The Rediscovery Hypothesis: Language Models Need to Meet Linguistics

by

TEZEKBAYEV MAXAT

Submitted to the Department of Mathematics
in partial fulfillment of the requirements for the degree of

Master of Applied Mathematics

at the

NAZARBAYEV UNIVERSITY

May 2022

Author .....
Department of Mathematics
May 8, 2022

Certified by .....
ZHENSIBEK ASSYLBEKOV
Assistant Professor
Thesis Supervisor

Accepted by .....
Gonzalo Hortelano
Acting Dean, School of Science and Humanities

The Rediscovery Hypothesis:

Language Models Need to Meet Linguistics

by

TEZEKBAYEV MAXAT

Submitted to the Department of Mathematics
on May 8, 2022, in partial fulfillment of the
requirements for the degree of
Master of Applied Mathematics

Abstract

There is an ongoing debate in the NLP community whether modern language models contain linguistic knowledge, recovered through so-called *probes*. This work examines whether linguistic knowledge is a necessary condition for the good performance of modern language models, which we call the *rediscovery hypothesis*.

In the first place, we show that language models that are significantly compressed but perform well on their pretraining objectives retain good scores when probed for linguistic structures. This result supports the rediscovery hypothesis and leads to an information-theoretic framework that relates language modeling objectives with linguistic information. This framework also provides a metric to measure the impact of linguistic information on the word prediction task. We reinforce our analytical results with various experiments, both on synthetic and on real NLP tasks in English.

Thesis Supervisor: ZHENSIBEK ASSYLBEKOV
Title: Assistant Professor

Contents

List of Figures

Chapter 1

Overview

1.1 Introduction

Vector representations of words obtained from self-supervised pretraining of neural language models (LMs) on massive unlabeled data have revolutionized NLP in the last decade. This success has spurred an interest in trying to understand what type of knowledge these models actually learn [58, 48].

Of particular interest here is "linguistic knowledge", which is generally measured by annotating test sets through experts following certain pre-defined linguistic schemas. These annotation schemas are based on language regularities manually defined by linguists. On the other side, we have language models which are pretrained predictors that assign a probability to the presence of a token given the surrounding context. Neural models solve this task by finding patterns in the text. We refer to the claim that such neural language models rediscover linguistic knowledge as the *rediscovery hypothesis*. It stipulates that the patterns of the language discovered by the model trying to solve the pretraining task correlate with the human-defined linguistic regularity. In this work we measure the amount of linguistics rediscovered by a pretrained model through the so-called probing tasks: hidden layers of a neural LM are fed to

a simple classifier—a probe—that learns to predict a linguistic structure of interest [16, 1, 10, 26, 65]. In Section 2 we attempt to challenge the *rediscovery hypothesis* through a variety of experiments to understand to what extent it holds.

Those experiments aim to verify whether the path through language regularity is indeed the one taken by pretrained LMs or whether there is another way to reach good LM performance without rediscovering linguistic structure. The experiments show that pretraining loss is indeed tightly linked to the amount of linguistic structure discovered by an LM. We, therefore, fail to reject the rediscovery hypothesis.

This negative attempt, as well as the abundance of positive examples in the literature, motivates us to prove mathematically the rediscovery hypothesis. In Section 3.1 we use information theory to prove the contrapositive of the hypothesis—removal of linguistic information from an LM degrades its performance. Moreover, we show that the decline in the LM quality depends on how strongly the removed property is interdependent with the underlying text: a greater dependence leads to a greater drop. We confirm this result empirically, both with synthetic data and with real annotations on English text.

The result that removing information that contains strong mutual information with the underlying text degrades (masked) word prediction might not seem surprising *a posteriori*. However, it is this surprise that lies at the heart of most of the work in recent years around the discovery of how easily this information can be extracted from intermediate representations. Our framework also provides a coefficient that measures the dependence between a probing task and the underlying text. This measure can be used to determine more complex probing tasks, whose rediscovery by language models would indeed be surprising.

1.2 Word Embeddings and Language Models

We explore one static embedding model, SGNS, and contextualized embedding model, CoVe.

Word2vec SGNS [41] is a shallow two-layer neural network that produces uncontextualized word embeddings. It is widely accepted that the SGNS vectors capture words semantics to a certain extent, which is confirmed by the folklore examples, such as $\mathbf{w}_{\text{king}} - \mathbf{w}_{\text{man}} + \mathbf{w}_{\text{woman}} \approx \mathbf{w}_{\text{queen}}$. SGNS [40] is a masked language model with all tokens but one in a sequence being masked. SGNS approximates its cross-entropy loss by negative sampling procedure [7].So, the question of the relationship between the SGNS objective function and its ability to discover linguistics is also relevant.

CoVe [38] uses the top-level activations of a two-layer BiLSTM encoder from an attentional sequence-to-sequence model [6] trained for English-to-German translation. CoVe is a conditional language model conditioned on the sequence in source language. The authors used the CommonCrawl-840B GloVe model [46] for English word vectors, which were completely fixed during pretraining, and we follow their setup. This entails that the embedding layer on the source side is not pruned during the LTH procedure. We also concatenate the encoder output with the GloVe embeddings as is done in the original paper [38, Eq. 6].

1.3 Pruning Method

The lottery ticket hypothesis (LTH) of [19] claims that a randomly-initialized neural network $f(\mathbf{x}; \boldsymbol{\theta})$ with trainable parameters $\boldsymbol{\theta} \in \mathbb{R}^n$ contains subnetworks $f(\mathbf{x}; \mathbf{m} \odot \boldsymbol{\theta})$, $\mathbf{m} \in \{0, 1\}^n$, such that, when trained in isolation, they can match the performance of the original network, where $\odot$ is the element-wise multiplication. The authors

suggest a simple procedure for identifying such subnetworks (Alg. 1.3.1). When this procedure is applied iteratively (Step 5 of Alg. 1.3.1), we get a sequence of pruned models $\{f(\mathbf{x}; \mathbf{m}_i \odot \boldsymbol{\theta}_0)\}$ in which each model has fewer parameters than its predecessor: $\|\mathbf{m}_i\|_0 < \|\mathbf{m}_{i-1}\|_0$, where $\|\mathbf{x}\|_0$ is a number of nonzero elements in $\mathbf{x} \in \mathbb{R}^d$. [19] used iterative pruning for image classification models and found subnetworks that were 10%–20% of the sizes of the original networks and met or exceeded their validation accuracies. Such a compression approach retains weights important for the main task while discarding others. We hypothesize that it might be those additional weights that contain the signals used by probes. But, if the subnetworks retain linguistic knowledge then this is evidence in favor of the rediscovery hypothesis.

Algorithm 1.3.1: Lottery ticket hypothesis—Identifying winning tickets [19]

1 Randomly initialize a neural network $f(\mathbf{x}; \boldsymbol{\theta}_0)$, $\boldsymbol{\theta}_0 \in \mathbb{R}^n$
2 Train the network for j iterations, arriving at parameters $\boldsymbol{\theta}_j$.
3 Prune p% of the parameters in $\boldsymbol{\theta}_j$, creating a mask $\boldsymbol{m} \in \{0, 1\}^n$.
4 Reset the remaining parameters to their values in $\boldsymbol{\theta}_0$, creating the winning ticket $f(\mathbf{x}; \boldsymbol{m} \odot \boldsymbol{\theta}_0)$.
5 Repeat from 2 if performing iterative pruning.
6 Train the winning ticket $f(\mathbf{x}; \mathbf{m} \odot \boldsymbol{\theta}_0)$ to convergence.

1.4 Measuring the Amount of Linguistic Knowledge

To properly measure the *amount of linguistic knowledge* in word vectors we define it as the performance of classifiers (probes) that take those vectors as input and are trained on linguistically annotated data. This definition has the advantage of being able to be measured exactly, at the cost of avoiding the discussion of whether POS tags or syntactic parse trees indeed denote linguistic knowledge captured by humans in their learning process.

We should note that this probing approach has received a lot of criticism recently [25, 51, 69] due to its inability to distinguish between information encoded in the pretrained vectors from the information learned by the probing classifier. However, in our study, the question is not *How much linguistics is encoded in the presentation vector?*, but rather *Does one vector contain more linguistic information than the other?* We compare different representations of the *same* dimensionality using probing classifiers of the *same* capacity. Even if part of the probing performance is due to the classifier itself we claim that the difference in the probing performance will be due to the difference in the amount of linguistic knowledge encoded in the representations we manipulate. This conjecture is strengthened by the findings of [74] who analyzed the representations from pretrained miniBERTAs and demonstrated that the trends found through edge probing [66] are the same as those found through better-designed probes such as Minimum Description Length [69]. Therefore in this work we adopt *edge probing* and *structural probing* for contextualized embeddings. For static embeddings, we use the traditional *word similarity* and *word analogy* tasks.

Edge probing [66] formulates several linguistics tasks of different nature as text span classification tasks. The probing model is a lightweight classifier on top of the pretrained representations trained to solve those linguistic tasks. In this work we use

the part-of-speech tagging (POS), constituent labeling, named entity labeling (NE), and semantic role labeling (SRL) tasks from the suite, in which a probing classifier receives a sequence of tokens and predicts a label for it. For example, in the case of constituent labeling, for a sentence *This probe* [*discovers linguistic knowledge*], the sequence in square brackets should be labeled as a verb phrase.

Structural probing [26] evaluates whether syntax trees are embedded in a linear transformation of a neural network's word representation space. The probe identifies a linear transformation under which squared Euclidean distance encodes the distance between words in the parse tree. [26] show that such transformations exist for both ELMo and BERT but not in static baselines, providing evidence that entire syntax trees can be easily extracted from the vector geometry of deep models.

Word similarity [17] and word analogy [42] tasks can be considered as non-parametric probes of static embeddings, and—differently from the other probing tasks—are not learned. The use of word embeddings in the word similarity task has been criticized for the instability of the results obtained [3]. Regarding the word analogy task, [61] raised concerns on the misalignment of assumptions in generating and testing word embeddings. However, the success of the static embeddings in performing well in these tasks was a crucial part of their widespread adoption.

Chapter 2

Main

The first question we pose in this work is the following: is the rediscovery of linguistic knowledge mandatory for models that perform well on their pretraining tasks, typically language modeling or translation; or is it a side effect of overparameterization?[1]

We analyze the correlation between linguistic knowledge and LM performance with pruned pretrained models. By compressing a network through pruning we retain the same overall architecture and can compare probing methods. More important, we hypothesize that pruning removes all unnecessary information with respect to the pruning objective (language modeling) and that it might be that information that is used to rediscover linguistic knowledge.

2.1 Experimental Setup

We prune SGNS and CoVe the embedding models with the LTH algorithm (Alg. 1.3.1) and evaluate them with probes from Section 1.4 at *each* pruning iteration. The models are pruned iteratively. Assuming that ℓ_i^ω is the validation loss of the embedding

[1]Overparamterization is defined informally as "having more parameters than can be estimated from the data", and therefore using a model richer than necessary for the task at hand. Those additional parameters could be responsible for the good performance of the probes.

model $\omega \in \{\text{SGNS, CoVe}\}$ at iteration i, and $\Delta s_i^{\omega,T} := s_i^{\omega,T} - s_0^{\omega,T}$ is the drop in the corresponding score on the probing task $T \in \{\text{NE, POS, Const., Struct., Sim., Analogy}\}$ compared to the score $s_0^{\omega,T}$ of the baseline (unpruned) model, we obtain pairs $(\ell_i^\omega, \Delta s_i^{\omega,T})$ for further analysis. SGNS is pruned in full, while in CoVe we prune everything except the source-side embedding layer. This exception is due to the design of the CoVe model, and we follow the original paper's setup [38].

Software and datasets. The SGNS model is pretrained on the `text8` data [35] using our custom implementation [5]. CoVe is trained on the English–German part of the IWSLT 2016 machine translation task [8] using the OpenNMT-py toolkit [28]. The training set consists of 210K sentence pairs from transcribed TED presentations that cover a wide variety of topics.

The edge probing classifier is trained on the standard benchmark dataset OntoNotes 5.0 [70] using the jiant toolkit [54]. The structural probe is trained on the English UD [63] using the code from the authors [24]. For word similarities we use the WordSim353 dataset [17], while for word analogies we use the Google dataset [40]. All those datasets are in English.

2.2 Results

First, we note that the lottery ticket hypothesis is confirmed for the embedding models since pruning up to 60% weights does not harm their performance significantly on held-out data (Fig. 2-1). In the case of SGNS, pruning up to 80% of weights does not affect its validation loss. Since solving SGNS objective is essentially a factorization of the pointwise mutual information matrix, in the form $\text{PMI} - \log k \approx \mathbf{WC}$ [33], this means that a factorization with sparse $\mathbf{W}$ and $\mathbf{C}$ is possible. This observation complements the findings of [68] who showed that near-to-optimal factorization is

possible with binary **W** and **C**.

The probing scores of the baseline (unpruned) models obtained by us are close to the scores from the original papers [66, 26], and are shown in Table 2.1.

Model	Task			
	NE	POS	Const.	Struct.
CoVe	.921	.936	.808	.726

Model	Task	
	Similarity	Analogy
SGNS	.716	.332

Table 2.1: Probing scores for the baseline (unpruned) models. We report the micro-averaged F1 score for the POS, NE, and Constituents; undirected unlabeled attachment score (UUAS) for the structural probe; Spearman's correlation with the human ratings for the similarity task; and accuracy for the analogy task.

Probing results are provided in Fig. 2-2 and 2-3, where we scatter-plot validation loss ℓ_i^ω vs drop in probing performance $\Delta s_i^{\omega,T}$ for each of the model-probe combinations. First, we note that, in most cases, the probing score correlates with the pretraining loss, which supports the rediscovery hypothesis. We note that the probing score decreases slower for some tasks (e.g., POS tagging), but is much steeper for others (e.g., constituents labeling). This is complementary to the findings of [74] who showed that the syntactic learning curve reaches plateau performance with less pretraining data while solving semantic tasks requires more training data. Our results suggest that similar behavior emerges with respect to the model size: simpler tasks (e.g., POS tagging) can be solved with smaller models, while more complex linguistic tasks (e.g., syntactic constituents or dependency parsing) require bigger model size.

We noticed that when restricting the loss values of CoVe to the better performing end (zoomed in regions in Figure 2-4), drop in probing scores (compared to the scores of the baseline unpruned model) become indistinguishable. Assuming that ℓ_i^{CoVe} is the validation loss of CoVe at iteration i, and $\Delta s_i^{\text{CoVe},T}$ is the corresponding drop in the score on the probing task $T \in \{\text{NE, POS, Constituents, Structural}\}$, we obtain pairs $(\ell_i^{\text{CoVe}}, \Delta s_i^{\text{CoVe},T})$ and setup a simple linear regression for the zoomed in regions in Figure 2-4:

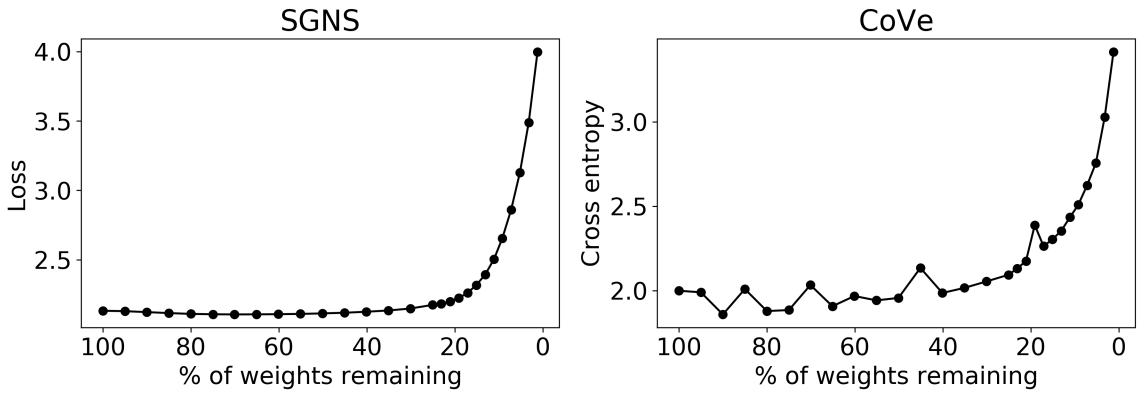

Figure 2-1: Results of applying the LTH (Algorithm 1.3.1) to SGNS and CoVe. In each case we scatter-plot the percentage of remained weights vs validation loss, which is cross entropy for CoVe, and a variant of negative sampling objective for SGNS.

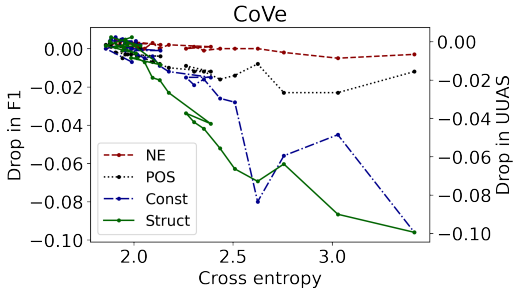

Figure 2-2: Probing results for the CoVe embeddings. Horizontal axes indicate validation loss values, which are cross-entropy values. Vertical axes indicate drops in probing performances. In case of edge probing, we use the NE, POS, and constituent labeling tasks from the suite of [66] and report the drop in micro-averaged F1 score compared to the baseline (unpruned) model. In case of structural probing, we use the distance probe of [26] and report the drop in undirected unlabeled attachment score (UUAS).

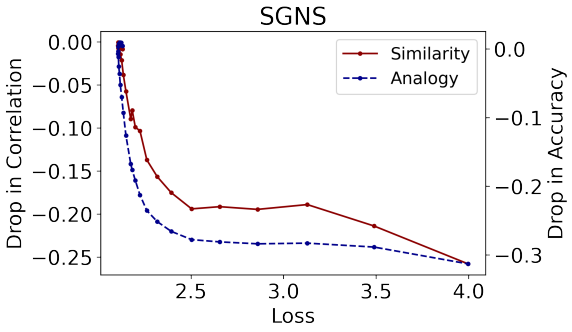

Figure 2-3: Similarity and analogy results for the SGNS embeddings. For similarities we report the drop in Spearman's correlation with the human ratings and for analogies in accuracy.

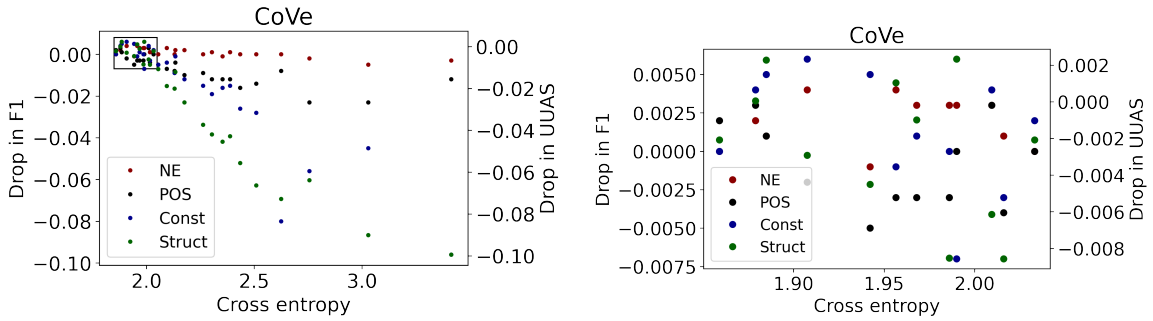

Figure 2-4: Zooming in on better performing CoVe models. Indication of the region to be zoomed in (left) and the zoomed in region (right).

$$\Delta s_i^{\text{CoVe},T} = \alpha + \beta \cdot \ell_i^{\text{CoVe}} + \epsilon_i.$$

Below are p-values when testing $H_0 : \beta = 0$ with a two-sided Student t-test. In all

Task	NE	POS	Constituents	Structural
p-value	0.725	0.306	0.184	0.160

Table 2.2: Testing $\beta = 0$, reported are p-values.

cases β is *not* significantly different from 0. Hence, the correlation between model performance and its linguistic knowledge may become very weak for better solutions to the pretraining objectives. We argue that this phenomenon is related to the amount of information remaining in an LM after pruning, and how much of this information is enough for the probe. It turns out that in the best CoVe (with the lowest pretraining loss) such information is redundant for the probe, and in the not-so-good (but still decent) CoVe there is just enough of information for the probe to show its best performance.

Chapter 3

An Information-Theoretic Framework

3.1 An Information-Theoretic Framework

Recall that the rediscovery hypothesis asserts that neural language models, in the process of their pretraining, rediscover linguistic knowledge. We will prove this claim by contraposition, which states that without linguistic knowledge, the neural LMs cannot perform at their best. A recent paper of [14] has already investigated how linearly removing certain linguistic information from BERT's layers impacts its accuracy of predicting a masked token, where *removing linearly* means that a linear classifier cannot predict the required linguistic property with above majority class accuracy. They showed *empirically* that dependency information, part-of-speech tags, and named entity labels are important for word prediction, while syntactic constituency boundaries (which mark the beginning and the end of a phrase) are not. One of the questions raised by the authors is how to *quantify* the relative importance of different properties encoded in the representation for the word prediction task. The current section of this work attempts to answer this question—we provide a metric ρ that is a reliable predictor of such importance. This metric occurs naturally when we take an information-theory lens and develop a theoretical framework that ties together lin-

guistic properties, word representations, and language modeling performance. We show that when a linguistic property is removed from word vectors, the decline in the quality of a language model depends on how strongly the removed property is interdependent with the underlying text, which is measured by ρ: *a greater ρ leads to a greater drop.*

The proposed metric has an undeniable advantage: its calculation does not require word representations themselves or a pretrained language model. All that is needed is the text and its linguistic annotation. Thanks to our Theorem 1, we can express the influence of a linguistic property on the word prediction task in terms of the coefficient ρ.

3.1.1 Notation

We will use plain-faced lowercase letters (x) to denote scalars and plain-faced uppercase letters (X) for random variables. Bold-faced lowercase letters ($\mathbf{x}$) will denote vectors—both random and non-random—in the Euclidean space $\mathbb{R}^d$, while bold-faced uppercase letters ($\mathbf{X}$) will be used for matrices.

Assuming there is a finite vocabulary $\mathcal{W}$, members of that vocabulary are called *tokens*. A sentence $W_{1:n}$ is a sequence of tokens $W_i \in \mathcal{W}$, this is $W_{1:n} = [W_1, W_2, \ldots, W_n]$. A linguistic annotation T of a sentence $W_{1:n}$ may take different forms. For example, it may be a sequence of per-token tags $T = [T_1, T_2, \ldots, T_n]$, or a parse-tree $T = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V} = \{W_1, \ldots, W_n\}$ and edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$. We only require that T is a *deterministic* function of $W_{1:n}$. Although in reality two people can give two different annotations of the same text due to inherent ambiguity of language or different linguistic theories, we will treat T as the final—also called gold—annotation after disagreements are resolved between annotators and a common reference annotation is agreed upon.

A language model is formulated as the probability distribution $q_{\boldsymbol{\theta}}(W_i \mid \boldsymbol{\xi}_i) \approx \Pr(W_i \mid C_i)$, where C_i is the context of W_i (see below for different types of context), and $\boldsymbol{\xi}_i$ is the vector representation of C_i. The cross-entropy loss of such a model is $\ell(W_i, \boldsymbol{\xi}_i) := \mathbb{E}_{(W_i,C_i)\sim\mathcal{D}}[-\log q_{\boldsymbol{\theta}}(W_i \mid \boldsymbol{\xi}_i)]$, where $\mathcal{D}$ is the true joint distribution of word-context pairs (W, C).

Discreteness of representations. Depending on the LM, C_i is usually either the left context $[W_1, \ldots, W_{i-1}]$. Although the possible set of all such contexts $\mathcal{C}$ is infinite, it is still *countable*. Thus the set of all contextual representations $\{\boldsymbol{\xi} : \boldsymbol{\xi}$ is a vector representation of $C \mid C \in \mathcal{C}\}$ is also countable. Hence, we treat $\boldsymbol{\xi}$ as *discrete* random vector.

3.2 Information Theory Background

For a random variable X with the distribution function $p(x)$, its *entropy* is defined as

$$\mathrm{H}[X] := \mathbb{E}_X[-\log p(X)].$$

The quantity inside the expectation, $-\log p(x)$, is usually referred to as *information content* or *surprisal*, and can be interpreted as quantifying the level of "surprise" of a particular outcome x. As $p(x) \to 0$, the surprise of observing x approaches $+\infty$, and conversely as $p(x) \to 1$, the surprise of observing x approaches 0. Then the entropy $\mathrm{H}[X]$ can be interpreted as the average level of "information" or "surprise" inherent in the X's possible outcomes. We will use the following

Property 3.2.1 *If $Y = f(X)$, then*

$$\mathrm{H}[Y] \leq \mathrm{H}[X], \tag{3.1}$$

i.e. the entropy of a variable can only decrease when the latter is passed through a (deterministic) function.

Now let X and Y be a pair of random variables with the joint distribution function $p(x, y)$. The *conditional entropy* of Y given X is defined as

$$\mathrm{H}[Y \mid X] := \mathbb{E}_{X,Y}[-\log p(Y \mid X)],$$

and can be interepreted as the amount of information needed to describe the outcome of Y given that the outcome of X is known. The following property is important for us.

Property 3.2.2 $\mathrm{H}[Y \mid X] = 0 \Leftrightarrow Y = f(X)$.

The *mutual information* of X and Y is defined as

$$\mathrm{I}[X;Y] := \mathbb{E}_{X,Y}\left[\log \frac{p(X,Y)}{p(X) \cdot p(Y)}\right]$$

and is a measure of mutual dependence between X and Y. We will need the following properties of the mutual information.

Property 3.2.3

1. $\mathrm{I}[X;Y] = \mathrm{H}[X] - \mathrm{H}[X \mid Y] = \mathrm{H}[Y] - \mathrm{H}[Y \mid X]$.
2. $\mathrm{I}[X;Y] = 0 \Leftrightarrow$ *X and Y are independent.*
3. *For a function f,* $\mathrm{I}[X;Y] \geq \mathrm{I}[X; f(Y)]$.
4. $\mathrm{H}[X,Y] = \mathrm{H}[X] + \mathrm{H}[Y] - \mathrm{I}[X;Y]$,

Property 3.2.3.3 is known as data processing inequality, and it means that post-processing cannot increase information. By *post-processing* we mean a transforma-

tion $f(Y)$ of a random variable Y, independent of other random variables. In Property 3.2.3.4, $\mathrm{H}[X, Y]$ is the *joint entropy* of X and Y which is defined as

$$\mathrm{H}[X, Y] := \mathbb{E}_{X,Y}[-\log p(X, Y)]$$

and is a measure of "information" associated with the outcomes of the tuple (X, Y).

3.3 Main Result

Our main result is the following

Theorem 1 *Let*

1. $\mathbf{x}_i$ *be a (contextualized) embedding of a token* W_i *in a sentence* $W_{1:n}$*, and denote*

$$\sigma_i := \mathrm{I}[W_i; \mathbf{x}_i] / \mathrm{H}[W_{1:n}], \tag{3.2}$$

2. T *be a linguistic annotation of* $W_{1:n}$*, and the dependence between* T *and* $W_{1:n}$ *is measured by the coefficient*

$$\rho := \mathrm{I}[T; W_{1:n}] / \mathrm{H}[W_{1:n}], \tag{3.3}$$

3. $\rho > 1 - \sigma_i$*,*

4. $\tilde{\mathbf{x}}_i$ *be a (contextualized) embedding of* W_i *that contains no information on* T*.*

Then the decline in the language modeling quality when using $\tilde{\mathbf{x}}_i$ *instead of* $\mathbf{x}_i$ *is approximately supralinear in* ρ*:*

$$\ell(W_i, \tilde{\mathbf{x}}_i) - \ell(W_i, \mathbf{x}_i) \gtrapprox \mathrm{H}[W_{1:n}] \cdot \rho + c \tag{3.4}$$

for $\rho > \rho_0$, with constants $\rho_0 > 0$, and c depending on $\mathrm{H}[W_{1:n}]$ and $\mathrm{I}[W_i; \mathbf{x}_i]$.

The proof is given in Section 3.4. Here we provide a less formal argument. Using visualization tricks as in [45] we can illustrate the essence of the proof by Figure 3-1. First of all, look at the entropy $\mathrm{H}[X]$ as the amount of information in the variable

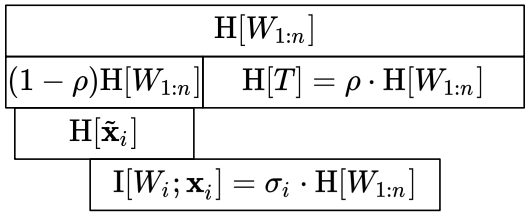

Figure 3-1: Illustration of Theorem 1.

X. Imagine amounts of information as bars. These bars overlap if there is shared information between the respective variables.

The annotation T and the embedding vector $\tilde{\mathbf{x}}_i$ are derived from the underlying text $W_{1:n}$, thus $W_{1:n}$ contains more information than T or $\tilde{\mathbf{x}}_i$, hence $\mathrm{H}[W_{1:n}]$ fully covers $\mathrm{H}[T]$ and $\mathrm{H}[\tilde{\mathbf{x}}_i]$. Since the mutual information $\mathrm{I}[W_i; \tilde{\mathbf{x}}_i]$ cannot exceed the information in $\tilde{\mathbf{x}}_i$, we can write

$$\mathrm{I}[W_i; \tilde{\mathbf{x}}_i] \leq \mathrm{H}[\tilde{\mathbf{x}}_i]. \tag{3.5}$$

Now recall the Eq. (3.3): it simply means that ρ is the fraction of information that is left in T after it was derived from $W_{1:n}$. This immediately implies that the information which is *not* in T (but was in $W_{1:n}$ initially) is equal to $(1-\rho) \cdot \mathrm{H}[W_{1:n}]$.

Since the embedding $\tilde{\mathbf{x}}_i$ contains no information about the annotation T, $\mathrm{H}[\tilde{\mathbf{x}}_i]$ and $\mathrm{H}[T]$ do not overlap. But this means that

$$\mathrm{H}[\tilde{\mathbf{x}}_i] \leq (1-\rho) \cdot \mathrm{H}[W_{1:n}], \tag{3.6}$$

because $\tilde{\mathbf{x}}_i$ is in the category of *no information about* T.

Combining (3.2), (3.5), (3.6), and the assumption $\rho > 1 - \sigma_i$, we have

$$\begin{aligned} &\mathrm{I}[W_i; \tilde{\mathbf{x}}_i] \leq (1-\rho) \cdot \mathrm{H}[W_{1:n}] < \sigma_i \cdot \mathrm{H}[W_{1:n}] = \mathrm{I}[W_i; \mathbf{x}_i], \\ \Rightarrow \quad &\mathrm{I}[W_i; \mathbf{x}_i] - \mathrm{I}[W_i; \tilde{\mathbf{x}}_i] > (\rho + \sigma_i - 1) \cdot \mathrm{H}[W_{1:n}]. \end{aligned} \tag{3.7}$$

The inequality (3.7) is almost the required (3.4)—it remains to show that the change in mutual information can be approximated by the change in LM loss; this is done in Lemma 3.4.2.

Role of ρ and σ_i. Equation (3.3) quantifies the dependence between T and $W_{1:n}$, and it simply means that the annotation T carries $100 \cdot \rho\%$ of information contained in the underlying text $W_{1:n}$ (in the information-theoretic sense). The quantity ρ is well known as the *entropy coefficient* in the information theory literature [53]. It can be thought of as an analog of the correlation coefficient for measuring not only linear or monotonic dependence between numerical variables but any kind of statistical dependence between any kind of variables (numerical and non-numerical). As we see from Equation (3.4), the coefficient ρ plays a key role in predicting the LM degradation when the linguistic structure T is removed from the embedding $\mathbf{x}_i$. In Section 3.5 we give a practical way of its estimation for the case when T is a per-token annotation of $W_{1:n}$.

Similarly, Equation (3.2) means that both W_i and $\mathbf{x}_i$ carry at least $100 \cdot \sigma_i\%$ of information contained in $W_{1:n}$. By Firth's distributional hypothesis [18], which states that "you shall know a word by the company it keeps", we assume that σ_i significantly exceeds zero.

Range of ρ. In general, mutual information is non-negative. Mutual information of the annotation T and the underlying text $W_{1:n}$ cannot exceed information contained in either of these variables, i.e. $0 \leq \mathrm{I}[T; W_{1:n}] \leq \mathrm{H}[W_{1:n}]$, and therefore $\rho \in [0, 1]$.

Absence of information. When we write "$\tilde{\mathbf{x}}_i$ contains no information on T", this means that the mutual information between $\tilde{\mathbf{x}}_i$ and T is zero:

$$\mathrm{I}[T; \tilde{\mathbf{x}}_i] = 0. \tag{3.8}$$

In the language of [50], Equation (3.8) assumes that all probes—even the best—perform poorly in extracting T from $\tilde{\mathbf{x}}_i$. This essentially means that the information on T has been filtered out of $\tilde{\mathbf{x}}$. In practice, we will approximate this with the Iterative Nullspace Projection [55].

3.4 Proof of Theorem 1

The proof is split into two steps: first, in Lemma 3.4.1 we show that the decrease in mutual information $\Delta\,\mathrm{I} := \mathrm{I}[W_i; \mathbf{x}_i] - \mathrm{I}[W_i; \tilde{\mathbf{x}}_i]$ is $\Omega(\rho)$, and then in Lemma 3.4.2 we approximate $\Delta\,\mathrm{I}$ by the increase in cross-entropy loss $\ell(W_i, \tilde{\mathbf{x}}_i) - \ell(W_i, \mathbf{x}_i)$.

Lemma 3.4.1 *Let T be an annotation of a sentence $W_{1:n} = [W_1, \ldots, W_i, \ldots, W_n]$ and let $\mathbf{x}_i$, $\tilde{\mathbf{x}}_i$ be (contextualized) word vectors corresponding to a token W_i such that*

$$\mathrm{I}[T; \tilde{\mathbf{x}}_i] = 0. \tag{3.9}$$

Denote

$$\rho := \mathrm{I}[T; W_{1:n}] / \mathrm{H}[W_{1:n}], \quad \rho \in [0, 1], \tag{3.10}$$

$$\sigma_i := \mathrm{I}[W_i; \mathbf{x}_i] / \mathrm{H}[W_{1:n}], \quad \sigma_i \in [0, 1]. \tag{3.11}$$

If $\rho > 1 - \sigma_i$*, then*

$$\mathrm{I}[W_i; \tilde{\mathbf{x}}_i] < \mathrm{I}[W_i; \mathbf{x}_i]. \tag{3.12}$$

and the difference $\Delta\,\mathrm{I} := \mathrm{I}[W_i; \mathbf{x}_i] - \mathrm{I}[W_i; \tilde{\mathbf{x}}_i]$ *is lowerbounded as*

$$\Delta\,\mathrm{I} \geq (\rho + \sigma_i - 1) \cdot \mathrm{H}[W_{1:n}]. \tag{3.13}$$

From (3.9) and by Property 3.2.3.4, we have

$$\begin{aligned} \mathrm{H}[T, \tilde{\mathbf{x}}_i] &= \mathrm{H}[T] + \mathrm{H}[\tilde{\mathbf{x}}_i] - \underbrace{\mathrm{I}[T; \tilde{\mathbf{x}}_i]}_{0} \\ &= \mathrm{H}[T] + \mathrm{H}[\tilde{\mathbf{x}}_i]. \end{aligned} \tag{3.14}$$

On the other hand, both the annotation T and the word vectors $\tilde{\mathbf{x}}_i$ are obtained from the underlying sentence $W_{1:n}$, and therefore $T = T(W_{1:n})$ and $\tilde{\mathbf{x}}_i = \tilde{\mathbf{x}}_i(W_{1:n})$, i.e. the tuple $(T, \tilde{\mathbf{x}}_i)$ is a function of $W_{1:n}$, and by Property 3.2.1,

$$\mathrm{H}[T, \tilde{\mathbf{x}}_i] \leq \mathrm{H}[W_{1:n}]. \tag{3.15}$$

From (3.14) and (3.15), we get

$$\mathrm{H}[T] + \mathrm{H}[\tilde{\mathbf{x}}_i] \leq \mathrm{H}[W_{1:n}]. \tag{3.16}$$

Since $T = T(W_{1:n})$ and $\tilde{\mathbf{x}}_i = \tilde{\mathbf{x}}_i(W_{1:n})$, by Property 3.2.2 we have $\mathrm{H}[T \mid W_{1:n}] = 0$ and $\mathrm{H}[\tilde{\mathbf{x}}_i \mid W_{1:n}] = 0$, and therefore

$$\mathrm{I}[T; W_{1:n}] = \mathrm{H}[T] - \mathrm{H}[T \mid W_{1:n}] = \mathrm{H}[T]$$

Plugging this into (3.16), rearranging the terms, and taking into account (3.10), we get

$$\mathrm{H}[\tilde{\mathbf{x}}_i] \leq \mathrm{H}[W_{1:n}] - \underbrace{\mathrm{I}[T; W_{1:n}]}_{\rho \cdot \mathrm{H}[W_{1:n}]} = (1 - \rho) \cdot \mathrm{H}[W_{1:n}]. \tag{3.17}$$

Also, by Property 3.2.3.1,

$$\mathrm{I}[W_i; \tilde{\mathbf{x}}_i] = \mathrm{H}[\tilde{\mathbf{x}}_i] - \underbrace{\mathrm{H}[\tilde{\mathbf{x}}_i \mid W_i]}_{\geq 0} \leq \mathrm{H}[\tilde{\mathbf{x}}_i] \tag{3.18}$$

From (3.11), (3.17), (3.18), and the assumption $\rho > 1 - \sigma_i$, we have

$$\mathrm{I}[W_i; \tilde{\mathbf{x}}_i] \leq \mathrm{H}[\tilde{\mathbf{x}}_i] \leq (1 - \rho) \cdot \mathrm{H}[W_{1:n}] < \sigma_i \cdot \mathrm{H}[W_{1:n}] = \mathrm{I}[W_i; \mathbf{x}_i],$$

which implies (3.12) and (3.13).

Equation (3.10) quantifies the dependence between T and $W_{1:n}$, and it means that T carries $100 \cdot \rho\%$ of information contained in $W_{1:n}$. Similarly, Equation (3.11) means that both W_i and $\mathbf{x}_i$ carry at least $100 \cdot \sigma_i\%$ of information contained in $W_{1:n}$. By Firth's distributional hypothesis [18], σ_i significantly exceeds zero.

The mutual information $\mathrm{I}[W_i; \boldsymbol{\xi}_i]$ can be interpreted as the performance of the best language model that tries to predict the token W_i given its contextual representation $\boldsymbol{\xi}_i$.[1] Therefore, inequalities (3.12) and (3.13) mean that removing a linguistic structure from word vectors *does* harm the performance of a language model based on such word

[1] Just as $\mathrm{I}[T_i; \boldsymbol{\xi}_i]$ is interpreted as the performance of the best probe that tries to predict the linguistic property T_i given the embedding $\boldsymbol{\xi}_i$ [51]

vectors, and the more the linguistic structure is interdependent with the underlying text (that is being predicted by the language model) the bigger is the harm.

We treat $\mathrm{I}[W;\boldsymbol{\xi}]$ as the performance of a language model. However, in practice, this performance is measured by the validation loss ℓ of such a model, which is usually the cross-entropy loss. Nevertheless, the change in mutual information *can* be estimated by the change in the LM objective, as we show below.

Lemma 3.4.2 *Let $\mathbf{x}_i$ and $\tilde{\mathbf{x}}_i$ be (contextualized) embeddings of a token W_i in a sentence $W_{1:n} = [W_1, \ldots, W_i, \ldots, W_n]$. Let $\ell(W_i, \boldsymbol{\xi}_i)$ be the cross-entropy loss of a neural (masked) language model $q_{\boldsymbol{\theta}}(w_i \mid \boldsymbol{\xi}_i)$, parameterized by $\boldsymbol{\theta}$, that provides distribution over the vocabulary $\mathcal{W}$ given vector representation $\boldsymbol{\xi}_i$ of the W_i's context. Then*

$$\mathrm{I}[W_i; \mathbf{x}_i] - \mathrm{I}[W_i; \tilde{\mathbf{x}}_i] \approx \ell(W_i, \tilde{\mathbf{x}}_i) - \ell(W_i, \mathbf{x}_i). \tag{3.19}$$

By Property 3.2.3.1, we have

$$\begin{aligned}\Delta\,\mathrm{I} :&= \mathrm{I}[W_i; \mathbf{x}_i] - \mathrm{I}[W_i; \tilde{\mathbf{x}}_i] = \mathrm{H}[W_i] - \mathrm{H}[W_i \mid \mathbf{x}_i] - \mathrm{H}[W_i] + \mathrm{H}[W_i \mid \tilde{\mathbf{x}}_i] \\ &= \mathrm{H}[W_i \mid \tilde{\mathbf{x}}_i] - \mathrm{H}[W_i \mid \mathbf{x}_i] \approx \mathrm{H}_q[W_i \mid \tilde{\mathbf{x}}_i] - \mathrm{H}_q[W_i \mid \mathbf{x}_i],\end{aligned} \tag{3.20}$$

where $\mathrm{H}_q[W_i \mid \boldsymbol{\xi}_i]$—a cross entropy—is an estimate of $\mathrm{H}[W_i \mid \boldsymbol{\xi}_i]$ when the true distribution $p(w_i \mid \boldsymbol{\xi}_i)$ is replaced by a parametric language model $q_{\boldsymbol{\theta}}(w_i \mid \boldsymbol{\xi}_i)$, which is exactly the cross-entropy loss function of a LM q:

$$\mathrm{H}_q[W_i \mid \boldsymbol{\xi}_i] = \mathbb{E}_{(W_i, C_i)\sim\mathcal{D}}[-\log q_{\boldsymbol{\theta}}\,(W_i \mid \boldsymbol{\xi}_i)] =: \ell(W_i, \boldsymbol{\xi}_i). \tag{3.21}$$

From (3.20) and (3.21), we have (3.19).

When we approximate $\mathrm{H}[W_i \mid \mathbf{x}_i] \approx \mathrm{H}_q[W_i \mid \mathbf{x}_i]$, here the parameters $\boldsymbol{\theta}$ of $q = q_{\boldsymbol{\theta}}(w_i \mid \mathbf{x}_i)$ are only the LM head's parameters (e.g., the weights and biases of the

output softmax layer in BERT), and we assume that $\mathbf{x}_i$ can be *any* representation (e.g., from the last encoding layer of BERT). We assume that $\boldsymbol{\theta}$ are chosen to minimize the KL-divergence between the true distribution $p(w_i \mid \mathbf{x}_i)$ and the model $q_{\boldsymbol{\theta}}(w_i \mid \mathbf{x}_i)$, thus making the approximation reasonable.

Similarly, when we approximate $\mathrm{H}[W_i \mid \tilde{\mathbf{x}}_i] \approx \mathrm{H}_q[W_i \mid \tilde{\mathbf{x}}_i]$ we again have the LM head $q_{\boldsymbol{\eta}}(w_i \mid \tilde{\mathbf{x}}_i)$, whose parameters $\boldsymbol{\eta}$ should be chosen to minimize the KL-divergence between true $p(w_i \mid \tilde{\mathbf{x}}_i)$ and the model $q_{\boldsymbol{\eta}}(w_i \mid \tilde{\mathbf{x}}_i)$, for the approximation to be reasonable. In the experimental part (Section 3.5), we do exactly this: after applying removal technique to a representation, we fine-tune the softmax layer (but keep the encoding layers frozen) so that it can adapt to the modified representation.

3.5 Experiments

In this section, we empirically verify the prediction of our theory—the stronger the dependence of a linguistic property with the underlying text the greater the decline in the performance of a language model that does not have access to such property.

Here we will focus on only one contextualized embedding model, BERT because it is the most mainstream model. Along with this, we will keep the SGNS for consideration as a model of static embeddings.

3.5.1 Removal Techniques

INLP [55] is a method of post-hoc removal of some property T from the pretrained embeddings $\mathbf{x}$. INLP neutralizes the ability to linearly predict T from $\mathbf{x}$ (here T is a single tag, and $\mathbf{x}$ is a single vector). It does so by training a sequence of auxiliary models $\tau_1, \ldots, \tau_k$ that predict T from $\mathbf{x}$, interpreting each one as conveying information on unique directions in the latent space that correspond to T, and iteratively removing each of these directions. In the i^{th} iteration, τ_i is a *linear* model param-

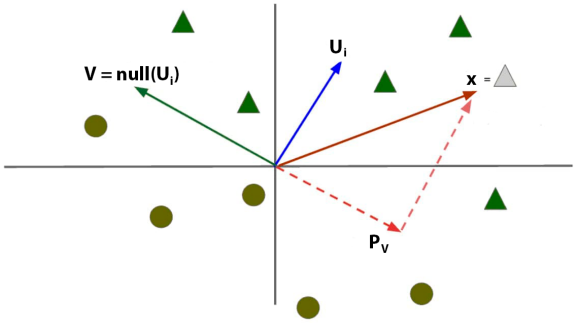

Figure 3-2: Nullspace projection for a 2-dimensional binary classifier. The decision boundary of $\mathbf{U}_i$ is $\mathbf{U}_i$'s null-space. Source: [55].

eterized by a matrix $\mathbf{U}_i$ and trained to predict T from $\mathbf{x}$. [55] use the Linear SVM [11], and we follow their setup. When the embeddings are projected onto null($\mathbf{U}_i$) by a projection matrix $\mathbf{P}_{\text{null}(\mathbf{U}_i)}$, we have

$$\mathbf{U}_i \mathbf{P}_{\text{null}(\mathbf{U}_i)} \mathbf{x} = \mathbf{0},$$

i.e. τ_i will be unable to predict T from $\mathbf{P}_{\text{null}(\mathbf{U}_i)}\mathbf{x}$. Figure 3-2 illustrates the method for the case when the property T has only two types of tags and $\mathbf{x} \in \mathbb{R}^2$. The number of iterations k is taken such that no linear classifier achieves above-majority accuracy when predicting T from $\tilde{\mathbf{x}} = \mathbf{P}_{\text{null}(\mathbf{U}_k)} \mathbf{P}_{\text{null}(\mathbf{U}_{k-1})} \ldots \mathbf{P}_{\text{null}(\mathbf{U}_1)} \mathbf{x}$. Before measuring the LM loss we allow the model to adapt to the modified embeddings $\tilde{\mathbf{x}}$ by fine-tuning its softmax layer that predicts W from $\tilde{\mathbf{x}}$, while keeping the rest of the model (encoding layers) frozen. This fine-tuning does not bring back information on T as the softmax layer linearly separates classes.

Tasks

We perform experiments on two real tasks and a series of synthetic tasks. In real tasks, we cannot arbitrarily change ρ—we can only measure it for some given annotations.

Although we have several real annotations that give some variation in the values of ρ, we want more control over this metric. Therefore, we come up with synthetic annotations that allow us to smoothly change ρ in a wider range and track the impact on the loss function of a language model.

To remove a property T from embeddings, INLP requires an annotated dataset. Therefore for INLP we use gold annotations.

Real tasks. We consider two real tasks with per-token annotations: part-of-speech tagging (POS) and named entity labeling (NER). The choice of per-token annotations is guided by the suggested method of estimating ρ (Section 3.5.1).

POS is the syntactic task of assigning tags such as noun, verb, adjective, etc. to individual tokens. We consider POS tagged datasets that are annotated with different tagsets:

- Universal Dependencies English Web Treebank [63], which uses two annotation schemas: UPOS corresponding to 17 universal tags across languages, and FPOS which encodes additional lexical and grammatical properties.
- English part of the OntoNotes corpus [70] based on the Penn Treebank annotation schema [36].

NER is the task of predicting the category of an entity referred to by a given token, e.g. does the entity refer to a person, a location, an organization, etc. This task is taken from the English part of the OntoNotes corpus. Table 3.1 reports some statistics on the size of different tagsets.

Synthetic tasks are created as follows. Let $T^{(0)}$ be an annotation of a corpus $W_{1:n}$, which has m unique tags, and let the corresponding $\rho^{(0)} := \mathrm{I}[T^{(0)}; W_{1:n}] / \mathrm{H}[W_{1:n}]$. We select the two least frequent tags from the tagset, and conflate them into one tag.

UD UPOS	UD FPOS	ON POS	ON NER
17	50	48	66

Table 3.1: Size of the tagsets for Universal Dependencies and OntoNotes PoS tagging and NER datasets.

This gives us an annotation $T^{(1)}$ which contains less information about $W_{1:n}$ than the annotation $T^{(0)}$, and thus has $\rho^{(1)} < \rho^{(0)}$. In Table 3.2 we give an example of such conflation for a POS-tagged sentence from the OntoNotes corpus [70].

$W_{1:9}$	When	a	browser	starts	to	edge	near	to	consuming
$T^{(0)}$	WRB	DT	NN	VBZ	TO	VB	RB	IN	VBG
$T^{(1)}$	X	DT	NN	VBZ	X	VB	RB	IN	VBG
$W_{10:18}$	500	MB	of	RAM	on	a	regular	basis	,
$T^{(0)}$	CD	NNS	IN	NN	IN	DT	JJ	NN	,
$T^{(1)}$	CD	NNS	IN	NN	IN	DT	JJ	NN	,
$W_{19:22}$	something	is	wrong	.					
$T^{(0)}$	NN	VBZ	JJ	.					
$T^{(1)}$	NN	VBZ	JJ	.					

Table 3.2: Example of conflating two least frequent tags (WRB and TO) into one tag (X).

Next, we select two least frequent tags from the annotation $T^{(1)}$ and conflate them. This will give an annotation $T^{(2)}$ with $\rho^{(2)} < \rho^{(1)}$. Iterating this process $m-1$ times we will end up with the annotation $T^{(m-1)}$ that tags all tokens with a single (most frequent) tag. In this last iteration, the annotation has no mutual information with $W_{1:n}$, i.e. $\rho^{(m-1)} = 0$.

Experimental Setup

We remove (pseudo-)linguistic structures (Section 3.5.1) from BERT and SGNS embeddings using the methods from Section 3.5.1,and measure the decline in the language modeling performance. INLP is applied to the last layers of BERT and SGNS. Assuming that $\Delta\ell_{\omega,T,\mu}$ is the validation loss increase of the embedding model

$\omega \in \{\text{BERT}, \text{SGNS}\}$ when information on $T \in \{\text{Synthetic}, \text{POS}, \text{NER}\}$ is removed from ω using the removal method $\mu \in \{\text{INLP}\}$, we compare $|\Delta\ell_{\omega,T,\mu}|$ against ρ defined by (3.3) which is the strength of interdependence between the underlying text $W_{1:n}$ and its annotation T. By Theorem 1, $\Delta\ell_{\omega,T,\mu}$ is $\Omega(\rho)$ for any combination of ω, T, μ.

Estimating ρ. Recall that $\rho := \mathrm{I}[T; W_{1:n}]/\,\mathrm{H}[W_{1:n}]$ (Eq. 3.3) and that the annotation T is a deterministic function of the underlying text $W_{1:n}$ (Sec. 3.1.1). In this case, we can write

$$\rho = \frac{\mathrm{H}[T] - \overbrace{\mathrm{H}[T \mid W_{1:n}]}^{0}}{\mathrm{H}[W_{1:n}]} = \frac{\mathrm{H}[T]}{\mathrm{H}[W_{1:n}]}. \tag{3.22}$$

and when T is a per-token annotation of $W_{1:n}$, i.e. $T = T_{1:n}$ (which is the case for the annotations that we consider), this becomes $\rho = \mathrm{H}[T_{1:n}]/\,\mathrm{H}[W_{1:n}]$. Thus to estimate ρ, we simply need to be able to estimate the latter two entropies. This can be done by training an autoregressive sequence model, such as LSTM, on $W_{1:n}$ and on $T_{1:n}$. The loss function of such a model—the cross-entropy loss—serves as an estimate of the required entropy. Notice, that we cannot use masked LMs for this estimation as they do not give a proper factorization of the probability $p(w_1, \ldots, w_n)$. Thus, we decided to choose the AWD-LSTM-MoS model of [73] which is a compact and competitive LM that can be trained in a reasonable time and with moderate computing resources. In addition, we also estimated the entropies through a vanilla LSTM with tied input and output embeddings [27], and a Kneser-Ney 3-gram model [44] to test how strongly our method depends on the underlying sequence model.

Limitations. The suggested method of estimating ρ through autoregressive sequence models is limited to per-token annotations only. However, according to formula (3.22), to estimate ρ for deeper annotations, it is sufficient to be able to estimate the entropy $\mathrm{H}[T]$ of such deeper linguistic structures T. For example, to estimate the

entropy of a parse tree, one can use the cross-entropy produced by a probabilistic parser. The only limitation is the determinism of the annotation process.

Amount of information to remove. INLP has a hyperparameter that controls how much of the linguistic information T is removed from the word vectors $\mathbf{x}$—the number of iterations. Following [14] we keep iterating the INLP procedure until the performance of a linear probe that predicts T from the filtered embeddings $\tilde{\mathbf{x}}$ drops to the majority-class accuracy. When this happens we treat the resulting filtered embeddings $\tilde{\mathbf{x}}$ as containing no information on T.

Optimization details. For the INLP experiments we use pretrained BERT-BASE from HuggingFace [71], and an SGNS pretrained in-house [5].

3.5.2 Results

Real tasks. Table 3.3 reports the loss drop of pretrained LM when removing linguistic information (POS or NER) from the pretrained model.

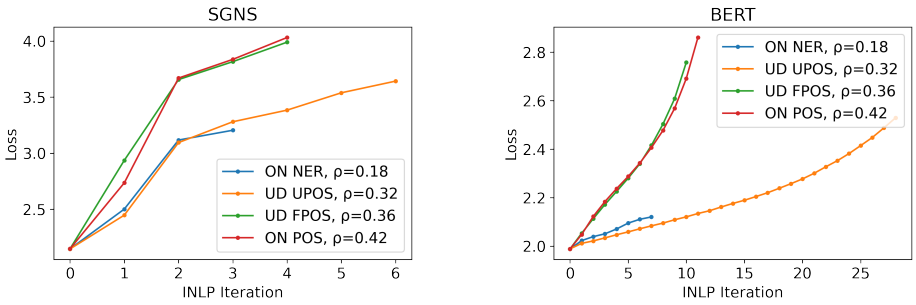

Figure 3-3: Loss increase w.r.t. INLP iteration for different tasks. ON stands for OntoNotes, UD for Universal Dependencies. The UD EWT dataset has two types of POS annotation: coarse tags (UPOS) and fine-grained tags (FPOS)

Figure 3-3 shows how quickly and how much the pre-training loss grows when applying the INLP procedure for various linguistic tasks. For each task, we show the

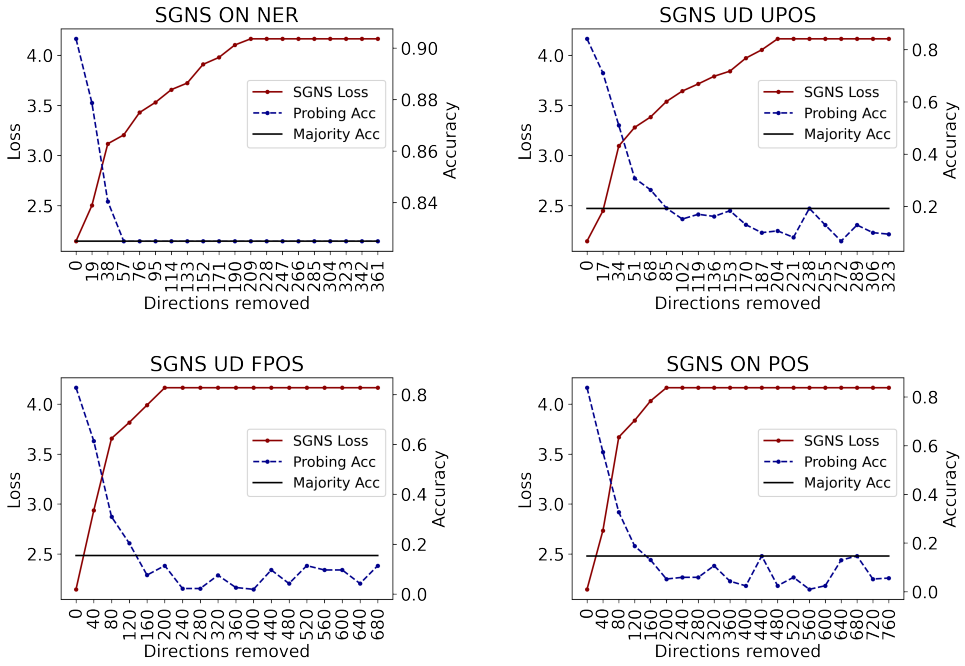

Figure 3-4: INLP Dynamics for SGNS. ON stands for OntoNotes, UD for Universal Dependencies. The UD EWT dataset has two types of POS annotation: coarse tags (UPOS) and fine-grained tags (FPOS).

minimum number of iterations at which the probing accuracy drops to (or below) the level of the majority accuracy.

We illustrate how pre-training loss and probing accuracy change with respect to INLP iteration in Figures 3-4 and 3-5. The number of directions being removed at each INLP iteration is equal to the number of tags in the respective task.

First, we compare UPOS tagsets versus FPOS tagset: intuitively FPOS should have a tighter link with underlying text, and therefore result in higher ρ and as a consequence in a higher drop in loss after the removal of this information from words representations. This is confirmed by the numbers reported in Table 3.3.

We also see a greater LM performance drop when the POS information is removed from the models compared to NE information removal. This is in line with Theorem 1 as POS tags depend stronger on the underlying text than NE labels as measured by ρ.

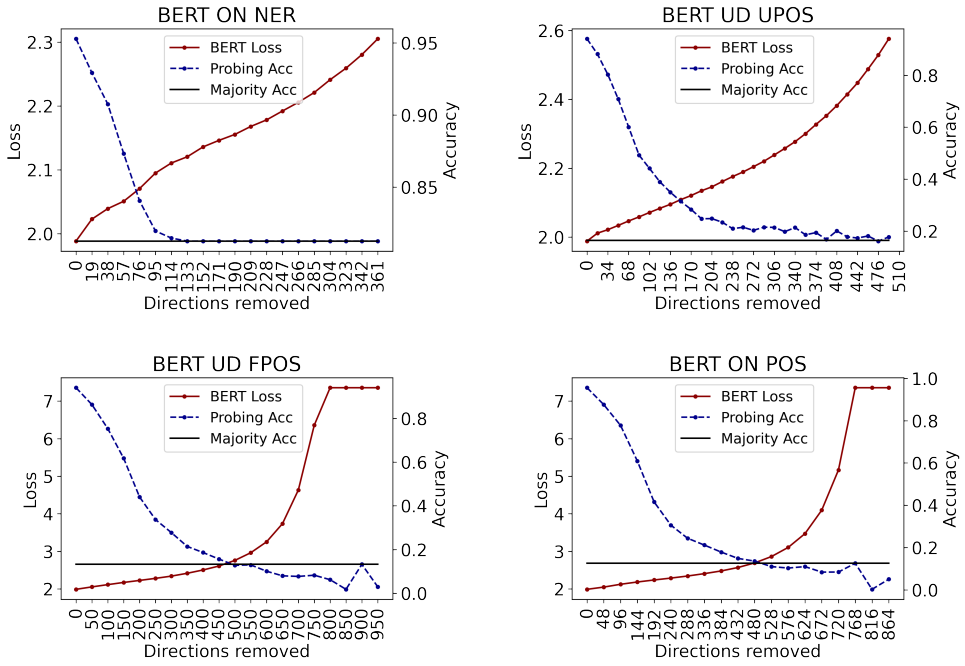

Figure 3-5: INLP Dynamics for BERT.

Finally, we see that although ρ indeed depends on the underlying sequence model that is used to estimate the entropies $\mathrm{H}[T_{1:n}]$ and $\mathrm{H}[W_{1:n}]$, all models—AWD-LSTM-MoS, LSTM, and KN-3—preserve the relative order for the annotations that we consider. E.g., all models indicate that OntoNotes NER annotation is the least interdependent with the underlying text, while OntoNotes POS annotation is the most interdependent one. In addition, it turns out that for a quick estimate of ρ, one can use the KN-3 model, which on a modern laptop calculates the entropy of texts of 100 million tokens in a few minutes, in contrast to the LSTM, which takes several hours on a modern GPU.

Synthetic tasks. To obtain synthetic data, we apply the procedure described in Sect. 3.5.1 to the OntoNotes POS annotation as it has the highest ρ in Table 3.3 and thus allows us to vary the metric in a wider range.

The results of evaluation on the synthetic tasks through the INLP are provided

Removal method μ	INLP			
Annotation T	ON NER	UD UPOS	UD FPOS	ON POS
ρ (AWD-LSTM-MoS)	0.18	0.32	0.36	0.42
ρ (LSTM)	0.18	0.32	0.37	0.42
ρ (KN-3)	0.18	0.36	0.41	0.42
$\Delta\ell_{\text{BERT},T,\mu}$	0.13	0.54	0.70	0.87
$\Delta\ell_{\text{SGNS},T,\mu}$	1.33	1.62	1.79	2.04

Table 3.3: Results on POS tagging and NER tasks. ON stands for OntoNotes, UD for Universal Dependencies. The UD EWT dataset has two types of POS annotation: coarse tags (UPOS) and fine-grained tags (FPOS). KN-3 is a Kneser-Ney 3-gram model.

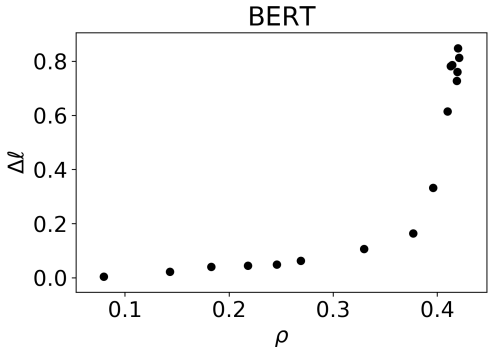

Figure 3-6: Synthetic task results for INLP. $\Delta\ell$ is the increase in cross-entropy loss when pseudo-linguistic information is removed from the BERT's last layer with the INLP procedure. ρ is estimated with the help of AWD-LSTM-MoS model [73] as described in Section 3.5.1.

in Figure 3-6. They validate the predictions of our theory—for the annotations with greater ρ there is a bigger drop in the LM performance (i.e. increase in the LM loss) when the information on such annotations is removed from the embeddings. We notice that $\Delta\ell$ is piecewise-linear in ρ with the slope changing at $\rho \approx 0.4$. We attribute this change to the following: for $\rho < 0.4$, the majority class (i.e. the most frequent tag) is the tag that encapsulates several conflated tags (see Subsection 3.5.1 for details), while for $\rho > 0.4$, the majority is NN tag. This switch causes a significant drop in the majority class accuracy which in turn causes a significant increase in the number of INLP iterations to reach that accuracy, and hence an increase in the amount of

information being removed which implies greater degradation of the LM performance.

3.6 Related Work

Theoretical analysis. Since the success of early word embedding algorithms like SGNS and GloVe, there were attempts to analyze theoretically the connection between their pretraining objectives and performance on downstream tasks such as word similarity and word analogy tasks. An incomplete list of such attempts includes those of [33, 4, 23, 20, 67, 15, 2]. Most of these works represent pretraining as a low-rank approximation of some co-occurrence matrix—such as PMI—and then use an empirical fact that the set of columns (or rows) of such a matrix is already a good solution to the analogy and similarity tasks. Recently, we have seen a growing number of works devoted to the theoretical analysis of contextualized embeddings. [29] showed that modern embedding models, as well as the old warrior SGNS, maximize an objective function that is a lower bound on the mutual information between different parts of the text. [32] formalized how solving certain pretraining tasks allows learning representations that provably decrease the sample complexity of downstream supervised tasks. Of particular interest is a recent paper by [60] that relates a pretraining performance of an autoregressive LM with a downstream performance for downstream tasks that *can* be reformulated as next word prediction tasks. The authors showed that for such tasks, if the pretraining objective is ϵ-optimal,[2] then the downstream objective of a linear classifier is $\mathcal{O}(\sqrt{\epsilon})$-optimal. In Section 3.1 we prove a similar statement, but the difference is that we study how the removal of linguistic information affects the pretraining objective and our approach is not limited to downstream tasks that can be reformulated as next word prediction.

[2] [60] say that the pre-training loss ℓ is ϵ-*optimal* if $\ell - \ell^* \leq \epsilon$, where ℓ^* is the minimum achievable loss.

Probing. Early work on probing tried to analyze LSTM language models [34, 62, 1, 10, 22, 30, 66]. Moreover, word similarity [17] and word analogy [43] tasks can be regarded as non-parametric probes of static embeddings such as SGNS [41] and GloVe [46]. Recently the probing approach has been used mainly for the analysis of contextualized word embeddings. [26] for example showed that entire parse trees can be linearly extracted from ELMo's [47] and BERT's [12] hidden layers. [66] probed contextualized embeddings for various linguistic phenomena and showed that, in general, contextualized embeddings improve over their non-contextualized counterparts largely on syntactic tasks (e.g., constituent labeling) in comparison to semantic tasks (e.g., coreference). The probing methodology has also shown that BERT learns some reflections of semantics [57] and factual knowledge [49] into the linguistic form which are useful in applications such as word sense disambiguation and question answering respectively. [74] analyzed how the quality of representations in a pretrained model evolves with the amount of pretraining data. They performed extensive probing experiments on various NLU tasks and found that pretraining with 10M sentences was already able to solve most of the syntactic tasks, while it required 1B training sentences to be able to solve tasks requiring semantic knowledge (such as Named Entity Labeling, Semantic Role Labeling, and some others as defined by [66]).

[14] propose to look at probing from a different angle, proposing *amnesic probing* which is defined as the drop in performance of a pretrained LM after the relevant linguistic information is removed from one of its layers. The notion of amnesic probing fully relies on the assumption that the amount of the linguistic information contained in the pretrained vectors should correlate with the drop in LM performance after this information is removed. In this work (Section 3.1) we theoretically prove this assumption. While [14] measured LM performance as word prediction accuracy, we focus on the native LM cross-entropy loss. In addition, we answer one of the questions

raised by the authors on how to measure the influence of different linguistic properties on the word prediction task—we provide an easy-to-estimate metric that does exactly this.

Criticism of probing. The probing approach has been criticized from different angles. Our attempt to systematize this line of work is given in Figure 3-7. The semi-

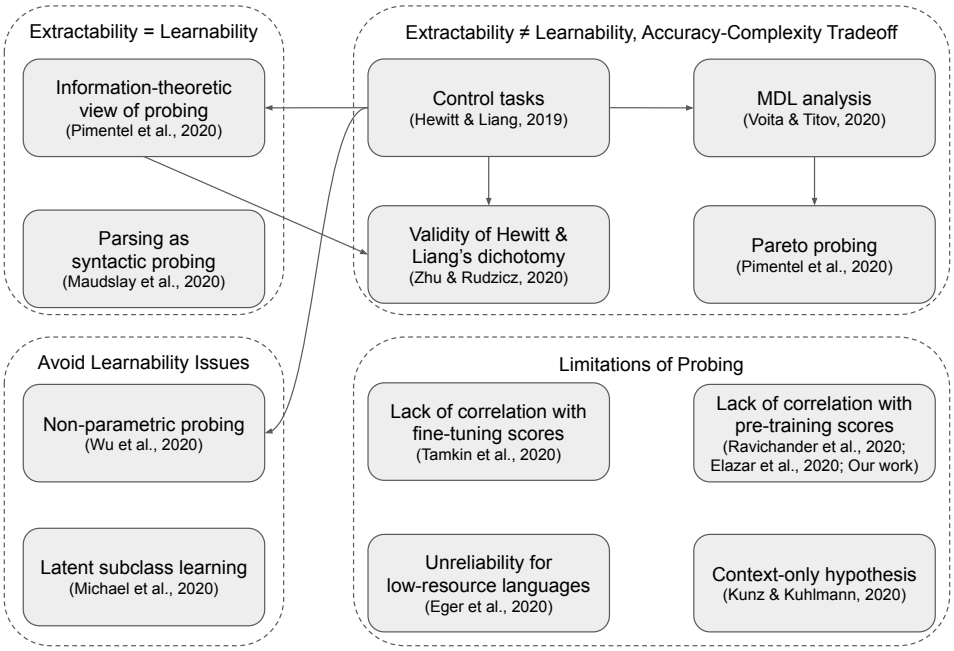

Figure 3-7: Criticism and improvement of the probing methodology. An arrow $A \rightarrow B$ means that B criticizes and/or improves A.

nal paper of [25] raises the issue of separation between *extracting* linguistic structures from contextualized embeddings and *learning* such structures by the probes themselves. This dichotomy was challenged by [51, 37], but later validated by [76] using an information-theoretic view on probing. Meanwhile, methods were proposed that take into account not only the probing performance but also the ease of extracting linguistic information [69] or the complexity of the probing model [50]. At the same time, [72] and [39] suggested avoiding learnability issues by non-parametric probing[3]

[3]Parametric probes transform embeddings to linguistic structures using *parameterized* operations on vectors (such as feedforward layers). Non-parametric probes transform embeddings using *non-*

and weak supervision respectively. The remainder of the criticism is directed at the limitations of probing such as insufficient reliability for low-resourced languages [13], lack of evidence that probes indeed extract linguistic structures but do not learn from the linear context only [31], lack of correlation with fine-tuning scores [64] and with pretraining scores [56, 14]. The first part of our work (Section 2) partly falls into this latter group, as we did not find any evidence for a correlation between probing scores and pretraining objectives for better performing CoVe [38].

Pruning language models. A recent work by [21] compressed BERT using conventional pruning and showed a linear correlation between pretraining loss and downstream task accuracy. [9] pruned pretrained BERT with LTH and fine-tuned it to downstream tasks, while [52] pruned fine-tuned BERT with LTH and then re-fine-tuned it. [59] showed that the weights needed for specific tasks are a small subset of the weights needed for masked language modeling, but they prune *during fine-tuning* which is beyond the scope of our work. [75] propose to learn the masks of the pretrained LM as an alternative to finetuning on downstream tasks and shows that it is possible to find a subnetwork of large pretrained model which can reach the performance on the downstream tasks comparable to finetuning on this task. Generally speaking, the findings of the above-mentioned papers are aligned with our findings that the performance of pruned models on downstream tasks is correlated with the pretraining loss. The one difference from Chapter 2 of our work is that most of the previous work looks at the performance of *fine-tuned* pruned models. In our work, we *probe* pruned models, i.e. the remaining weights of language models are *not* adjusted to the downstream probing task. It is not obvious whether the conclusions from the former should carry over to the latter.

parameterized operations on vectors (such as vector addition/subtraction, inner product, Euclidean distance, etc.). The approach of [72] builds a so-called impact matrix and then feeds it into a graph-based algorithm to induce a dependency tree, all done without learning any parameters.

Chapter 4

Conclusion

In this work, we tried to better understand the phenomenon of the *rediscovery hypothesis* in pretrained language models. Our main contribution is two-fold: we demonstrate that the *rediscovery hypothesis*

- holds across various SGNS and CoVE even when a significant amount of weights get pruned (in English);
- can be formally defined within an information-theoretic framework and proved (assuming that the linguistic annotation is a deterministic function of the underlying text and that the annotation is sufficiently interdependent with the text).

First (Chapter 2), we performed probing of different pruned instances of the original models. If models are overparametrized, then it could be that the pruned model only keeps the connections that are important for the pretraining task, but not for auxiliary tasks like probing. Our experiments show that there is a correlation between the pretrained model's cross-entropy loss and probing performance on various linguistic tasks. We believe that such correlation can be interpreted as strong evidence in favor of the *rediscovery hypothesis*.

Section 3.1 proposes an information-theoretic formulation of the rediscovery hypothesis and a measure of the dependence between the text and linguistic annotations. In particular, Theorem 1 states that neural language models *do need to discover linguistic information* to perform at their best: when the language models are denied access to linguistic information, their performance is not optimal. The *metric* ρ, which naturally appears in the development of our information-theoretic framework proposes a measure of the relationship between a given linguistic property and the underlying text. We show that it is also a reliable predictor of how suboptimal a language model will be if it is denied access to a given linguistic property. Our proposed methods of estimating ρ can be used in further studies to *quantify* the dependence between text and its per-token annotation.

A number of previous studies [66, 10, 26, 74] have shown some evidence in favour of *rediscovery hypothesis*, while many others [69, 51, 25, 14] questioned the methodology of those studies. However, none of the above-mentioned works questioned the rediscovery hypothesis *per se* nor demonstrated any evidence for rejecting it. We believe that our work brings an additional understanding of this phenomenon. In addition, it is the first attempt to formalize the rediscovery hypothesis and provide a theoretical frame for it.

Bibliography

[1] Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[2] Carl Allen and Timothy M. Hospedales. Analogies explained: Towards understanding word embeddings. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 223–231. PMLR, 2019.

[3] Maria Antoniak and David Mimno. Evaluating the stability of embedding-based word similarities. *Trans. Assoc. Comput. Linguistics*, 6:107–119, 2018.

[4] Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. A latent variable model approach to pmi-based word embeddings. *Trans. Assoc. Comput. Linguistics*, 4:385–399, 2016.

[5] Zhenisbek Assylbekov. Sgns implementation in pytorch. https://github.com/zh3nis/SGNS, 2020. Accessed: 2021-11-07.

[6] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[7] Yoshua Bengio and Jean-Sébastien Senecal. Quick training of probabilistic neural nets by importance sampling. In Christopher M. Bishop and Brendan J. Frey, editors, *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics, AISTATS 2003, Key West, Florida, USA, January 3-6, 2003*. Society for Artificial Intelligence and Statistics, 2003.

[8] Mauro Cettolo, Niehues Jan, Stüker Sebastian, Luisa Bentivogli, Roldano Cattoni, and Marcello Federico. The IWSLT 2016 evaluation campaign. In *Proceedings of IWSLT*, 2016.

[9] Tianlong Chen, Jonathan Frankle, Shiyu Chang, Sijia Liu, Yang Zhang, Zhangyang Wang, and Michael Carbin. The lottery ticket hypothesis for pre-trained BERT networks. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[10] Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. What you can cram into a single \$&!#* vector: Probing sentence embeddings for linguistic properties. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2126–2136. Association for Computational Linguistics, 2018.

[11] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Mach. Learn.*, 20(3):273–297, 1995.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019.

[13] Steffen Eger, Johannes Daxenberger, and Iryna Gurevych. How to probe sentence embeddings in low-resource languages: On structural design choices for probing task evaluation. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 108–118, Online, November 2020. Association for Computational Linguistics.

[14] Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Trans. Assoc. Comput. Linguistics*, 9:160–175, 2021.

[15] Kawin Ethayarajh, David Duvenaud, and Graeme Hirst. Towards understanding linear word analogies. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3253–3262. Association for Computational Linguistics, 2019.

[16] Allyson Ettinger, Ahmed Elgohary, and Philip Resnik. Probing for semantic evidence of composition by means of simple classification tasks. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 134–139, Berlin, Germany, August 2016. Association for Computational Linguistics.

[17] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. Placing search in context: the concept revisited. *ACM Trans. Inf. Syst.*, 20(1):116–131, 2002.

[18] John R Firth. A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis. Special volume of the Philological Society*, 22(1), 1957.

[19] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[20] Alex Gittens, Dimitris Achlioptas, and Michael W. Mahoney. Skip-gram - zipf + uniform = vector additivity. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 69–76. Association for Computational Linguistics, 2017.

[21] Mitchell A. Gordon, Kevin Duh, and Nicholas Andrews. Compressing BERT: studying the effects of weight pruning on transfer learning. In Spandana Gella, Johannes Welbl, Marek Rei, Fabio Petroni, Patrick S. H. Lewis, Emma Strubell, Min Joon Seo, and Hannaneh Hajishirzi, editors, *Proceedings of the 5th Workshop on Representation Learning for NLP, RepL4NLP@ACL 2020, Online, July 9, 2020*, pages 143–155. Association for Computational Linguistics, 2020.

[22] Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. Colorless green recurrent networks dream hierarchically. In Marilyn A. Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 1195–1205. Association for Computational Linguistics, 2018.

[23] Tatsunori B. Hashimoto, David Alvarez-Melis, and Tommi S. Jaakkola. Word embeddings as metric recovery in semantic spaces. *Trans. Assoc. Comput. Linguistics*, 4:273–286, 2016.

[24] John Hewitt. structural-probes. `https://github.com/john-hewitt/structural-probes`, 2019. Accessed: 2021-11-07.

[25] John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2733–2743. Association for Computational Linguistics, 2019.

[26] John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4129–4138. Association for Computational Linguistics, 2019.

[27] Hakan Inan, Khashayar Khosravi, and Richard Socher. Tying word vectors and word classifiers: A loss framework for language modeling. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[28] Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. OpenNMT: Open-source toolkit for neural machine translation. In Mohit Bansal and Heng Ji, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, System Demonstrations*, pages 67–72. Association for Computational Linguistics, 2017.

[29] Lingpeng Kong, Cyprien de Masson d'Autume, Lei Yu, Wang Ling, Zihang Dai, and Dani Yogatama. A mutual information maximization perspective of language representation learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[30] Adhiguna Kuncoro, Chris Dyer, John Hale, Dani Yogatama, Stephen Clark, and Phil Blunsom. LSTMs can learn syntax-sensitive dependencies well, but modeling structure makes them better. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1426–1436. Association for Computational Linguistics, 2018.

[31] Jenny Kunz and Marco Kuhlmann. Classifier probes may just learn from linear context features. In Donia Scott, Núria Bel, and Chengqing Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 5136–5146. International Committee on Computational Linguistics, 2020.

[32] Jason D. Lee, Qi Lei, Nikunj Saunshi, and Jiacheng Zhuo. Predicting what you already know helps: Provable self-supervised learning. *CoRR*, abs/2008.01064, 2020.

[33] Omer Levy and Yoav Goldberg. Neural word embedding as implicit matrix factorization. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing*

Systems 2014, December 8-13 2014, Montreal, Quebec, Canada, pages 2177–2185, 2014.

[34] Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Trans. Assoc. Comput. Linguistics*, 4:521–535, 2016.

[35] Matt Mahoney. About the test data. `http://mattmahoney.net/dc/textdata.html`, 2011. Accessed: 2021-11-07.

[36] Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. Building a large annotated corpus of english: The penn treebank. *Comput. Linguistics*, 19(2):313–330, 1993.

[37] Rowan Hall Maudslay, Josef Valvoda, Tiago Pimentel, Adina Williams, and Ryan Cotterell. A tale of a probe and a parser. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7389–7395. Association for Computational Linguistics, 2020.

[38] Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. Learned in translation: Contextualized word vectors. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6294–6305, 2017.

[39] Julian Michael, Jan A. Botha, and Ian Tenney. Asking without telling: Exploring latent ontologies in contextual representations. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6792–6812. Association for Computational Linguistics, 2020.

[40] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In Yoshua Bengio and Yann LeCun, editors, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013.

[41] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119, 2013.

[42] Tomás Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In Lucy Vanderwende, Hal Daumé III,

and Katrin Kirchhoff, editors, *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 746–751. The Association for Computational Linguistics, 2013.

[43] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In Lucy Vanderwende, Hal Daumé III, and Katrin Kirchhoff, editors, *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 746–751. The Association for Computational Linguistics, 2013.

[44] Hermann Ney, Ute Essen, and Reinhard Kneser. On structuring probabilistic dependences in stochastic language modelling. *Comput. Speech Lang.*, 8(1):1–38, 1994.

[45] Christopher Olah. Visual information theory. `http://colah.github.io/posts/2015-09-Visual-Information/`, 2015.

[46] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543. ACL, 2014.

[47] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In Marilyn A. Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 2227–2237. Association for Computational Linguistics, 2018.

[48] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? 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 2463–2473, Hong Kong, China, November 2019. Association for Computational Linguistics.

[49] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick S. H. Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. Language models as knowledge bases? In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2463–2473. Association for Computational Linguistics, 2019.

[50] Tiago Pimentel, Naomi Saphra, Adina Williams, and Ryan Cotterell. Pareto probing: Trading off accuracy for complexity. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 3138–3153. Association for Computational Linguistics, 2020.

[51] Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. Information-theoretic probing for linguistic structure. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4609–4622. Association for Computational Linguistics, 2020.

[52] Sai Prasanna, Anna Rogers, and Anna Rumshisky. When BERT plays the lottery, all tickets are winning. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 3208–3229. Association for Computational Linguistics, 2020.

[53] WHTSAVWT Press, SA Teukolsky, WT Vetterling, and BP Flannery. Conditional entropy and mutual information. In *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 2007.

[54] Yada Pruksachatkun, Philip Yeres, Haokun Liu, Jason Phang, Phu Mon Htut, Alex Wang, Ian Tenney, and Samuel R. Bowman. jiant: A software toolkit for research on general-purpose text understanding models. In Asli Çelikyilmaz and Tsung-Hsien Wen, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, ACL 2020, Online, July 5-10, 2020*, pages 109–117. Association for Computational Linguistics, 2020.

[55] Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7237–7256. Association for Computational Linguistics, 2020.

[56] Abhilasha Ravichander, Yonatan Belinkov, and Eduard H. Hovy. Probing the probing paradigm: Does probing accuracy entail task relevance? *CoRR*, abs/2005.00719, 2020.

[57] Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B. Viégas, Andy Coenen, Adam Pearce, and Been Kim. Visualizing and measuring the geometry of BERT. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing*

Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pages 8592–8600, 2019.

[58] Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A primer in bertology: What we know about how BERT works. *CoRR*, abs/2002.12327, 2020.

[59] Victor Sanh, Thomas Wolf, and Alexander M. Rush. Movement pruning: Adaptive sparsity by fine-tuning. *CoRR*, abs/2005.07683, 2020.

[60] Nikunj Saunshi, Sadhika Malladi, and Sanjeev Arora. A mathematical exploration of why language models help solve downstream tasks. In *International Conference on Learning Representations*, 2021.

[61] Natalie Schluter. The word analogy testing caveat. In Marilyn A. Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 242–246. Association for Computational Linguistics, 2018.

[62] Xing Shi, Inkit Padhi, and Kevin Knight. Does string-based neural MT learn source syntax? In Jian Su, Xavier Carreras, and Kevin Duh, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1526–1534. The Association for Computational Linguistics, 2016.

[63] Natalia Silveira, Timothy Dozat, Marie-Catherine de Marneffe, Samuel R. Bowman, Miriam Connor, John Bauer, and Christopher D. Manning. A gold standard dependency corpus for english. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asunción Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014*, pages 2897–2904. European Language Resources Association (ELRA), 2014.

[64] Alex Tamkin, Trisha Singh, Davide Giovanardi, and Noah D. Goodman. Investigating transferability in pretrained language models. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16-20 November 2020*, pages 1393–1401. Association for Computational Linguistics, 2020.

[65] Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4593–4601. Association for Computational Linguistics, 2019.

[66] 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. What do you learn from context? probing for sentence structure in contextualized word representations. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[67] Ran Tian, Naoaki Okazaki, and Kentaro Inui. The mechanism of additive composition. *Mach. Learn.*, 106(7):1083–1130, 2017.

[68] Julien Tissier, Christophe Gravier, and Amaury Habrard. Near-lossless binarization of word embeddings. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7104–7111. AAAI Press, 2019.

[69] Elena Voita and Ivan Titov. Information-theoretic probing with minimum description length. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 183–196. Association for Computational Linguistics, 2020.

[70] Ralph Weischedel, Sameer Pradhan, Lance Ramshaw, Martha Palmer, Nianwen Xue, Mitchell Marcus, Ann Taylor, Craig Greenberg, Eduard Hovy, Robert Belvin, et al. Ontonotes release 5.0. LDC2013T19, Philadelphia, Penn.: Linguistic Data Consortium, 2013.

[71] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In Qun Liu and David Schlangen, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2020 - Demos, Online, November 16-20, 2020*, pages 38–45. Association for Computational Linguistics, 2020.

[72] Zhiyong Wu, Yun Chen, Ben Kao, and Qun Liu. Perturbed masking: Parameter-free probing for analyzing and interpreting BERT. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4166–4176. Association for Computational Linguistics, 2020.

[73] Zhilin Yang, Zihang Dai, Ruslan Salakhutdinov, and William W. Cohen. Breaking the softmax bottleneck: A high-rank RNN language model. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC,*

Canada, April 30 - May 3, 2018, Conference Track Proceedings. OpenReview.net, 2018.

[74] Yian Zhang, Alex Warstadt, Haau-Sing Li, and Samuel R. Bowman. When do you need billions of words of pretraining data? *CoRR*, abs/2011.04946, 2020.

[75] Mengjie Zhao, Tao Lin, Fei Mi, Martin Jaggi, and Hinrich Schütze. Masking as an efficient alternative to finetuning for pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2226–2241, Online, November 2020. Association for Computational Linguistics.

[76] Zining Zhu and Frank Rudzicz. An information theoretic view on selecting linguistic probes. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 9251–9262. Association for Computational Linguistics, 2020.