W}||_* = \sum_{i=1}^{\min(|\mathcal{T}|,d)} \sigma_i(\mathbf{W}) \qquad (2)$$

where $\sigma_i(\mathbf{W})$ is the ith singular value of $\mathbf{W}$—which, in a way, measures the "size" of the matrix.

This yields the following objective for $\lambda \geq 0$:

$$\underbrace{-\sum_{i=1}^{n} \log p(t^{(i)} \mid \mathbf{h}^{(i)})}_{\text{cross-entropy}} + \lambda \cdot \underbrace{||\mathbf{W}||_*}_{\text{nuclear norm}} \qquad (3)$$

Training a probe to minimize this objective is equivalent to trading off its performance (high likelihood on the training data) for a lower complexity (nuclear norm of $\mathbf{W}$). This trade-off can be controlled through the hyper-parameter λ.

As a parametric complexity metric, we also consider the **rank** of the matrix. One definition of a matrix's rank is the number of non-zero singular values $\sigma_i(\mathbf{W})$. The rank can easily be restricted to a maximum value $r \in \mathbb{N}_+$ by splitting the matrix in two $\mathbf{W} = \mathbf{W}_l^\top \mathbf{W}_r$, where $\mathbf{W}_l \in \mathbb{R}^{r \times |\mathcal{T}|}$ and $\mathbf{W}_r \in \mathbb{R}^{r \times d}$. The nuclear norm is the tightest convex relaxation of the rank (Recht et al., 2010).[4]

While low-rank regularization is assumed to produce models that generalize better (Hinton and Van Camp, 1993; Langenberg et al., 2019), contrary to the classic bias–variance tradeoff, Goldblum et al. (2020) found that biasing towards small nuclear norms instead *hurts* generalization. Furthermore, our probe family consists of linear transformations, which are fed a relatively small number of features and trained with large training sets. As such, we are in an underfitting situation and any regularization should indeed hurt test performance.

4.2 Relation to Minimum Description Length

A recent proposal by Voita and Titov (2020) suggests that minimum description length (MDL; Rissanen, 1978) is a useful approach to the problem of balancing performance and complexity. The idea behind MDL is analogous to that of Bayesian evidence: We have a family of probabilistic models and a prior over those models. The likelihood term tells us how well we have coded the data and the prior term tells us the length of the model's code:

$$\underbrace{\prod_{i=1}^{n} p(t^{(i)} \mid \mathbf{h}^{(i)}, \mathbf{W})}_{\text{likelihood}} \times \underbrace{p(\mathbf{W})}_{\text{prior}} \qquad (4)$$

If we define our distribution over matrices as

$$p(\mathbf{W}) \propto \exp\left(-\lambda/2 \cdot ||\mathbf{W}||_*^2\right) \qquad (5)$$

---

[3]We note that we do not endorse only presenting PH scores, though, since it would again reduces this analysis to a single number. Such scores should be presented together with their Pareto curves to be maximally illustrative.

[4]Note that one may want to augment the linear probe with a padded vector, i.e., $\tilde{\mathbf{h}} = [\mathbf{h}; 1]$ to include a bias term in the model. In this case, our probe takes the form of $p(t \mid \mathbf{h}) = \text{softmax}(\mathbf{W}\,\tilde{\mathbf{h}})$ where, now, $\mathbf{W} \in \mathbb{R}^{|\mathcal{T}| \times (d+1)}$.

we recover our nuclear norm complexity term as the $\log$ of the prior. The distribution defined in eq. (5) is mathematically equivalent to the matrix normal distribution. To show this, we note that $-\frac{\lambda}{2}||\mathbf{W}||_*^2 = -\frac{1}{2}\text{tr}\left(\mathbf{W}^\top \left(\lambda^{-1}\mathbf{I}\right)^{-1} \mathbf{W}\right)$ and present the definition of the matrix normal (Gupta and Nagar, 2018, Chapter 2) as

$$p(\mathbf{W} \mid \mathbf{M}, \mathbf{V}, \mathbf{U}) = \qquad (6)$$
$$\frac{\exp\left(-\frac{1}{2}\text{tr}[\mathbf{V}^{-1}(\mathbf{W} - \mathbf{M})^\top \mathbf{U}^{-1}(\mathbf{W} - \mathbf{M})]\right)}{2\pi^{kd/2}\mathbf{V}^{k/2}\mathbf{U}^{d/2}}$$

where $k = |\mathcal{T}|$ and d are the sizes of matrix $\mathbf{W}$, $\mathbf{M}$ is the matrix's expected value, and $\mathbf{V}$ and $\mathbf{U}$ are analogous to the covariance matrices of typical Gaussian distributions. By setting the mean to the zero matrix, $\mathbf{V}$ to $\mathbf{I}$ (the identity matrix) and $\mathbf{U}$ to $\lambda^{-1}\mathbf{I}$, we recover eq. (5).

Naturally, there are many extensions within the MDL framework, e.g., variational coding MDL (Blier and Ollivier, 2018). In the case of non-linear models, Bayesian neural networks (Neal, 2012) are a natural choice. However, a fundamental problem will always remain—the results are dependent on the choice of prior. Indeed, in the simple case of linear probes, we can always "hack" the prior to favor certain probes over others that may not correspond to our intuitions of model complexity. For this reason, we also analyze a set of non-parametric metrics of complexity that do not require the probe user to pre-specify a prior over models.

5 Non-Parametric Metrics of Complexity

The parametric metrics of model complexity in § 4 have an explicit constraint that the models must belong to the same parametric family. Specifically, it requires that we are able to define a penalty (generally dependent on the parameters) that enforces how complex each model should be. In this section, we move away from parametric notions of model complexity to non-parametric metrics.

We opt to work with a notion of non-parametric complexity based on the ease with which a model can memorize training data. These non-parametric measures are rarely explicitly discussed as complexity metrics—although they are intuitive for that purpose—and have become common recently: Zhang et al. (2017) originated this trend by shuffling outputs of image data so the images were no longer predictive of the labels, using this result to illustrate the effective memorization capacity of modern neural networks. The first of our two

experiments to obtain non-parametric complexity measures is similar to theirs. We train our probe in a dataset with shuffled labels and get its accuracy in this training set. We will refer to this complexity metric as the **label-shuffled** scenario.[5]

Neural networks can take advantage of structured input (e.g., real images as opposed to noisy ones) to easily memorize their labels (Zhang et al., 2017). These structured inputs may be easier to represent internally regardless of the outputs, given current theories that early stages of training are committed to memorizing inputs (Arpit et al., 2017). As such, we may also want to analyze a probe's capacity to memorize unstructured input— in the case of language, we can easily remove structure by shuffling the word sequences themselves, creating random Zipfian-distributed noise, which are harder for neural networks to exploit (Liu et al., 2018). By providing probes with unstructured input, we measure a more domain-independent sense of complexity than the ability to map structured inputs to random labels, because the model cannot rely on syntactic patterns when memorizing shuffled training data. We will refer to this second scenario, wherein both labels and inputs are shuffled, as **fully shuffled**.

The distinction between memorization of real data and memorization of unstructured data is crucial, as experimenters choose the class of probes being learned. A comparison between label-shuffled and fully shuffled compression exposes the degree to which the class of probes employs a bias towards the true input structure in its compression. Similarly, comparisons between different classes of probes can test the same assumed bias.

We highlight that, while our non-parametric complexity metrics permit arbitrary classes of probes to be included in a probe hypothesis space, the selection criteria of possible probes may still reflect the assumed structure of the data, affecting compression and generalization. For example, linear probes reflect an assumption that the information lies in an Euclidean (sub)space; however, this assumption may not be true: Reif et al. (2019) reveal that a word's sequential position often rests on a
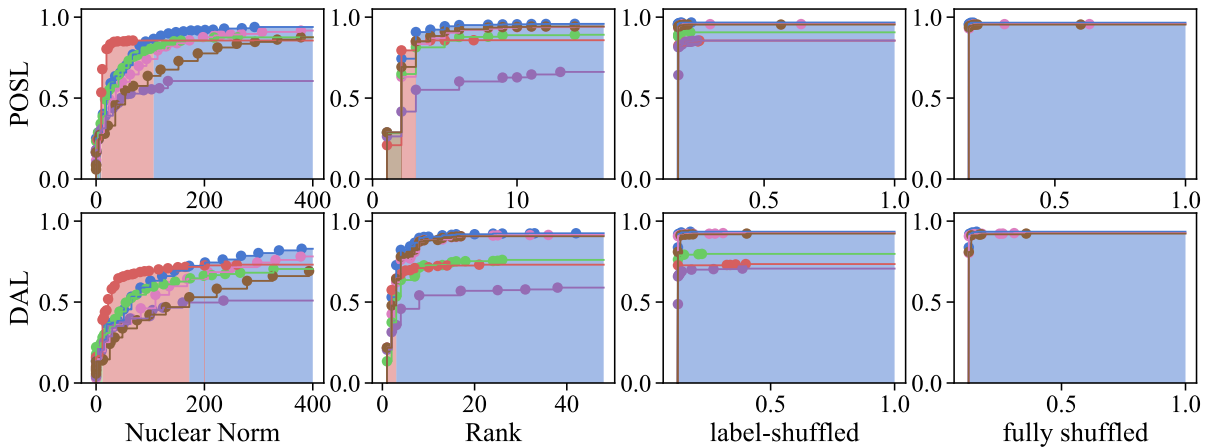

Figure 2: Pareto curves for experiments in English, using both parametric and non-parametric complexity metrics. The x-axis corresponds to the probe's complexity. The y-axis measures the probes accuracy on the task. Probing the representations: ALBERT, BERT, RoBERTa, fastText, one-hot, and random.

spiral manifold in BERT, while the syntactic distances described by Hewitt and Manning (2019) are Pythagorean in nature. One advantage of these methods is the ability to compare between probe classes, which offers a test of the geometric assumptions behind model selection.[6] Another advantage is that, unlike regularization-based parametric methods, they require no modification of the training procedure and can therefore run much faster.

6 POSL and DAL Experiments

We present our experimental findings on the previously discussed part-of-speech labeling (POSL) and dependency arc labeling (DAL) probing tasks using both our parametric complexity metrics (§ 4) and the non-parametric ones (§ 5). For both tasks, we use data from Universal Dependencies Treebanks version 2.5 (Zeman et al., 2019) and we probe a set of 5 typologically diverse languages: Basque (Aranzabe et al. 2015; BDT licensed under CC BY-NC-SA 3.0), English (Bies et al. 2012; Silveira et al. 2014; EWT licensed under CC BY-SA 4.0), Finnish (Haverinen et al. 2014, TDT licensed under CC BY-SA 4.0), Marathi (Ravishankar 2017, UFÁL licensed under CC BY-SA 4.0) and Turkish (Sulubacak et al. 2016, IMST licensed under CC BY-NC-SA 3.0). When investigating POSL, we take the target space $\mathcal{T}$ to be the set of universal part-of-speech tags for a specific language. We then train a classifier to predict these POS tags from word representations obtained from the

analyzed model (e.g., BERT). Similarly, for DAL, the target space $\mathcal{T}$ is defined as the set of arc dependency labels in the language, but we predict these labels from pairs of representations—the two words composing the arc.

We analyze the contextual representations from BERT (Devlin et al., 2019), ALBERT (Lan et al., 2020) and RoBERTa (Liu et al., 2019)—noting that ALBERT and RoBERTa are trained in English alone, so we only evaluate their performance on that language.[7] For each of these models, we feed it a sentence and average the output word piece (Wu et al., 2016) representations for each word, as tokenized in the treebank. We further analyze fastText's non-contextual representations (Bojanowski et al., 2017) as well as one-hot and random representations such as those considered by Pimentel et al. (2020). One-hot and random representations map each word type in the training data to a vector we sample from a standard normal distribution (zero mean and unit variance). New representations are sampled on the spot (untrained) for any out of vocabulary words. All representations are kept fixed during training, except for one-hot, which are learned with the other network parameters.

6.1 Linear Probes with Norm Constraints

For each language–representation–task triple, we train 100 linear probes, 50 optimizing eq. (3)[8] and 50 others with the rank constraint. The left-half of

---

[6]Another non-parametric method, online coding MDL (Voita and Titov, 2020) can likewise be compared across arbitrary model classes, because its complexity metric is based on probabilities produced and not probe parameters.

[7]We use the base versions (as opposed to the large ones) of ALBERT, RoBERTa and multilingual BERT—as implemented by Wolf et al. (2019)

[8]We vary λ in log-uniform intervals from 2^{-10} and 8.0—while also including experiments with no nuclear norm constraint (i.e., with $\lambda = 0$) for completion.
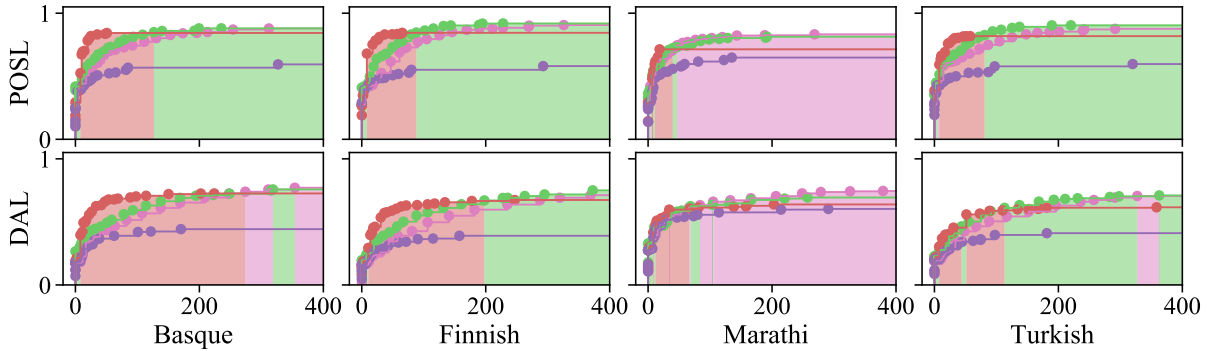
Figure 3: Results for multiple languages using the nuclear norm complexity metric. The x-axis corresponds to the probe's complexity. The y-axis measures the probes accuracy on the task. BERT, fastText, one-hot, and random.
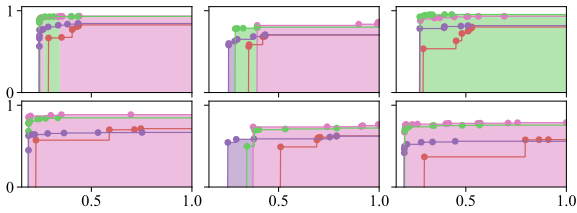

Figure 4: Results using the label-shuffled complexity metric in (left) Basque, (middle) Marathi, and (right) Turkish on (top) POSL, and (bottom) DAL. The x-axis corresponds to the probe's complexity. The y-axis measures the probes accuracy on the task. Probing the representations: BERT, fastText, one-hot, and random.

Fig. 2 presents the Pareto frontiers for both these probes trained on English, while Fig. 3 show the nuclear norm experiments in other languages.[9] As discussed in § 2.3, optimizing for complexity alone leads to trivial results—in all these languages one-hot representations would result in the best accuracy when using the nuclear norm complexity metric. We show that, counter-intuitively, fastText and one-hot representations Pareto-dominate BERT on the POSL task in Basque, Finnish and Turkish, producing higher accuracies with probes of any complexity (as defined by their nuclear norms). Thus, from the POSL experiments we cannot conclude BERT has any more syntactic information. In English, the one-hot and ALBERT representations form the Pareto-dominant set; the former in the simple scenario and the later in the complex scenario.

6.2 MLPs and Memorization

When using our non-parametric complexity metrics, we again train a number of classifiers for each

language–representation–task triple. The classifiers chosen for this analysis were multilayer perceptrons (MLP) with ReLU non-linearities. We trained 50 MLPs for each language–representation–task triple, sampling the number of layers uniformly from $[0, 5]$, the dropout from $[0.0, 0.5]$, and the hidden size log-uniformly from $[2^5, 2^{10}]$. Note that zero layers is a linear probe. (Our MLP is described in App. A.) Each of these architectures was trained both on the standard training set as in this set's label-shuffled and fully shuffled alternatives.

Fig. 4 presents POSL and DAL multilingual results under the non-parametric complexity metric; the right half of Fig. 2 presents English results.[10] The most striking characteristic of the Pareto frontiers is how simple architectures (i.e.,with relatively low memorization capacity) achieve as high an accuracy as the more complex ones. This is not surprising, though, when we compare this finding to the parametric ones; there we see linear probes are already almost as good as MLPs on these tasks. We take this as support for our intuition that toyish probing tasks are not very interesting or informative. We discuss this point in the next section.

7 A Call for Harder Probing Tasks

7.1 The False Promise of Toy Probing Tasks

In § 2.1, we reviewed arguments that researchers have put forth to justify toy tasks, while the argument for toy tasks from a standpoint of model complexity is addressed in § 2.3. Nevertheless, BERT, ELMo and other pre-trained representations rose to fame based on their ability to boost neural models

---

[9]Since rank constrained results showed a similar trend to the nuclear norm ones, results for other languages were moved into the appendices. The interested reader will also find zoomed-in versions (in the y-axis) of these plots there, as well as Pareto hypervolume tables.

[10]Experiments using the fully shuffled complexity metric only apply to contextual representations, since shuffling the sentence does not affect non-contextual ones. As such we only present them for English—we only analyze one contextual representation in other languages, i.e.,BERT.

Language	POSL	DAL
Basque	86%	67%
English	86%	68%
Finnish	87%	63%
Marathi	72%	62%
Turkish	83%	58%

Table 1: Test accuracies for a dictionary lookup strategy based on the labels in the training set.

to human-level scores on large, non-trivial tasks, e.g., natural language inference (Liu et al., 2019) and question answering (Lan et al., 2020)—with different performance patterns being observed on the toyish probing tasks.

As reported by Pimentel et al. (2020), BERT embeddings do not yield substantial improvements over non-contextual-embedding baselines, e.g., fastText, on toyish probing tasks. We reproduce similar experiments, albeit with our methodology, in § 6.2. In the case of POSL, we observe that fastText's embeddings achieve higher accuracy in many cases. In the case of DAL, however, we do observe that BERT leads to relatively small improvements over fastText across a typologically diverse set of languages. This result is not surprising because DAL is a more complex task than POSL: When one probes on simple tasks, models pretrained on more data do not help much. Furthermore, a quick visual analysis of § 6.1 reveals that one can achieve relatively high accuracy on both POSL and DAL with a linear probe. This is confirmed by § 6.2, which shows that simple probes, i.e.,probes with less capacity for memorization, result in as high accuracy as complex ones. In fact, we run an extra experiment, shown in Tab. 1, which shows that a trivial dictionary lookup strategy (details are presented in App. C) already achieves relatively high accuracies in POSL in all languages.

We interpret this to mean that current probing tasks are uninteresting—hiding from us the amount of syntactic information contextual representations actually encode. Furthermore, the simplicity of such toyish tasks artificially makes type-level embeddings—e.g., fastText—seem nearly as good as contextual ones.

7.2 Dependency Parsing

Following the previous argument, we believe *harder* probing tasks should be used. We take the lead by looking at dependency parsing, which
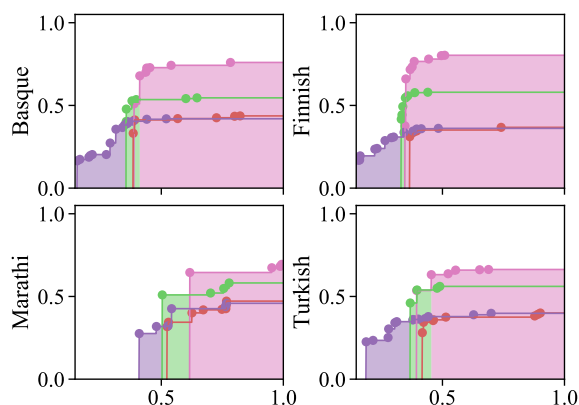

Figure 5: Dependency parsing Pareto curves using the label-shuffled complexity metric in a diverse set of languages. The x-axis corresponds to the probe's complexity. The y-axis measures the probes accuracy on the task. BERT, fastText, one-hot, and random.

Representation	Basque	English	Finnish	Marathi	Turkish
random	0.32	0.39	0.28	0.24	0.30
one-hot	0.26	0.32	0.22	0.20	0.22
fastText	0.35	0.40	0.38	**0.27**	0.35
BERT	**0.45**	**0.57**	**0.51**	0.25	**0.39**
ALBERT	-	0.57	-	-	-
RoBERTa	-	0.54	-	-	-

Table 2: Pareto hypervolume results on dependency parsing under the label-shuffled complexity metric.

depends on the whole sentence's context, and is much harder than toyish tasks like POSL and DAL. We train a simplified version of Dozat and Manning's (2017) biaffine parser, removing its power to process context by discarding its LSTM—as we describe in detail in App. B. This parser gives us the probability of a head for each word in a sentence, which allows us to recover the whole dependency tree. We then evaluate these trees using unlabeled attachment score (UAS). For our label-shuffled experiments, we permute the heads per sentence—creating non-tree dependencies.

Figs. 1 and 5 present label-shuffled results for this task. Such figures are much more interesting than the POSL and DAL ones, showing the expected trade-off between accuracy and complexity. Tab. 2 makes the amount of syntax encoded in contextual representations much clearer when compared to fastText. This is specially true if we compare these results to the Pareto hypervolumes of the POSL and DAL tasks (presented in Tab. 3 in the appendix). We take this experiment to conclude two things: (i) harder tasks are necessary to study neural representations; (ii) contextual representations encode much more knowledge about syntax (as expected) then do non-contextual ones.

8 A Closer Look at Model Complexity

This work represents a new entry into a growing literature on taking the capabilities of probes into account when analyzing a model (Hewitt and Liang, 2019; Voita and Titov, 2020; Whitney et al., 2020). The fundamental point we wish to espouse in this paper is that evaluating a probe for linguistic structure is fundamentally asking a question about a trade-off between accuracy and complexity. However, we wish to highlight that evaluating a probe's complexity is a very open problem. Indeed, the larger question of model complexity has been treated for over 50 years in a number of disciplines. In statistics, model complexity is researched in the model selection literature, e.g., the classical techniques of **Bayesian information criterion** (Schwarz, 1978) and **Akaike information criterion** (Akaike, 1974). In computer science, learning theorists have introduced the **Vapnik–Chervonenkis dimension** (Vapnik and Chervonenkis, 1971), **Pollard's pseudo-dimension** (Pollard, 1984), and **Rademacher complexity** (Bartlett and Mendelson, 2002). Algorithmic information theorists provide **Kolmogorov complexity** (Kolmogorov, 1963)—closely related to MDL—encoding the size of the model.

A concrete discussion of complexity requires several distinctions regarding these measures. The first is the **object of analysis** of the complexity measure which can be either a **model family**, i.e the whole set of functions realizable by a choice of architecture and hyperparameters, or a **learned model**, which takes into account a specific set of learned parameters. The second is the **aspect being analyzed** by the used measure which could be, for example, the **capacity** of the function class (i.e., whether it is possible to set model weights so the architecture represents a specific target function—its hard constraints) or its **bias** (i.e., the soft constraints that influence whether the training process is likely to guide a model towards this target function).

Importantly, the measure of complexity the scientist employs will impact their scientific findings about how much linguistic structure they read into a neural network's hidden states. For instance, in some of our experiments we regularize the probe directly with a relaxation of a nuclear norm constraint, thus imposing a bias without modifying the total capacity of the model. Meanwhile, our rank-based method controls the capacity of the probe directly, enforcing a model family's

complexity. Finally, our non-parametric methods, and selectivity, estimate the capacity of a model family (under a specific hyperparameter choice) by approximating a "hard" function in which labels are randomly assigned—differences in accuracy between these three complexity measures indicate a complex relationship between implicit regularization; input language structure; and model capacity. In comparison to our **computable** complexity measures, popular **hypothetical** notions also vary in how they explore the data's domain space to analyze a model family: Rademacher complexity uses true observations as inputs, but VC dimension considers adversarially selected data—being defined according to the "worst" possible sample allowed in the input domain.

As new techniques for considering the complexity of models emerge, it is critical to develop in parallel tools for reasoning about what aspect and object of "complexity" is really being measured. When one introduces a metric as modeling complexity, it can be explicitly situated within such a taxonomy; these considerations should be made explicit in the presentation. A sufficiently developed theory of probing will reveal not only the information contained in a representation, but the underlying geometry of the representation space, by comparing the performance of different model families. Such developments are left to future work.

9 Conclusion

In this paper, we argued for a new approach to probing, treating it as a bi-objective optimization problem. It has no single optimal solution, but can be analyzed—under the lens of Pareto efficiency—to arrive on a set of optimal solutions. These Pareto optimal solutions make explicit the trade-off between accuracy and complexity, also permitting for a deeper analysis of the probed representations.

The second part of our paper argues that we need to select harder tasks for the purpose of probing representations for syntactic knowledge. For tasks such as POSL or DAL, which require only shallow notions of syntax, non-contextual representations can do almost as well as contextual ones—pretraining on large amounts of data, or encoding contextual knowledge in the representations, does not help much for these tasks. We then run a battery of experiments on the harder task of dependency parsing; these show that contextual representations indeed provide much more usable syntactic knowledge than non-contextual ones.

Acknowledgments

References

Hirotugu Akaike. 1974. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.

Guillaume Alain and Yoshua Bengio. 2017. Understanding intermediate layers using linear classifier probes. In *The 5th International Conference on Learning Representations*.

Afra Alishahi, Grzegorz Chrupała, and Tal Linzen. 2019. Analyzing and interpreting neural networks for NLP: A report on the first BlackboxNLP workshop. *Natural Language Engineering*, 25(4):543–557.

Maria Jesus Aranzabe, Aitziber Atutxa, Kepa Bengoetxea, Arantza Diaz de Ilarraza, Iakes 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*, pages 233–241.

Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, and Simon Lacoste-Julien. 2017. A closer look at memorization in deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 233–242.

Anne Auger, Johannes Bader, Dimo Brockhoff, and Eckart Zitzler. 2012. Hypervolume-based multiobjective optimization: Theoretical foundations and practical implications. *Theoretical Computer Science*, 425:75–103.

Peter L. Bartlett and Shahar Mendelson. 2002. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482.

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.

Ann Bies, Justin Mott, Colin Warner, and Seth Kulick. 2012. English web treebank. *Linguistic Data Consortium*.

Léonard Blier and Yann Ollivier. 2018. The description length of deep learning models. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, pages 2216–2226.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Emil Cauer, Wolfgang Mathis, and Rainer Pauli. 2000. Life and work of Wilhelm Cauer (1900–1945). In *Proceedings of the 14th International Symposium Mathematical Theory of Networks and Systems*, pages 1–10.

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 *The 5th International Conference on Learning Representations*.

Micah Goldblum, Jonas Geiping, Avi Schwarzschild, Michael Moeller, and Tom Goldstein. 2020. Truth or backpropaganda? An empirical investigation of deep learning theory. In *International Conference on Learning Representations*.

Arjun K. Gupta and Daya K. Nagar. 2018. *Matrix Variate Distributions*, volume 104. CRC Press.

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*. Association for Computational Linguistics.

Katri Haverinen, Jenna Nyblom, Timo Viljanen, Veronika Laippala, Samuel Kohonen, Anna Missilä, Stina Ojala, Tapio Salakoski, and Filip Ginter. 2014. Building the essential resources for Finnish: the Turku dependency treebank. *Language Resources and Evaluation*, 48(3):493–531.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. 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 2733–2743, Hong Kong, China. 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, Minneapolis, Minnesota. Association for Computational Linguistics.

Geoffrey E. Hinton and Drew Van Camp. 1993. Keeping the neural networks simple by minimizing the description length of the weights. In *Proceedings of the Sixth Annual Conference on Computational Learning Theory*, pages 5–13.

Dieuwke Hupkes, Sara Veldhoen, and Willem Zuidema. 2018. Visualisation and 'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. *Journal of Artificial Intelligence Research*, 61:907–926.

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.

Andrei N. Kolmogorov. 1963. On tables of random numbers. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 369–376.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *The 8th International Conference on Learning Representations*.

Peter Langenberg, Emilio Rafael Balda, Arash Behboodi, and Rudolf Mathar. 2019. On the effect of low-rank weights on adversarial robustness of neural networks. *CoRR*, abs/1901.10371.

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

Nelson F. Liu, Omer Levy, Roy Schwartz, Chenhao Tan, and Noah A. Smith. 2018. LSTMs exploit linguistic attributes of data. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 180–186, Melbourne, Australia. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Radford M. Neal. 2012. *Bayesian Learning for Neural Networks*, volume 118. Springer Science & Business Media.

Giorgos Orphanos, Dimitris Kalles, A Papagelis, and Dimitris Christodoulakis. 1999. Decision trees and NLP: A case study in POS tagging. In *Proceedings of Annual Conference on Artificial Intelligence (ACAI)*.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 2089–2096, Istanbul, Turkey. European Languages Resources Association (ELRA).

Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. 2020. Information-theoretic probing for linguistic structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

David Pollard. 1984. *Convergence of Stochastic Processes*. Springer Science & Business Media.

Dimitris C. Psichogios and Lyle H. Ungar. 1992. A hybrid neural network-first principles approach to process modeling. *AIChE Journal*, 38(10):1499–1511.

Vinit Ravishankar. 2017. A Universal Dependencies treebank for Marathi. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 190–200, Prague, Czech Republic.

Benjamin Recht, Maryam Fazel, and Pablo A. Parrilo. 2010. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3):471–501.

Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B. Viegas, Andy Coenen, Adam Pearce, and Been Kim. 2019. Visualizing and measuring the geometry of BERT. In *Advances in Neural Information Processing Systems 32*, pages 8594–8603.

Jorma Rissanen. 1978. Modeling by shortest data description. *Automatica*, 14(5):465–471.

Naomi Saphra and Adam Lopez. 2019. Understanding learning dynamics of language models with SVCCA. 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 3257–3267, Minneapolis, Minnesota. Association for Computational Linguistics.

Gideon Schwarz. 1978. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464.

Natalia Silveira, Timothy Dozat, Marie-Catherine de Marneffe, Samuel Bowman, Miriam Connor, John Bauer, and Christopher D. Manning. 2014. A gold standard dependency corpus for English. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*.

Umut Sulubacak, Memduh Gokirmak, Francis Tyers, Çağrı Çöltekin, Joakim Nivre, and Gülşen Eryiğit. 2016. Universal Dependencies for Turkish. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3444–3454, Osaka, Japan. The COLING 2016 Organizing Committee.

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

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R. Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. 2019b. What do you learn from context? Probing for sentence structure in contextualized word representations. In *The 7th International Conference on Learning Representations*.

Vladimir N. Vapnik and Alexey Chervonenkis. 1971. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and Its Applications*, 16:264–280.

David Vilares, Michalina Strzyz, Anders Søgaard, and Carlos Gómez-Rodríguez. 2020. Parsing as pretraining. In *Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20)*.

Elena Voita and Ivan Titov. 2020. Information-theoretic probing with minimum description length. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

William F. Whitney, Min Jae Song, David Brandfonbrener, Jaan Altosaar, and Kyunghyun Cho. 2020. Evaluating representations by the complexity of learning low-loss predictors. *arXiv preprint arXiv:2009.07368*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. HuggingFace's Transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Daniel Zeman, Joakim Nivre, Mitchell Abrams, Noëmi Aepli, Željko Agić, Lars Ahrenberg, Gabrielė Aleksandravičiūtė, Lene Antonsen, Katya Aplonova, Maria Jesus Aranzabe, Gashaw Arutie, Masayuki Asahara, Luma Ateyah, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, Elena Badmaeva, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, Verginica Barbu Mititelu, Victoria Basmov, Colin Batchelor, John Bauer, Sandra Bellato, Kepa Bengoetxea, Yevgeni Berzak, Irshad Ahmad Bhat, Riyaz Ahmad Bhat, Erica Biagetti, Eckhard Bick, Agnė Bielinskienė, Rogier Blokland, Victoria Bobicev, Loïc Boizou, Emanuel Borges Völker, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Adriane Boyd, Kristina Brokaitė, Aljoscha Burchardt, Marie Candito, Bernard Caron, Gauthier Caron, Tatiana Cavalcanti, Gülşen Cebiroğlu Eryiğit, Flavio Massimiliano Cecchini, Giuseppe G. A. Celano, Slavomír Čéplö, Savas Cetin, Fabricio Chalub, Jinho Choi, Yongseok Cho, Jayeol Chun, Alessandra T. Cignarella, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Marine Courtin, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Elvis de Souza, Arantza Diaz de Ilarraza, Carly Dickerson, Bamba Dione, Peter Dirix, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Hanne Eckhoff, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Olga Erina, Tomaž Erjavec, Aline Etienne, Wograine Evelyn, Richárd Farkas, Hector Fernandez Alcalde, Jennifer Foster, Cláudia Freitas, Kazunori Fujita, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Moa Gärdenfors, Sebastian Garza, Kim Gerdes, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta González Saavedra, Bernadeta Griciūtė, Matias Grioni, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Nizar Habash, Jan Hajič, Jan Hajič jr., Mika Hämäläinen, Linh Hà Mỹ, Na-Rae Han, Kim Harris, Dag Haug, Johannes Heinecke, Felix Hennig, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Jena Hwang, Takumi Ikeda, Radu Ion, Elena Irimia, Ọlájídé Ishola, Tomáš Jelínek, Anders Johannsen, Fredrik Jørgensen, Markus Juutinen, Hüner Kaşıkara, Andre Kaasen, Nadezhda Kabaeva, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Boris Katz, Tolga Kayadelen, Jessica Kenney, Václava Kettnerová, Jesse Kirchner, Elena Klementieva, Arne Köhn, Kamil Kopacewicz, Natalia Kotsyba, Jolanta Kovalevskaitė, Simon Krek, Sookyoung Kwak, Veronika Laippala, Lorenzo Lambertino, Lucia Lam, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Phương Lê Hồng, Alessandro Lenci, Saran Lertpradit, Herman Leung, Cheuk Ying Li, Josie Li, Keying Li, KyungTae Lim, Maria Liovina, Yuan 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, Sarah McGuinness, Gustavo Mendonça, Niko Miekka, Margarita Misirpashayeva, Anna Missilä, Cătălin Mititelu, Maria Mitrofan, Yusuke Miyao, Simonetta Montemagni, Amir More, Laura Moreno Romero, Keiko Sophie Mori, Tomohiko Morioka, Shinsuke Mori, Shigeki Moro, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Robert Munro, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Juan Igna-

cio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, Lương Nguyễn Thị, Huyền Nguyễn Thị Minh, Yoshihiro Nikaido, Vitaly Nikolaev, Rattima Nitisaroj, Hanna Nurmi, Stina Ojala, Atul Kr. Ojha, Adédayọ̀ Olúòkun, Mai Omura, Petya Osenova, Robert Östling, Lilja Øvrelid, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Guilherme Paulino-Passos, Angelika Peljak-Łapińska, Siyao Peng, Cenel-Augusto Perez, Guy Perrier, Daria Petrova, Slav Petrov, Jason Phelan, Jussi Piitulainen, Tommi A Pirinen, Emily Pitler, Barbara Plank, Thierry Poibeau, Larisa Ponomareva, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Tiina Puolakainen, Sampo Pyysalo, Peng Qi, Andriela Rääbis, Alexandre Rademaker, Loganathan Ramasamy, Taraka Rama, Carlos Ramisch, Vinit Ravishankar, Livy Real, Siva Reddy, Georg Rehm, Ivan Riabov, Michael Rießler, Erika Rimkutė, Larissa Rinaldi, Laura Rituma, Luisa Rocha, Mykhailo Romanenko, Rudolf Rosa, Davide Rovati, Valentin Roșca, Olga Rudina, Jack Rueter, Shoval Sadde, Benoît Sagot, Shadi Saleh, Alessio Salomoni, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Dage Särg, Baiba Saulīte, Yanin Sawanakunanon, Nathan Schneider, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Mo Shen, Atsuko Shimada, Hiroyuki Shirasu, Muh Shohibussirri, Dmitry Sichinava, Aline Silveira, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Aaron Smith, Isabella Soares-Bastos, Carolyn Spadine, Antonio Stella, Milan Straka, Jana Strnadová, Alane Suhr, Umut Sulubacak, Shingo Suzuki, Zsolt Szántó, Dima Taji, Yuta Takahashi, Fabio Tamburini, Takaaki Tanaka, Isabelle Tellier, Guillaume Thomas, Liisi Torga, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Francis Tyers, Sumire Uematsu, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Andrius Utka, Sowmya Vajjala, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Eric Villemonte de la Clergerie, Veronika Vincze, Lars Wallin, Abigail Walsh, Jing Xian Wang, Jonathan North Washington, Maximilan Wendt, Seyi Williams, Mats Wirén, Christian Wittern, Tsegay Woldemariam, Tak-sum Wong, Alina Wróblewska, Mary Yako, Naoki Yamazaki, Chunxiao Yan, Koichi Yasuoka, Marat M. Yavrumyan, Zhuoran Yu, Zdeněk Žabokrtský, Amir Zeldes, Manying Zhang, and Hanzhi Zhu. 2019. Universal dependencies 2.5. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2017. Understanding deep learning requires rethinking generalization. In *The 5th International Conference on Learning Representations*.

A Multilayer Perceptron

In this section, we detail the architecture of the MLP classifiers. We define an MLP block as a series of non-linear transformations:

$$\mathbf{h}^{(\ell)} = \mathrm{ReLU}(\mathbf{W}^{(\ell)}\,\mathbf{h}^{(\ell-1)}) \qquad (7)$$

where $\mathbf{h}^{(0)} = \mathbf{h}$ is the initial representation. The final representation $\mathbf{h}^{(L)}$ is taken to be the output of an MLP block with L layers. An MLP block is parameterized by a matrix $\mathbf{W}^\ell \in \mathbb{R}^{d_\ell \times d_{\ell-1}}$, where d_0 is $\mathbf{h}$'s dimension, d_1 is the MLP's hidden size (as sampled according to § 6.2), and $d_\ell = {}^{d_{\ell-1}}/2$ for other layers. We write: $\mathbf{h}^{(L)} = \mathrm{MLP}(\mathbf{h})$. The hidden state $\mathbf{h}^{(L)}$ is then linearly transformed and normalized with a softmax to obtain a probability distribution for either the POSL or DAL tasks:

$$p(t \mid \mathbf{h}) = \mathrm{softmax}(\mathbf{W}^{\mathrm{out}}\,\mathbf{h}^{(L)}) \qquad (8)$$

B Dependency Parser

The dependency parser we make use of is a simplified version of Dozat and Manning's (2017) biaffine parser. The primary simplification is achieved by removing the LSTM, which reduces its complexity as well as restricts its access to context. We train two MLPs with the same architecture as the one described in § 6.2. Let $[\mathbf{h}_1, \ldots, \mathbf{h}_{|s|}]$ be a sentence representation. The first MLP is used to model a head dependency representation and the second is used to model a tail dependency relation. The output of both MLPs is expressed as follows

$$\mathbf{h}_{i,\mathrm{head}} = \mathrm{MLP}_{\mathrm{head}}(\mathbf{h}_i) \qquad (9)$$
$$\mathbf{h}_{i,\mathrm{tail}} = \mathrm{MLP}_{\mathrm{tail}}(\mathbf{h}_i)$$

which yields a pair of column vectors. We further define a biaffine function of the two representations as follows

$$l_{ij} = \mathbf{h}_{i,\mathrm{head}}^\top \mathbf{W}\,\mathbf{h}_{j,\mathrm{tail}} \qquad (10)$$

Finally, the output of this biaffine projection is used to construct the probability of i being the (only) head of j, i.e.,

$$p_{\mathrm{parse}}(i \text{ is the head of } j) = \frac{e^{l_{ij}}}{\sum_{i'=1}^{|s|} e^{l_{i'j}}} \qquad (11)$$

Such a parser does not enforce a spanning tree constraint during estimation. In our experiments, we simply predict each word's head to be the

word which maximizes $p_{\text{parse}}(i$ is the head of $j)$. Thus, our predictions will not necessarily be actual trees. We note, however, that given a trained parser p_{parse}, a dependency tree could be constructed using Edmonds' algorithm.

C   Lookup Model

In this section, we present the design of a very simple lookup model for the POSL and DAL tasks. We present the detailed implementation of both models below, but their general idea is looking at the training set for an instance's most frequent label, and falling back to an overall general label frequency in case it is not found.

POSL.   In this task, the lookup model has two behaviors: (i) if a word appears in the training set, it guesses its most common label; (ii) if an out-of-vocabulary word, it guesses the most overall frequent label in the training set.

DAL.   In the DAL task, the lookup model has four behaviors: (i) for an arc which appear in the training set, it guesses its most common label; (ii) for an unknown arc, it guesses the most overall frequent label in the training set for the arc's tail word; (iii) if the tail of the arc is an an out of vocabulary word, it guesses the most frequent label in the training set for the head word; (iv) finally, if the head is also out of vocabulary word, it guesses the most overall frequent arc label in the training set.

D   Detailed Results

In this section, we present further results which did not fit into the main text. We initially present results in dependency parsing, using the nuclear norm as our parametric complexity metric. Figs. 7 and 8 show that, again, the one-hot representation produces the best results in highly constrained scenarios (i.e., with very simple probes). Furthermore, comparing these results with § 7.2 we see that linear probes cannot do as well as MLPs in this task, suggesting it is indeed harder. Fig. 6 presents fully shuffled results for dependency parsing in English. Comparing it to § 7.2 we see the probes' capacity to memorize is much smaller on unstructured input.

Tab. 3 presents the Pareto hypervolume for all the analyzed models, in all languages, for POSL and DAL. Analyzing this table we again see that contextual representations do not improve over the non-contextual ones on these tasks by much—even producing worse results in some.

Finally, Fig. 9 presents results using the Rank parametric complexity metric, while Fig. 10 presents zoomed-in (in the y-axis) results for the Nuclear Norm parametric complexity metric.
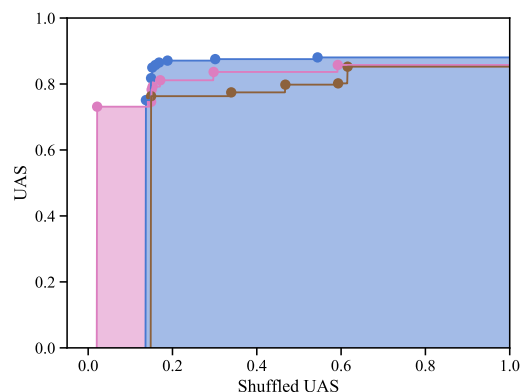

Figure 6: Pareto curves on dependency parsing with the fully shuffled complexity metric in English.

Metric	Representation	POSL					DAL				
		Basque	English	Finnish	Marathi	Turkish	Basque	English	Finnish	Marathi	Turkish
Nuclear Norm	BERT	0.85	0.91	0.86	0.80	0.84	0.79	0.87	0.77	0.73	0.71
Nuclear Norm	fastText	0.86	0.87	0.90	0.78	0.88	0.76	0.74	0.77	0.69	0.70
Nuclear Norm	one-hot	0.84	0.85	0.84	0.70	0.81	0.72	0.73	0.67	0.63	0.61
Nuclear Norm	random	0.58	0.63	0.57	0.62	0.58	0.47	0.56	0.42	0.59	0.43
Nuclear Norm	ALBERT	-	0.93	-	-	-	-	0.89	-	-	-
Nuclear Norm	RoBERTa	-	0.87	-	-	-	-	0.79	-	-	-
Rank	BERT	0.78	0.80	0.80	0.74	0.75	0.78	0.85	0.80	0.71	0.69
Rank	fastText	0.76	0.77	0.78	0.70	0.76	0.71	0.70	0.74	0.66	0.65
Rank	one-hot	0.74	0.76	0.73	0.62	0.70	0.65	0.69	0.61	0.57	0.53
Rank	random	0.53	0.54	0.51	0.58	0.52	0.44	0.53	0.41	0.55	0.41
Rank	ALBERT	-	0.84	-	-	-	-	0.87	-	-	-
Rank	RoBERTa	-	0.81	-	-	-	-	0.85	-	-	-
label-shuffled	BERT	0.71	0.79	0.71	0.50	0.67	0.72	0.80	0.74	0.46	0.63
label-shuffled	fastText	0.70	0.75	0.71	0.57	0.69	0.68	0.69	0.72	0.46	0.60
label-shuffled	one-hot	0.57	0.70	0.59	0.45	0.52	0.50	0.64	0.53	0.28	0.30
label-shuffled	random	0.63	0.70	0.62	0.52	0.59	0.54	0.61	0.50	0.45	0.45
label-shuffled	ALBERT	-	0.80	-	-	-	-	0.81	-	-	-
label-shuffled	RoBERTa	-	0.78	-	-	-	-	0.80	-	-	-
fully shuffled	BERT	0.71	0.79	0.71	0.49	0.67	0.72	0.81	0.74	0.48	0.62
fully shuffled	ALBERT	-	0.80	-	-	-	-	0.82	-	-	-
fully shuffled	RoBERTa	-	0.79	-	-	-	-	0.80	-	-	-

Table 3: Pareto hypervolumes for POSL and DAL tasks. Since Nuclear Norm values are unbounded, we limited them to 400. We also normalized Nuclear Norm and Rank results, dividing the volume by their maximum complexity (e.g., the maximum rank or norm).
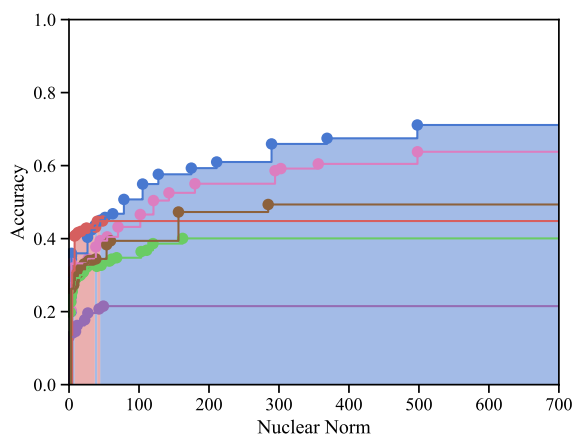

Figure 7: Pareto curves on dependency parsing with the Nuclear Norm complexity metric in English. The x-axis corresponds to the complexity, while the y-axis measures the probes performance on the task. Since the nuclear norm is unbounded, we maxed it to 700 in the parsing task. Probing the representations: ALBERT, BERT, RoBERTa, fastText, one-hot, and random.
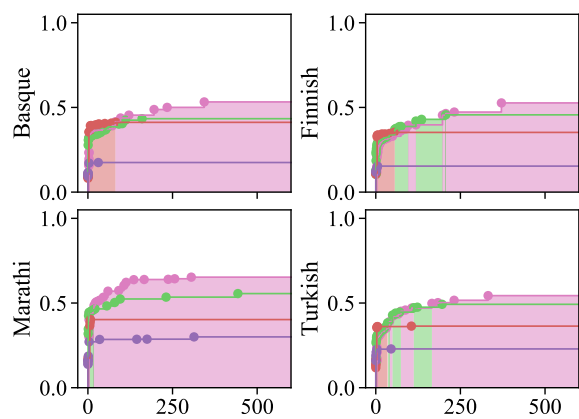

Figure 8: Pareto curves on dependency parsing with the Nuclear Norm complexity metric in a diverse set of languages. The x-axis corresponds to the complexity, while the y-axis measures the probes performance on the task. Since the nuclear norm is unbounded, we maxed it to 700 in the parsing task. Probing the representations: BERT, fastText, one-hot, and random.
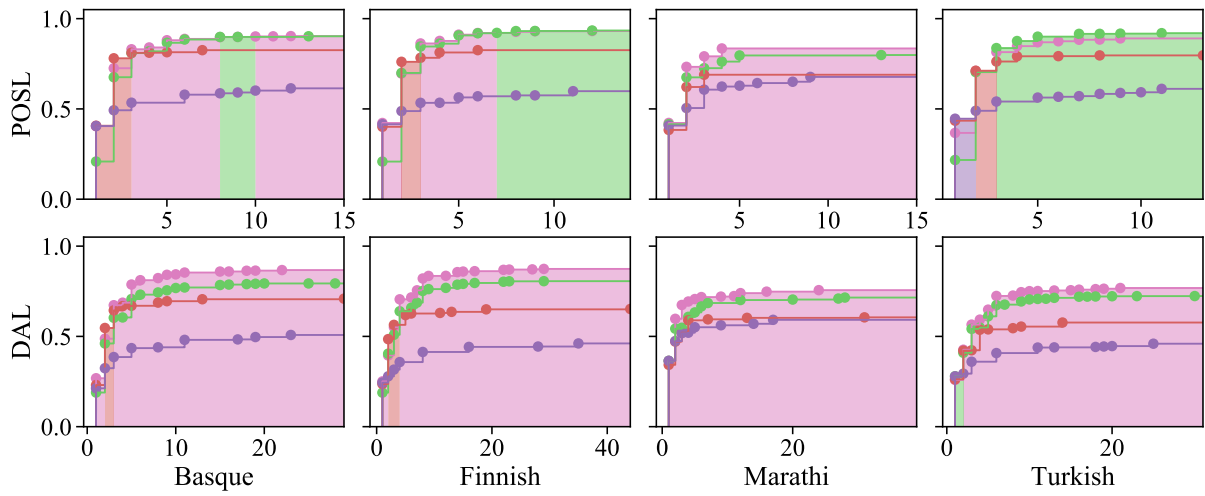
Figure 9: Pareto curves on POSL and DAL with the Rank complexity metric in a diverse set of languages. The x-axis corresponds to the complexity, while the y-axis measures the probes performance on the task. Probing the representations: BERT, fastText, one-hot, and random.
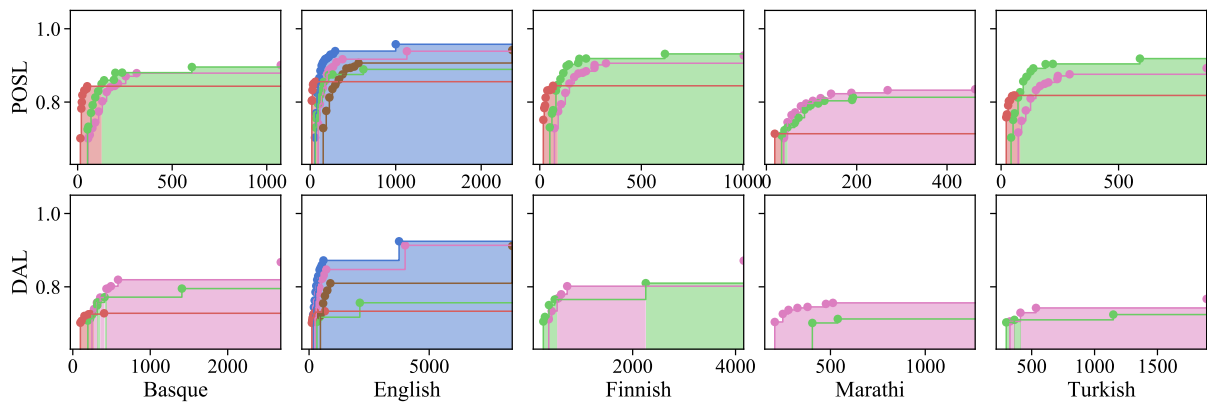

Figure 10: Zoomed in (in the y-axis) Pareto curves on POSL and DAL under the nuclear norm complexity metric. The x-axis corresponds to the complexity, while the y-axis measures the probes performance on the task. In this plot we do not max the nuclear norm, showing its full range. Probing the representations: BERT, fastText, one-hot, and random.