Information-Theoretic Probing for Linguistic Structure

Tiago Pimentel[ð] **Josef Valvoda**[ð] **Rowan Hall Maudslay**[ð] **Ran Zmigrod**[ð]
Adina Williams[ə] **Ryan Cotterell**[ð,ʕ]
[ð]University of Cambridge [ə]Facebook AI Research [ʕ]ETH Zürich
tp472@cam.ac.uk, jv406@cam.ac.uk, rh635@cam.ac.uk,
rz279@cam.ac.uk, adinawilliams@fb.com, rdc42@cl.cam.ac.uk

Abstract

The success of neural networks on a diverse set of NLP tasks has led researchers to question how much do these networks actually know about natural language. Probes are a natural way of assessing this. When probing, a researcher chooses a linguistic task and trains a supervised model to predict annotation in that linguistic task from the network's learned representations. If the probe does well, the researcher may conclude that the representations encode knowledge related to the task. A commonly held belief is that using simpler models as probes is better; the logic is that such models will *identify linguistic structure*, but not *learn the task itself*. We propose an information-theoretic formalization of probing as estimating mutual information that contradicts this received wisdom: one should always select the highest performing probe one can, even if it is more complex, since it will result in a tighter estimate. The empirical portion of our paper focuses on obtaining tight estimates for how much information BERT knows about parts of speech in a set of five typologically diverse languages that are often underrepresented in parsing research, plus English, totaling six languages. We find BERT accounts for only at most 5% more information than traditional, type-based word embeddings.

1 Introduction

Neural networks are the backbone of modern state-of-the-art Natural Language Processing (NLP) systems. One inherent by-product of training a neural network is the production of real-valued representations. Many speculate that these representations encode a continuous analogue of discrete linguistic properties, e.g., part-of-speech tags, due to the networks' impressive performance on many NLP tasks (Belinkov et al., 2017). As a result of this speculation, one common thread of research focuses on the construction of **probes**, i.e., supervised models that are trained to extract the linguistic properties directly (Belinkov et al., 2017; Conneau et al., 2018; Peters et al., 2018b; Zhang and Bowman, 2018; Tenney et al., 2019; Naik et al., 2018). A syntactic probe, then, is a model for extracting syntactic properties, such as part-of-speech, from the representations (Hewitt and Liang, 2019).

In this work, we question what the goal of probing for linguistic properties ought to be. Informally, probing is often described as an attempt to discern how much information representations encode about a specific linguistic property. We make this statement more formal: We assert that the goal of probing ought to be estimating the mutual information (Cover and Thomas, 2012) between a representation-valued random variable and a linguistic property-valued random variable. This formulation gives probing a clean, information-theoretic foundation, and allows us to consider what "probing" actually means.

Our analysis also provides insight into how to choose a probe family: We show that choosing the highest-performing probe, independent of its complexity, is optimal for achieving the best estimate of mutual information (MI). This contradicts the received wisdom that one should always select simple probes over more complex ones (Alain and Bengio, 2017; Liu et al., 2019; Hewitt and Manning, 2019). In this context, we also discuss the recent work of Hewitt and Liang (2019) who propose **selectivity** as a criterion for choosing families of probes. Hewitt and Liang (2019) define selectivity as the performance difference between a probe on the target task and a control task, writing "[t]he selectivity of a probe puts linguistic task accuracy in context with the probe's capacity to memorize from word types." They further ponder: "when a probe achieves high accuracy on a linguistic task using a representation, can we conclude that the represen-

tation encodes linguistic structure, or has the probe just learned the task?" Information-theoretically, there is *no difference* between learning the task and probing for linguistic structure, as we will show; thus, it follows that one should always employ the best possible probe for the task without resorting to artificial constraints.

In support of our discussion, we empirically analyze word-level part-of-speech labeling, a common syntactic probing task (Hewitt and Liang, 2019; Sahin et al., 2019), within our framework. Working on a typologically diverse set of languages (Basque, Czech, English, Finnish, Tamil, and Turkish), we show that the representations from BERT, a common contextualized embedder, only account for at most 5% more of the part-of-speech tag entropy than a control. These modest improvements suggest that most of the information needed to tag part-of-speech well is encoded at the lexical level, and does not require the sentential context of the word. Put more simply, words are not very ambiguous with respect to part of speech, a result known to practitioners of NLP (Garrette et al., 2013). We interpret this to mean that part-of-speech labeling is not a very informative probing task.

We also remark that formulating probing information-theoretically gives us a simple, but stunning result: contextual word embeddings, e.g., BERT (Devlin et al., 2019) and ELMo (Peters et al., 2018a), contain *the same* amount of information about the linguistic property of interest as the original sentence. This follows naturally from the data-processing inequality under a very mild assumption. What this suggests is that, in a certain sense, probing for linguistic properties in representations may not be a well grounded enterprise at all.

2 Word-Level Syntactic Probes for Contextual Embeddings

Following Hewitt and Liang (2019), we consider probes that examine syntactic knowledge in contextualized embeddings. These probes only consider a single token's embedding and try to perform the task using only that information. Specifically, in this work, we consider part-of-speech (POS) labeling: determining a word's part of speech in a given sentence. For example, we wish to determine whether the word *love* is a NOUN or a VERB. This task requires the sentential context for success. As an example, consider the utterance "love is blind" where, only with the context, is it clear that *love* is a NOUN. Thus, to do well on this task, the contextualized embeddings need to encode enough about the surrounding context to correctly guess the POS.

2.1 Notation

Let S be a random variable ranging over all possible sequences of words. For the sake of this paper, we assume the vocabulary $\mathcal{V}$ is finite and, thus, the values S can take are in $\mathcal{V}^*$. We write $\mathbf{s} \in S$ as $\mathbf{s} = w_1 \cdots w_{|\mathbf{s}|}$ for a specific sentence, where each $w_i \in \mathcal{V}$ is a specific word in the sentence and the position $i \in \mathbb{N}^+$. We also define the random variable W that ranges over the vocabulary $\mathcal{V}$. We define both a sentence-level random variable S and a word-level random variable W since each will be useful in different contexts during our exposition.

Next, let T be a random variable whose possible values are the analyses t that we want to consider for word w_i in its sentential context, $\mathbf{s} = w_1 \cdots w_i \cdots w_{|\mathbf{s}|}$. In this work, we will focus on predicting the part-of-speech tag of the i^{th} word w_i. We denote the set of values T can take as the set $\mathcal{T}$. Finally, let R be a representation-valued random variable for the i^{th} word w_i in a sentence derived from the entire sentence $\mathbf{s}$. We write $\mathbf{r} \in \mathbb{R}^d$ for a value of R. While any given value $\mathbf{r}$ is a continuous vector, there are only a countable number of values R can take. To see this, note there are only a countable number of sentences in $\mathcal{V}^*$.

Next, we assume there exists a true distribution $p(t, \mathbf{s}, i)$ over analyses t (elements of $\mathcal{T}$), sentences $\mathbf{s}$ (elements of $\mathcal{V}^*$), and positions i (elements of $\mathbb{N}^+$). Note that the conditional distribution $p(t \mid \mathbf{s}, i)$ gives us the true distribution over analyses t for the i^{th} word in the sentence $\mathbf{s}$. We will augment this distribution such that p is additionally a distribution over $\mathbf{r}$, i.e.,

$$p(\mathbf{r}, t, \mathbf{s}, i) = \delta(\mathbf{r} \mid \mathbf{s}, i)\, p(t, \mathbf{s}, i) \tag{1}$$

where we define the augmentation as a Dirac's delta function

$$\delta(\mathbf{r} \mid \mathbf{s}, i) = \mathbb{1}\{\mathbf{r} = \text{BERT}(\mathbf{s})_i\} \tag{2}$$

Since contextual embeddings are a deterministic function of a sentence $\mathbf{s}$, the augmented distribution in eq. (1) has no more randomness than the original—its entropy is the same. We assume the values of the random variables defined above are distributed according to this (unknown) p. While we do not have access to p, we assume the data in our corpus were drawn according to it. Note that

W—the random variable over possible words—is distributed according to the marginal distribution

$$p(w) = \sum_{\mathbf{s} \in \mathcal{V}^*} \sum_{i=1}^{|\mathbf{s}|} \delta(w \mid \mathbf{s}, i)\, p(\mathbf{s}, i) \tag{3}$$

where we define the deterministic distribution

$$\delta(w \mid \mathbf{s}, i) = \mathbb{1}\{\mathbf{s}_i = w\} \tag{4}$$

2.2 Probing as Mutual Information

The task of supervised probing is an attempt to ascertain how much information a specific representation $\mathbf{r}$ tells us about the value of t. This is naturally expressed as the mutual information, a quantity from information theory:

$$\mathrm{I}(T; R) = \mathrm{H}(T) - \mathrm{H}(T \mid R) \tag{5}$$

where we define the entropy, which is constant with respect to the representations, as

$$\mathrm{H}(T) = -\sum_{t \in \mathcal{T}} p(t) \log p(t) \tag{6}$$

and where we define the conditional entropy as

$$\mathrm{H}(T \mid R) = \int p(\mathbf{r})\, \mathrm{H}\left(T \mid R = \mathbf{r}\right) d\mathbf{r} \tag{7}$$

$$= \sum_{\mathbf{s} \in \mathcal{V}^*} \sum_{i=1}^{|\mathbf{s}|} p(\mathbf{s}, i)\, \mathrm{H}\left(T \mid R = \textsc{bert}(\mathbf{s})_i\right)$$

the point-wise conditional entropy inside the sum is defined as

$$\mathrm{H}(T \mid R = \mathbf{r}) = -\sum_{t \in \mathcal{T}} p(t \mid \mathbf{r}) \log p(t \mid \mathbf{r}) \tag{8}$$

Again, we will not know any of the distributions required to compute these quantities; the distributions in the formulae are marginals and conditionals of the true distribution discussed in eq. (1).

2.3 Bounding Mutual Information

The desired conditional entropy, $\mathrm{H}(T \mid R)$ is not readily available, but with a model $q_{\boldsymbol{\theta}}(\mathbf{t} \mid \mathbf{r})$ in hand, we can upper-bound it by measuring their empirical cross entropy

$$\mathrm{H}(T \mid R) := -\mathop{\mathbb{E}}_{(t,\mathbf{r}) \sim p(\cdot,\cdot)} \left[\log p(t \mid \mathbf{r})\right] \tag{9}$$

$$= -\mathop{\mathbb{E}}_{(t,\mathbf{r}) \sim p(\cdot,\cdot)} \left[\log \frac{p(t \mid \mathbf{r}) q_{\boldsymbol{\theta}}(t \mid \mathbf{r})}{q_{\boldsymbol{\theta}}(t \mid \mathbf{r})}\right]$$

$$= -\mathop{\mathbb{E}}_{(t,\mathbf{r}) \sim p(\cdot,\cdot)} \left[\log q_{\boldsymbol{\theta}}(t \mid \mathbf{r}) + \log \frac{p(t \mid \mathbf{r})}{q_{\boldsymbol{\theta}}(t \mid \mathbf{r})}\right]$$

$$= \underbrace{\mathrm{H}_{q_{\boldsymbol{\theta}}}(T \mid R)}_{\textit{estimate}} - \underbrace{\mathop{\mathbb{E}}_{\mathbf{r} \sim p(\cdot)} \mathrm{KL}(p(\cdot \mid \mathbf{r}) \,||\, q_{\boldsymbol{\theta}}(\cdot \mid \mathbf{r}))}_{\textit{expected estimation error}}$$

where $\mathrm{H}_{q_{\boldsymbol{\theta}}}(T \mid R)$ is the cross-entropy we obtain by using $q_{\boldsymbol{\theta}}$ to get this estimate. Since the KL divergence is always positive, we may lower-bound the desired mutual information

$$\mathrm{I}(T; R) := \mathrm{H}(T) - \mathrm{H}(T \mid R)$$
$$\geq \mathrm{H}(T) - \mathrm{H}_{q_{\boldsymbol{\theta}}}(T \mid R) \tag{10}$$

This bound gets tighter, the more similar (in the sense of the KL divergence) $q_{\boldsymbol{\theta}}(\cdot \mid \mathbf{r})$ is to the true distribution $p(\cdot \mid \mathbf{r})$.

Bigger Probes are Better. If we accept mutual information as a natural measure for how much representations encode a target linguistic task (§2.2), then the best estimate of that mutual information is the one where the probe $q_{\boldsymbol{\theta}}(t \mid \mathbf{r})$ is best at the target task. In other words, we want the best probe $q_{\boldsymbol{\theta}}(t \mid \mathbf{r})$ such that we get the tightest bound to the actual distribution $p(t \mid \mathbf{r})$. This paints the question posed by Hewitt and Liang (2019), who write

> "when a probe achieves high accuracy on a linguistic task using a representation, can we conclude that the representation encodes linguistic structure, or has the probe just learned the task?"

as a false dichotomy.[1] From an information-theoretic view, we will always prefer the probe that does better at the target task, since there is *no difference* between learning a task and the representations encoding the linguistic structure.

3 Control Functions

To place the performance of a probe in perspective, Hewitt and Liang (2019) develop the notion of a control task. Inspired by this, we develop an analogue we term **control functions**, which are functions of the representation-valued random variable R. Similar to Hewitt and Liang (2019)'s control tasks, the goal of a control function $\mathbf{c}(\cdot)$ is to place the mutual information $\mathrm{I}(T; R)$ in the context of a baseline that the control function encodes. Control functions have their root in the data-processing inequality (Cover and Thomas, 2012), which states that, for any function $\mathbf{c}(\cdot)$, we have

$$\mathrm{I}(T; R) \geq \mathrm{I}(T; \mathbf{c}(R)) \tag{11}$$

In other words, information can only be lost by processing data. A common adage associated with this inequality is "garbage in, garbage out."

[1] Assuming that the authors intended 'or' here as strictly non-inclusive.

3.1 Type-Level Control Functions

We will focus on type-level control functions in this paper; these functions have the effect of decontextualizing the embeddings. Such functions allow us to inquire how much the contextual aspect of the contextual embeddings help the probe perform the target task. To show that we may map from contextual embeddings to the identity of the word type, we need the following assumption about the embeddings.

Assumption 1. *Every contextualized embedding is unique, i.e., for any pair of sentences* $\mathbf{s}, \mathbf{s}' \in \mathcal{V}^*$*, we have* $(\mathbf{s} \neq \mathbf{s}') \;||\; (i \neq j) \Rightarrow \textsc{bert}(\mathbf{s})_i \neq \textsc{bert}(\mathbf{s}')_j$ *for all* $i \in \{1, \ldots |\mathbf{s}|\}$ *and* $j \in \{1, \ldots, |\mathbf{s}'|\}$.

We note that Assumption 1 is mild. Contextualized word embeddings map words (in their context) to $\mathbb{R}^d$, which is an uncountably infinite space. However, there are only a countable number of sentences, which implies only a countable number of sequences of real vectors in $\mathbb{R}^d$ that a contextualized embedder may produce. The event that any two embeddings would be the same across two distinct sentences is infinitesimally small.[2] Assumption 1 yields the following corollary.

Corollary 1. *There exists a function* $\texttt{id} : \mathbb{R}^d \to V$ *that maps a contextualized embedding to its word type. The function* `id` *is not a bijection since multiple embeddings will map to the same type.*

Using Corollary 1, we can show that any *non-contextualized* word embedding will contain *no more* information than a contextualized word embedding. More formally, we do this by constructing a look-up function $\mathbf{e} : V \to \mathbb{R}^d$ that maps a word to a word embedding. This embedding may be one-hot, randomly generated ahead of time, or the output of a data-driven embedding method, e.g. fastText (Bojanowski et al., 2017). We can then construct a control function as the composition of the look-up function $\mathbf{e}$ and the id function `id`. Using the data-processing inequality, we can prove that in a word-level prediction task, any non-contextual (type level) word-embedding will contain no more information than a contextualized (token level) one, such as BERT and ELMo. Specifically, we have

$$\begin{aligned} &\mathrm{I}(T; R) \geq \\ &\quad \mathrm{I}(T; \texttt{id}(R)) = \mathrm{I}(T; W) \geq \mathrm{I}(T; \mathbf{e}(W)) \end{aligned} \tag{12}$$

This result[3] is intuitive and, perhaps, trivial—context matters information-theoretically. However, it gives us a principled foundation by which to measure the effectiveness of probes as we will show in §3.2.

3.2 How Much Information Did We Gain?

We will now quantify how much a contextualized word embedding knows about a task with respect to a specific control function $\mathbf{c}(\cdot)$. We term how much more information the contextualized embeddings have about a task than a control variable the **gain**, which we define as

$$\begin{aligned} \mathcal{G}(T, R, \mathbf{c}) &= \mathrm{I}(T; R) - \mathrm{I}(T; \mathbf{c}(R)) \\ &= \mathrm{H}(T \mid \mathbf{c}(R)) - \mathrm{H}(T \mid R) \geq 0 \end{aligned} \tag{13}$$

The gain function will be our method for measuring how much more information contextualized representations have over a controlled baseline, encoded as the function $\mathbf{c}$. We will empirically estimate this value in §6.

Interestingly enough, the gain has a straightforward interpretation.

Proposition 1. *The gain function is equal to the following conditional mutual information*

$$\mathrm{I}(T; R \mid \mathbf{c}(R)) = \mathcal{G}(T, R, \mathbf{c}) \tag{14}$$

Proof.

$$\begin{aligned} \mathrm{I}(T; R \mid \mathbf{c}(R)) &:= \mathrm{I}(T; R) - \mathrm{I}(T; R; \mathbf{c}(R)) \\ &= \mathrm{I}(T; R) - \mathrm{I}(T; \mathbf{c}(R)) \\ &= \mathcal{G}(T, R, \mathbf{c}) \end{aligned}$$

The jump from the first to the second equality follows since R encodes all the information about T provided by $\mathbf{c}(R)$ by construction. □

Proposition 1 gives us a clear understanding of the quantity we wish to estimate: It is how much information about a task is encoded in the representations, given some control knowledge. If properly designed, this control transformation will remove information from the probed representations.

3.3 Approximating the Gain

The gain, as defined in eq. (13), is intractable to compute. In this section we derive a pair of variational bounds on $\mathcal{G}(T, R, \mathbf{e})$—one upper and one

[2] Indeed, even if we sampled every embedding randomly from a d-dimensional Gaussian, the probability that we would ever sample the same real value is zero.

[3] Note that although this result holds in theory, in practice the functions `id` and $\mathbf{e}(\cdot)$ might be arbitrarily hard to estimate. This is discussed in length in §4.3.

lower. To approximate the gain, we will simultaneously minimize an upper and a lower-bound on eq. (13). We begin by approximating the gain in the following manner

$$\mathcal{G}(T, R, \mathbf{e}) \approx \underbrace{\mathrm{H}_{q_{\boldsymbol{\theta}2}}(T; \mathbf{c}(R)) - \mathrm{H}_{q_{\boldsymbol{\theta}1}}(T \mid R)}_{\textit{estimated } \mathcal{G}_{q_{\boldsymbol{\theta}}}(T,R,\mathbf{e})} \tag{15}$$

these cross-entropies can be empirically estimated. We will assume access to a corpus $\{(t_i, \mathbf{r}_i)\}_{i=1}^{N}$ that is human-annotated for the target linguistic property; we further assume that these are samples $(t_i, \mathbf{r}_i) \sim p(\cdot, \cdot)$ from the true distribution. This yields a second approximation that is tractable:

$$\mathrm{H}_{q_{\boldsymbol{\theta}}}(T; R) \approx \frac{1}{N} \sum_{i=1}^{N} \log q_{\boldsymbol{\theta}}(t_i \mid \mathbf{r}_i) \tag{16}$$

This approximation is exact in the limit $N \to \infty$ by the law of large numbers.

We note the approximation given in eq. (15) may be either positive or negative and its estimation error follows from eq. (9)

$$\begin{aligned} \Delta &= \mathop{\mathbb{E}}_{\mathbf{r}\sim p(\cdot)} \mathrm{KL}(p(\cdot \mid \mathbf{r}) \mid\mid q_{\boldsymbol{\theta}1}(\cdot \mid \mathbf{r})) \\ &- \mathop{\mathbb{E}}_{\mathbf{r}\sim p(\cdot)} \mathrm{KL}(p(\cdot \mid \mathbf{c}(\mathbf{r})) \mid\mid q_{\boldsymbol{\theta}2}(\cdot \mid \mathbf{c}(\mathbf{r}))) \\ &= \mathrm{KL}_{q_{\boldsymbol{\theta}1}}(T, R) - \mathrm{KL}_{q_{\boldsymbol{\theta}2}}(T, \mathbf{c}(R)) \end{aligned} \tag{17}$$

where we abuse the KL notation to simplify the equation. This is an undesired behavior since we know the gain itself is non-negative, by the data-processing inequality, but we have yet to devise a remedy.

We justify the approximation in eq. (15) with a pair of variational bounds. The following two corollaries are a result of Theorem 2 in App. A.

Corollary 2. *We have the following upper-bound on the gain*

$$\mathcal{G}(T, R, \mathbf{e}) \leq \mathcal{G}_{q_{\boldsymbol{\theta}}}(T, R, \mathbf{e}) + \mathrm{KL}_{q_{\boldsymbol{\theta}1}}(T, R) \tag{18}$$

Corollary 3. *We have the following lower-bound on the gain*

$$\mathcal{G}(T, R, \mathbf{e}) \geq \mathcal{G}_{q_{\boldsymbol{\theta}}}(T, R, \mathbf{e}) - \mathrm{KL}_{q_{\boldsymbol{\theta}2}}(T, \mathbf{c}(R)) \tag{19}$$

The conjunction of Corollary 2 and Corollary 3 suggest a simple procedure for finding a good approximation: We choose $q_{\boldsymbol{\theta}1}(\cdot \mid r)$ and $q_{\boldsymbol{\theta}2}(\cdot \mid r)$ so as to *minimize* eq. (18) and *maximize* eq. (19), respectively. These distributions contain no overlapping parameters, by construction, so these two optimization routines may be performed independently. We will optimize both with a gradient-based procedure, discussed in §6.

4 Understanding Probing Information-Theoretically

In §3 we developed an information-theoretic framework for thinking about probing contextual word embeddings for linguistic structure. However, we now cast doubt on whether probing makes sense as a scientific endeavour. We prove in §4.1 that contextualized word embeddings, by construction, contain no more information about a word-level syntactic task than the original sentence itself. Nevertheless, we do find a meaningful scientific interpretation of control functions. We expound upon this in §4.2, arguing that control functions are useful, not for understanding representations, but rather for understanding the influence of sentential context on word-level syntactic tasks, e.g., labeling words with their part of speech.

4.1 You Know Nothing, BERT

To start, we note the following corollary

Corollary 4. *It directly follows from Assumption 1 that* BERT *is a bijection between sentences* $\mathbf{s}$ *and sequences of embeddings* $\langle \mathbf{r}_1, \ldots, \mathbf{r}_{|\mathbf{s}|} \rangle$. *As* BERT *is a bijection, it has an inverse, which we will denote as* BERT^{-1}.

Theorem 1. *The function* $\text{BERT}(S)$ *cannot provide more information about* T *than the sentence* S *itself.*

Proof.

$$\begin{aligned} \mathrm{I}(T; S) &\geq \mathrm{I}(T; \text{BERT}(S)) \\ &\geq \mathrm{I}(T; \text{BERT}^{-1}(\text{BERT}(S))) \\ &= \mathrm{I}(T; S) \end{aligned} \tag{20}$$

This implies $\mathrm{I}(T; S) = \mathrm{I}(T; \text{BERT}(S))$. We remark this is not a BERT-specific result—it rests on the fact that the data-processing inequality is tight for bijections. □

While Theorem 1 is a straightforward application of the data-processing inequality, it has deeper ramifications for probing. It means that if we search for syntax in the contextualized word embeddings of a sentence, we should not expect to find any

more syntax than is present in the original sentence. In a sense, Theorem 1 is a cynical statement: the endeavour of finding syntax in contextualized embeddings sentences is nonsensical. This is because, under Assumption 1, we know the answer *a priori*—the contextualized word embeddings of a sentence contain exactly the same amount of information about syntax as does the sentence itself.

4.2 What Do Control Functions Mean?

Information-theoretically, the interpretation of control functions is also interesting. As previously noted, our interpretation of control functions in this work does not provide information about the representations themselves. Actually, the same reasoning used in Corollary 1 could be used to devise a function $\texttt{id}_s(\mathbf{r})$ which led from a single representation back to the whole sentence. For a type-level control function $\mathbf{c}$, by the data-processing inequality, we have that $\mathrm{I}(T; W) \geq \mathrm{I}(T; \mathbf{c}(R))$. Consequently, we can get an upper-bound on how much information we can get out of a decontextualized representation. If we assume we have perfect probes, then we get that the true gain function is $\mathrm{I}(T; S) - \mathrm{I}(T; W) = \mathrm{I}(T; S \mid W)$. This quantity is interpreted as the amount of knowledge we gain about the word-level task T by knowing S (i.e., the sentence) in addition to W (i.e., the word). Therefore, a perfect probe would provide insights about language and not about the actual representations, which are no more than a means to an end.

4.3 Discussion: Ease of Extraction

We do acknowledge another interpretation of the work of Hewitt and Liang (2019) *inter alia*; BERT makes the syntactic information present in an ordered sequence of words more easily extractable. However, ease of extraction is not a trivial notion to formalize, and indeed, we know of no attempt to do so; it is certainly more complex to determine than the number of layers in a multi-layer perceptron (MLP). Indeed, a MLP with a single hidden layer can represent any function over the unit cube, with the caveat that we may need a very large number of hidden units (Cybenko, 1989).

Although for perfect probes the above results should hold, in practice $\texttt{id}(\cdot)$ and $\mathbf{c}(\cdot)$ may be hard to approximate. Furthermore, if these functions were to be learned, they might require an unreasonably large dataset. A random embedding control function, for example, would require an infinitely large dataset to be learned—or at least one that contained all words in the vocabulary V. "Better" representations should make their respective probes more easily learnable—and consequently their encoded information more accessible.

We suggest that future work on probing should focus on operationalizing ease of extraction more rigorously—even though we do not attempt this ourselves. The advantage of simple probes is that they may reveal something about the *structure* of the encoded information—i.e., is it structured in such a way that it can be easily taken advantage of by downstream consumers of the contextualized embeddings? We suspect that many researchers who are interested in less complex probes have implicitly had this in mind.

5 A Critique of Control Tasks

While this paper builds on the work of Hewitt and Liang (2019), and we agree with them that we should have control tasks when probing for linguistic properties, we disagree with parts of the methodology for the control task construction. We present these disagreements here.

5.1 Structure and Randomness

Hewitt and Liang (2019) introduce control tasks to evaluate the effectiveness of probes. We draw inspiration from this technique as evidenced by our introduction of control functions. However, we take issue with the suggestion that controls should have **structure** and **randomness**, to use the terminology from Hewitt and Liang (2019). They define structure as "the output for a word token is a deterministic function of the word type." This means that they are stripping the language of ambiguity with respect to the target task. In the case of part-of-speech labeling, *love* would either be a NOUN or a VERB in a control task, never both: this is a problem. The second feature of control tasks is randomness, i.e., "the output for each word type is sampled independently at random.[4]" In conjunction, structure and randomness may yield a relatively trivial task that does not look at all like natural language.

What is more, there is a closed-form solution for an optimal, retrieval-based "probe" that has zero parameters: [5] If a word type appears in the training set, return the label with which it was annotated

[4] But not necessarily uniformly.

[5] Actually, to be more precise, it will have $|V| + 1$ parameters. One for each word in the vocabulary, plus one for the most frequent label.

there, otherwise return the most frequently occurring label across all words in the training set. This probe will achieve an accuracy that is 1 minus the out-of-vocabulary rate (the number of tokens in the test set that correspond to novel types divided by the number of tokens) times the percentage of tags in the test set that do not correspond to the most frequent tag (the error rate of the guess-the-most-frequent-tag classifier). In short, the best model for a control task is a pure memorizer that guesses the most frequent tag for out-of-vocabulary words.

5.2 What's Wrong with Memorization?

Hewitt and Liang (2019) propose that probes should be optimised to maximise accuracy *and* selectivity. Recall selectivity is given by the distance between the accuracy on the original task and the accuracy on the control task using the same architecture. Given their characterization of control tasks, maximising selectivity leads to a selection of a model that is bad at memorization. But why should we punish memorization? Much of linguistic competence is about generalization, however memorization also plays a key role (Fodor et al., 1974; Nooteboom et al., 2002; Fromkin et al., 2018), with word learning (Carey, 1978) being an obvious example. Indeed, maximizing selectivity as a criterion for creating probes seems to artificially disfavor this property.

5.3 What Low-Selectivity Means

Hewitt and Liang (2019) acknowledge that for the more complex task of dependency edge prediction, a MLP probe is more accurate and, therefore, preferable despite its low selectivity. However, they offer two counter-examples where the less selective neural probe exhibits drawbacks when compared to its more selective, linear counterpart. We believe both examples are a symptom of using a simple probe rather than of selectivity being a useful metric for probe selection. First, Hewitt and Liang (2019, §3.6) point out that, in their experiments, the MLP-1 model frequently mislabels the word with suffix *-s* as NNPS on the POS labeling task. They present this finding as a possible example of a less selective probe being less faithful in representing what linguistic information has the model learned. Our analysis leads us to believe that, on contrary, this shows that one should be using the best possible probe to minimize the chance of misrepresentation. Since more complex probes achieve higher accuracy on the task, as evidence by the findings of Hewitt and Liang (2019), we believe that the overall trend of misrepresentation is higher for the probes with higher selectivity. The same applies for the second example discussed in section Hewitt and Liang (2019, §4.2) where a less selective probe appears to be less faithful. The authors show that the representations on ELMo's second layer fail to outperform its word type ones (layer zero) on the POS labeling task when using the MLP-1 probe. While they argue this is evidence for selectivity being a useful metric in choosing appropriate probes, we argue that this demonstrates yet again that one needs to use a more complex probe to minimize the chances of misrepresenting what the model has learned. The fact that the linear probe shows a difference only demonstrates that the information is perhaps more accessible with ELMo, not that it is not present; see §4.3.

6 Experiments

We consider the task of POS labeling and use the universal POS tag information (Petrov et al., 2012) from the Universal Dependencies 2.4 (Nivre et al., 2019). We probe the multilingual release of BERT[6] on six typologically diverse languages: Basque, Czech, English, Finnish, Tamil, and Turkish; and we compute the contextual representations of each sentence by feeding it into BERT and averaging the output word piece representations for each word, as tokenized in the treebank.

6.1 Control Functions

We will consider three different control functions. Each is defined as the composition $\mathbf{c} = \mathbf{e} \circ \texttt{id}$ with a different look-up function. These look-up functions are

- $\mathbf{e}_{fastText}$ returns a language specific fastText embedding (Bojanowski et al., 2017);
- $\mathbf{e}_{onehot}$ returns a one-hot embedding;[7]
- $\mathbf{e}_{random}$ returns a fixed random embedding.[8]

All of these functions are type level in that they remove the influence of the context on the word.

[6] We used the implementation made available by Wolf et al. (2019)

[7] We initialize random embeddings at the type level, and let them train during the model's optimization.

[8] We generate the random embeddings once before the task, at the type level. Results for this control are in the Appendix.

	# Tokens				bert	fastText		onehot	
Language	Train	Test	# POS	$\mathrm{H}(T)$	$\mathrm{H}(T \mid R)$	$\mathrm{H}(T \mid \mathbf{c}(R))$	$\mathcal{G}(T, R, \mathbf{c})$	$\mathrm{H}(T \mid \mathbf{c}(R))$	$\mathcal{G}(T, R, \mathbf{c})$
Basque	71,483	23,959	16	3.18	0.31	0.30	-0.01 (0.3%)	0.82	0.51 (16.0%)
Czech	1,164,956	172,420	18	3.33	0.08	0.14	0.06 (1.8%)	0.36	0.28 (08.4%)
English	177,583	22,044	17	3.62	0.21	0.39	0.18 (5.0%)	0.64	0.43 (11.9%)
Finnish	138,695	18,263	16	3.16	0.24	0.20	-0.04 (1.3%)	0.86	0.62 (19.6%)
Tamil	5,460	1,656	14	3.21	0.58	0.69	0.11 (3.4%)	1.65	1.05 (32.7%)
Turkish	36,562	9,567	15	3.02	0.33	0.27	-0.09 (3.0%)	0.86	0.50 (16.6%)

Table 1: Amount of information shared by BERT, fastText or onehot embeddings and a POS tagging task. When put into context, multilingual BERT does not tell us much more about syntax than trivial baselines. $\mathrm{H}(T)$ is estimated with a plug-in estimator from same treebanks we use to train the POS labelers.

6.2 Probe Architecture

As expounded upon above, our purpose is to achieve the best bound on mutual information we can. To this end, we employ a deep MLP as our probe. We define the probe as

$$q_{\boldsymbol{\theta}}(t \mid \mathbf{r}) = \text{softmax}\left(W^{(m)}\sigma\left(W^{(m-1)}\cdots\sigma(W^{(1)}\,\mathbf{r})\right)\right) \tag{21}$$

an m-layer neural network with the non-linearity $\sigma(\cdot) = \mathrm{ReLU}(\cdot)$. The initial projection matrix is $W^{(1)} \in \mathbb{R}^{r_1 \times d}$ and the final projection matrix is $W^{(m)} \in \mathbb{R}^{|\mathcal{T}| \times r_{m-1}}$, where $r_i = \frac{r}{2^{i-1}}$. The remaining matrices are $W^{(i)} \in \mathbb{R}^{r_i \times r_{i-1}}$, so we half the number of hidden states in each layer. We optimize over the hyperparameters—number of layers, hidden size, one-hot embedding size, and dropout—by using random search. For each estimate, we train 50 models and choose the one with the best validation cross-entropy. The cross-entropy in the test set is then used as our entropy estimate.

6.3 Results

We know BERT can generate text in many languages, here we assess how much does it actually know about syntax in those languages. And how much more does it know than simple type-level baselines. Tab. 1 presents this results, showing how much information BERT, fastText and onehot embeddings encode about POS tagging. We see that—in all analysed languages—type level embeddings can already capture most of the uncertainty in POS tagging. We also see that BERT only shares a small amount of extra information with the task, having small (or even negative) gains in all languages.

BERT presents negative gains in some of the analysed languages. Although this may seem to contradict the information processing inequality, it is actually caused by the difficulty of approximating `id` and $\mathbf{c}(\cdot)$ with a finite training set—causing $\mathrm{KL}_{q_{\boldsymbol{\theta}1}}(T \mid R)$ to be larger than $\mathrm{KL}_{q_{\boldsymbol{\theta}2}}(T \mid \mathbf{c}(R))$. We believe this highlights the need to formalize *ease of extraction*, as discussed in §4.3.

Finally, when put into perspective, multilingual BERT's representations do not seem to encode much more information about syntax than a trivial baseline. BERT only improves upon fastText in three of the six analysed languages—and even in those, it encodes at most (in English) 5% additional information.

7 Conclusion

We proposed an information-theoretic formulation of probing: we define probing as the task of estimating conditional mutual information. We introduce control functions, which allows us to put the amount of information encoded in contextual representations in the context of knowledge judged to be trivial. We further explored this formalization and showed that, given perfect probes, probing can only yield insights into the language itself and tells us nothing about the representations under investigation. Keeping this in mind, we suggested a change of focus—instead of focusing on probe size or information, we should look at *ease of extraction* going forward.

On another note, we apply our formalization to evaluate multilingual BERT's syntax knowledge on a set of six typologically diverse languages. Although it does encode a large amount of information about syntax (more than 81% in all languages[9]), it only encodes at most 5% more information than some trivial baseline knowledge (a type-level representation). This indicates that the task of POS labeling (word-level POS tagging) is not an ideal task for contemplating the syntactic understanding of contextual word embeddings.

[9]This is measured as the relative difference between $\mathrm{H}(T)$ and $\mathrm{H}(T \mid R)$. On average, this value is 91%.

References

Guillaume Alain and Yoshua Bengio. 2017. Understanding intermediate layers using linear classifier probes.

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.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *TACL*, 5:135–146.

Susan Carey. 1978. The child as word learner. In *Linguistic theory and psychological reality*. Cambridge, MA: The MIT Press.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics.

Thomas M. Cover and Joy A. Thomas. 2012. *Elements of Information Theory*. John Wiley & Sons.

George Cybenko. 1989. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314.

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.

Jerry A. Fodor, Thomas G. Bever, and Merrill F. Garrett. 1974. *The Psychology of Language: An Introduction to Psycholinguistics and Generative Grammar*. New York: McGraw-Hill.

Victoria Fromkin, Robert Rodman, and Nina Hyams. 2018. *An introduction to language*. Cengage Learning.

Dan Garrette, Jason Mielens, and Jason Baldridge. 2013. Real-world semi-supervised learning of POS-taggers for low-resource languages. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 583–592, Sofia, Bulgaria. Association for Computational Linguistics.

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.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019. Linguistic knowledge and transferability of contextual 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 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Joakim Nivre, Mitchell Abrams, 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, 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, Gülşen Cebiroğlu Eryiğit, Flavio Massimiliano Cecchini, Giuseppe G. A. Celano, Slavomír Čéplö, Savas Cetin, Fabricio Chalub, Jinho Choi, Yongseok Cho, Jayeol Chun, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Marine Courtin, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, 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, Tomaž Erjavec, Aline Etienne, 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, Matias Grioni, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Nizar Habash, Jan Hajič, Jan Hajič jr., 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, Hüner Kaşıkara, Andre Kaasen, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Boris Katz, Tolga Kayadelen, Jessica Kenney, Václava Kettnerová, Jesse Kirchner, 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, Kyung-Tae Lim, 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, Yusuke Miyao, Simonetta Montemagni, Amir More, Laura Moreno Romero, Keiko Sophie Mori, Tomohiko Morioka, Shinsuke Mori, Shigeki Moro, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Juan Ignacio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, LÆřÆąng Nguyễn Thị, Huyền Nguyễn Thị Minh, Yoshihiro Nikaido, Vitaly Nikolaev, Rattima Nitisaroj, Hanna Nurmi, Stina Ojala, 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, Jussi Piitulainen, Tommi A. Pirinen, Emily Pitler, Barbara Plank, Thierry Poibeau, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Tiina Puolakainen, Sampo Pyysalo, Andriela Rääbis, Alexandre Rademaker, Loganathan Ramasamy, Taraka Rama, Carlos Ramisch, Vinit Ravishankar, Livy Real, Siva Reddy, Georg Rehm, Michael Rießler, Erika Rimkutė, Larissa Rinaldi, Laura Rituma, Luisa Rocha, Mykhailo Romanenko, Rudolf Rosa, Davide Rovati, Valentin RoÈŹca, 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, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Aaron Smith, Isabela 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, 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, Daniel Zeman, Manying Zhang, and Hanzhi Zhu. 2019. Universal dependencies 2.4. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Sieb G. Nooteboom, Fred Weerman, and F. N. K. Wijnen. 2002. *Storage and computation in the language faculty*. Springer Science & Business Media.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018a. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018b. Dissecting contextual word embeddings: Architecture and representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509, Brussels, Belgium. Association for Computational Linguistics.

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'12)*, pages 2089–2096, Istanbul, Turkey. European Language Resources Association (ELRA).

Gözde Gül Sahin, Clara Vania, Ilia Kuznetsov, and Iryna Gurevych. 2019. LINSPECTOR: multilingual probing tasks for word representations. *CoRR*, abs/1903.09442.

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. 2019. What do you learn

from context? probing for sentence structure in contextualized word representations.

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.

Kelly Zhang and Samuel Bowman. 2018. Language modeling teaches you more than translation does: Lessons learned through auxiliary syntactic task analysis. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 359–361, Brussels, Belgium. Association for Computational Linguistics.

A Variational Bounds

Theorem 2. *The estimation error between $\mathcal{G}_{q_{\boldsymbol{\theta}}}(T, R, \mathbf{e})$ and the true gain can be upper- and lower-bounded by two distinct Kullback–Leibler divergences.*

Proof. We first find the error given by our estimate

$$\begin{aligned}
\mathcal{G}(T, R, \mathbf{e}) &:= \mathrm{H}(T; \mathbf{c}(R)) - \mathrm{H}(T \mid R) \\
&= \mathrm{H}_{q_{\boldsymbol{\theta}2}}(T \mid \mathbf{c}(R)) - \mathop{\mathbb{E}}_{\mathbf{r}\sim p(\cdot)} \mathrm{KL}(p(\cdot \mid \mathbf{c}(\mathbf{r})) \mid\mid q_{\boldsymbol{\theta}2}(\cdot \mid \mathbf{c}(\mathbf{r}))) \\
&\quad - \mathrm{H}_{q_{\boldsymbol{\theta}1}}(T \mid R) + \mathop{\mathbb{E}}_{\mathbf{r}\sim p(\cdot)} \mathrm{KL}(p(\cdot \mid \mathbf{r}) \mid\mid q_{\boldsymbol{\theta}1}(\cdot \mid \mathbf{r})) \\
&= \mathrm{H}_{q_{\boldsymbol{\theta}2}}(T \mid \mathbf{c}(R)) - \mathrm{KL}_{q_{\boldsymbol{\theta}2}}(T, \mathbf{c}(R)) - \mathrm{H}_{q_{\boldsymbol{\theta}1}}(T \mid R) + \mathrm{KL}_{q_{\boldsymbol{\theta}1}}(T \mid R) \\
&= \mathrm{H}_{q_{\boldsymbol{\theta}2}}(T \mid \mathbf{c}(R)) - \mathrm{H}_{q_{\boldsymbol{\theta}1}}(T \mid R) + \mathrm{KL}_{q_{\boldsymbol{\theta}1}}(T \mid R) - \mathrm{KL}_{q_{\boldsymbol{\theta}2}}(T, \mathbf{c}(R)) \\
&= \underbrace{\mathcal{G}_{q_{\boldsymbol{\theta}}}(T, R, \mathbf{e})}_{\textit{estimated gain}} + \underbrace{\mathrm{KL}_{q_{\boldsymbol{\theta}1}}(T \mid R) - \mathrm{KL}_{q_{\boldsymbol{\theta}2}}(T, \mathbf{c}(R))}_{\textit{estimation error}}
\end{aligned} \tag{22}$$

Making use of this error, we trivially find an upper-bound on the estimation error as

$$\begin{aligned}
\Delta &= \mathrm{KL}_{q_{\boldsymbol{\theta}1}}(T \mid R) - \mathrm{KL}_{q_{\boldsymbol{\theta}2}}(T, \mathbf{c}(R)) \\
&\leq \mathrm{KL}_{q_{\boldsymbol{\theta}1}}(T \mid R)
\end{aligned} \tag{23}$$

which follows since KL divergences are never negative. Analogously, we find a lower-bound as

$$\begin{aligned}
\Delta &= \mathrm{KL}_{q_{\boldsymbol{\theta}1}}(T \mid R) - \mathrm{KL}_{q_{\boldsymbol{\theta}2}}(T, \mathbf{c}(R)) \\
&\geq -\mathrm{KL}_{q_{\boldsymbol{\theta}2}}(T, \mathbf{c}(R))
\end{aligned} \tag{24}$$

□

B Further Results

In this section, we present accuracies for the models trained using BERT, fastText and onehot embeddings, and the full results on random embeddings. Tab. 2 shows that both BERT and fastText present high accuracies in all languages, except Tamil. Onehot and random results are considerably worse, as expected, since they could not do more than take random guesses (e.g. guessing the most frequent label in the training test) in any word which was not seen during training.

	accuracies				**random**	
Language	BERT	fastText	onehot	random	$\mathrm{H}(T \mid \mathbf{c}(R))$	$\mathcal{G}(T, R, \mathbf{c})$
Basque	0.93	0.93	0.81	0.82	0.80	0.49 (15.4%)
Czech	0.98	0.97	0.91	0.87	0.54	0.46 (13.8%)
English	0.96	0.91	0.84	0.84	0.68	0.47 (13.0%)
Finnish	0.95	0.96	0.80	0.80	0.89	0.65 (20.6%)
Tamil	0.87	0.84	0.66	0.66	1.52	0.94 (29.3%)
Turkish	0.93	0.94	0.79	0.80	0.83	0.50 (16.6%)

Table 2: Accuracies of the models trained on BERT, fastText, onehot and random embeddings for the POS tagging task.