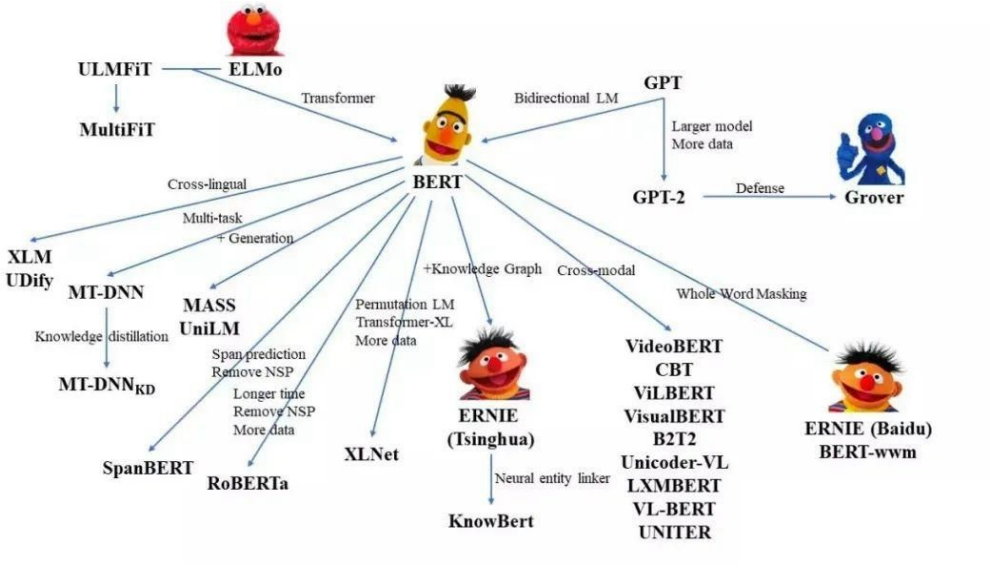
Analyzing & Interpretability in NLP

Ali Modarressi and Hosein Mohebbi

May 22, 2021

Tehran Institute for Advanced Studies


ULMFiT
ELMo
MultiFiT
Transformer
Bidirectional LM
GPT
Larger model
More data
BERT
GPT-2
Defense
Grover
Cross-lingual
Multi-task
+ Generation
XLM
UDify
MT-DNN
MASS
UniLM
Knowledge distillation
MT-DNN$_{KD}$
Span prediction
Remove NSP
Longer time
Remove NSP
More data
SpanBERT
RoBERTa
Permutation LM
Transformer-XL
More data
XLNet
+Knowledge Graph
ERNIE
(Tsinghua)
Neural entity linker
KnowBert
Cross-modal
VideoBERT
CBT
ViLBERT
VisualBERT
B2T2
Unicoder-VL
LXMBERT
VL-BERT
UNITER
Whole Word Masking
ERNIE (Baidu)
BERT-wwm

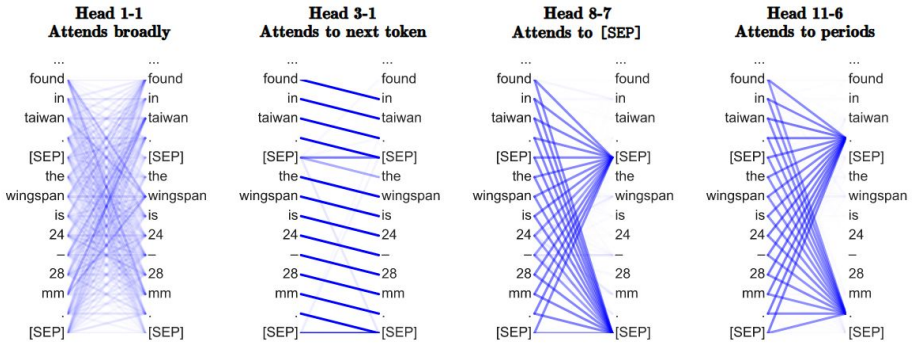

SESAME STREET

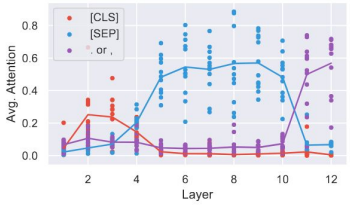
Contents

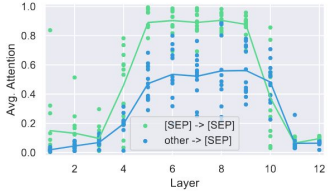
Decision Understanding

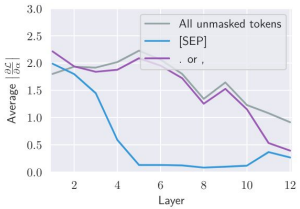
Decision Understanding

Attention-based: raw attentions









What Does BERT Look At?
An Analysis of BERT's Attention

Kevin Clark† Urvashi Khandelwal† Omer Levy‡ Christopher D. Manning†
†Computer Science Department, Stanford University
‡Facebook AI Research
{kevclark,urvashik,manning}@cs.stanford.edu
omerlevy@fb.com

Abstract

Large pre-trained neural networks such as BERT have had great recent success in NLP, motivating a growing body of research investigating what aspects of language they are able to learn from unlabeled data. Most recent analysis has focused on model outputs (e.g., language model surprisal) or internal vector representations (e.g., probing classifiers). Complementary to these works, we propose methods for analyzing the attention mechanisms of pre-trained models and apply them to BERT. BERT's attention heads exhibit patterns such as attending to delimiter tokens, specific positional offsets, or broadly attending over the whole sentence, with heads in the same layer often exhibiting similar behaviors. We further show that certain attention heads correspond well to linguistic notions of syntax and coreference. For example, we find heads that attend to the direct objects of verbs, determiners of nouns, objects of prepositions, and coreferent mentions with remarkably high accuracy. Lastly, we propose an attention-based probing classifier and use it to further demonstrate that substantial syntactic information is captured in BERT's attention.

1 Introduction

Large pre-trained language models achieve very high accuracy when fine-tuned on supervised tasks (Dai and Le, 2015; Peters et al., 2018; Radford et al., 2018), but it is not fully understood why. The strong results suggest pre-training teaches the models about the structure of language, but what specific linguistic features do they learn?

Recent work has investigated this question by examining the *outputs* of language models on carefully chosen input sentences (Linzen et al., 2016) or examining the internal *vector representations* of the model through methods such as probing classifiers (Adi et al., 2017; Belinkov et al., 2017). Complementary to these approaches, we study[1] the *attention maps* of a pre-trained model. Attention (Bahdanau et al., 2015) has been a highly successful neural network component. It is naturally interpretable because an attention weight has a clear meaning: how much a particular word will be weighted when computing the next representation for the current word. Our analysis focuses on the 144 attention heads in BERT[2] (Devlin et al., 2019), a large pre-trained Transformer (Vaswani et al., 2017) network that has demonstrated excellent performance on many tasks.

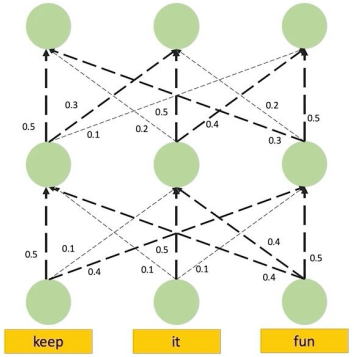
We first explore generally how BERT's attention heads behave. We find that there are common patterns in their behavior, such as attending to fixed positional offsets or attending broadly over the whole sentence. A surprisingly large amount of BERT's attention focuses on the deliminator token [SEP], which we argue is used by the model as a sort of no-op. Generally we find that attention heads in the same layer tend to behave similarly.

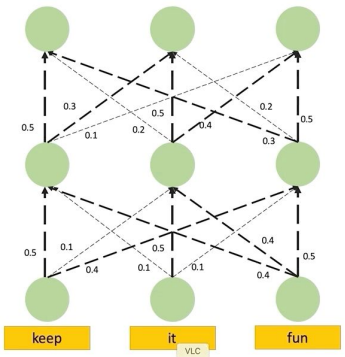
We next probe each attention head for linguistic phenomena. In particular, we treat each head as a simple no-training-required classifier that, given a word as input, outputs the most-attended-to other word. We then evaluate the ability of the heads to classify various syntactic relations. While no single head performs well at many relations, we find that particular heads correspond remarkably well to particular relations. For example, we find heads that find direct objects of verbs, determiners of nouns, objects of prepositions, and objects of possessive pronouns with >75% accuracy. We perform a similar analysis for coreference resolution, also finding a BERT head that performs quite well. These results are intriguing because the behavior of the attention heads emerges purely from self-supervised training on unlabeled data, without explicit supervision for syntax or coreference.

[1]Code will be released at https://github.com/clarkkev/attention-analysis.
[2]We use the English base-sized model.

Decision Understanding

Attention-based: attention rollout and attention flow





Quantifying Attention Flow in Transformers

Samira Abnar
ILLC, University of Amsterdam
s.abnar@uva.nl

Willem Zuidema
ILLC, University of Amsterdam
w.h.zuidema@uva.nl

Abstract

In the Transformer model, "self-attention" combines information from attended embeddings into the representation of the focal embedding in the next layer. Thus, across layers of the Transformer, information originating from different tokens gets increasingly mixed. This makes attention weights unreliable as explanations probes. In this paper, we consider the problem of quantifying this flow of information through self-attention. We propose two methods for approximating the attention to input tokens given attention weights, *attention rollout* and *attention flow*, as post hoc methods when we use attention weights as the relative relevance of the input tokens. We show that these methods give complementary views on the flow of information, and compared to raw attention, both yield higher correlations with importance scores of input tokens obtained using an ablation method and input gradients.

1 Introduction

Attention (Bahdanau et al., 2015; Vaswani et al., 2017) has become the key building block of neural sequence processing models, and visualizing attention weights is the easiest and most popular approach to interpret a model's decisions and to gain insights about its internals (Vaswani et al., 2017; Xu et al., 2015; Wang et al., 2016; Lee et al., 2017; Dehghani et al., 2019; Rocktäschel et al., 2016; Chen and Ji, 2019; Coenen et al., 2019; Clark et al., 2019). Although it is wrong to equate attention with explanation (Pruthi et al., 2019; Jain and Wallace, 2019), it can offer plausible and meaningful interpretations (Wiegreffe and Pinter, 2019; Vashishth et al., 2019; Vig, 2019). In this paper, we focus on problems arising when we move to the higher layers of a model, due to lack of token identifiability of the embeddings in higher layers (Brunner et al., 2020).

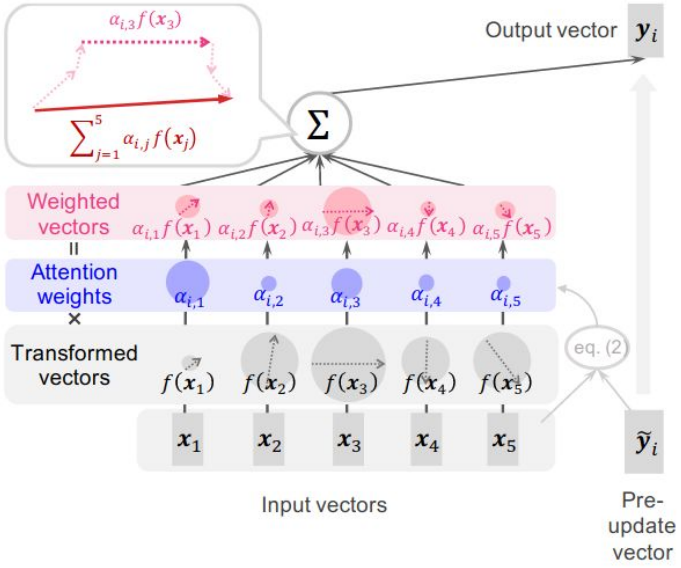
We propose two simple but effective methods to compute attention scores to input tokens (i.e., *token attention*) at each layer, by taking raw attentions (i.e., *embedding attention*) of that layer as well as those from the precedent layers. These methods are based on modelling the information flow in the network with a *DAG* (Directed Acyclic Graph), in which the nodes are input tokens and hidden embeddings, edges are the attentions from the nodes in each layer to those in the previous layer, and the weights of the edges are the attention weights. The first method, *attention rollout*, assumes that the identities of input tokens are linearly combined through the layers based on the attention weights. To adjust attention weights, it rolls out the weights to capture the propagation of information from input tokens to intermediate hidden embeddings. The second method, *attention flow*, considers the attention graph as a flow network. Using a maximum flow algorithm, it computes maximum flow values, from hidden embeddings (sources) to input tokens (sinks). In both methods, we take the residual connection in the network into account to better model the connections between input tokens and hidden embedding. We show that compared to raw attention, the token attentions from attention rollout and attention flow have higher correlations with the importance scores obtained from input gradients as well as *blank-out*, an input ablation based attribution method. Furthermore, we visualize the token attention weights and demonstrate that they are better approximations of how input tokens contribute to a predicted output, compared to raw attention.

It is noteworthy that the techniques we propose in this paper, are not toward making hidden embeddings more identifiable, or providing better attention weights for better performance, but a new set of attention weights that take token identity problem into consideration and can serve as a better diagnostic tool for visualization and debugging.

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4190–4197
July 5 - 10, 2020.

Decision Understanding

Attention-based: attention-norm



Attention is Not Only a Weight: Analyzing Transformers with Vector Norms

Goro Kobayashi[1] Tatsuki Kuribayashi[1,2] Sho Yokoi[1,3] Kentaro Inui[1,3]
[1] Tohoku University [2] Langsmith Inc. [3] RIKEN
{goro.koba, kuribayashi, yokoi, inui}@ecei.tohoku.ac.jp

Abstract

Attention is a key component of Transformers, which have recently achieved considerable success in natural language processing. Hence, attention is being extensively studied to investigate various linguistic capabilities of Transformers, focusing on analyzing the parallels between *attention weights* and specific linguistic phenomena. This paper shows that attention weights alone are only one of the two factors that determine the output of attention and proposes a norm-based analysis that incorporates the second factor, the norm of the transformed input vectors. The findings of our norm-based analyses of BERT and a Transformer-based neural machine translation system include the following: (i) contrary to previous studies, BERT pays poor attention to special tokens, and (ii) reasonable word alignment can be extracted from attention mechanisms of Transformer. These findings provide insights into the inner workings of Transformers.

1 Introduction

Transformers (Vaswani et al., 2017; Devlin et al., 2019; Yang et al., 2019; Liu et al., 2019; Lan et al., 2020) have improved the state-of-the-art in a wide range of natural language processing tasks. The success of the models has not yet been sufficiently explained; hence, substantial research has focused on assessing the linguistic capabilities of these models (Rogers et al., 2020; Clark et al., 2019).

One of the main features of Transformers is that they utilize an attention mechanism without the use of recurrent or convolutional layers. The attention mechanism computes an output vector by accumulating relevant information from a sequence of input vectors. Specifically, it assigns attention weights (i.e., relevance) to each input, and sums up input vectors based on their weights. The analysis of correlations between attention weights and various linguistic phenomena (i.e., *weight-based analysis*) is a prominent research area (Clark et al., 2019; Kovaleva et al., 2019; Reif et al., 2019; Lin et al., 2019; Mareček and Rosa, 2019; Htut et al., 2019; Raganato and Tiedemann, 2018; Tang et al., 2018).

This paper first shows that weight-based analysis is insufficient to analyze the attention mechanism. Weight-based analysis is a common approach to analyze the attention mechanism by simply tracking attention weights. The attention mechanism can be expressed as a *weighted* sum of *linearly transformed vectors* (Section 2.2); however, the effect of transformed vectors in weight-based analysis is ignored. We propose a *norm-based analysis* that considers the previously ignored factors (Section 3). In this analysis, we measure the norms (lengths) of the vectors that were summed to compute the output vector of the attention mechanism.

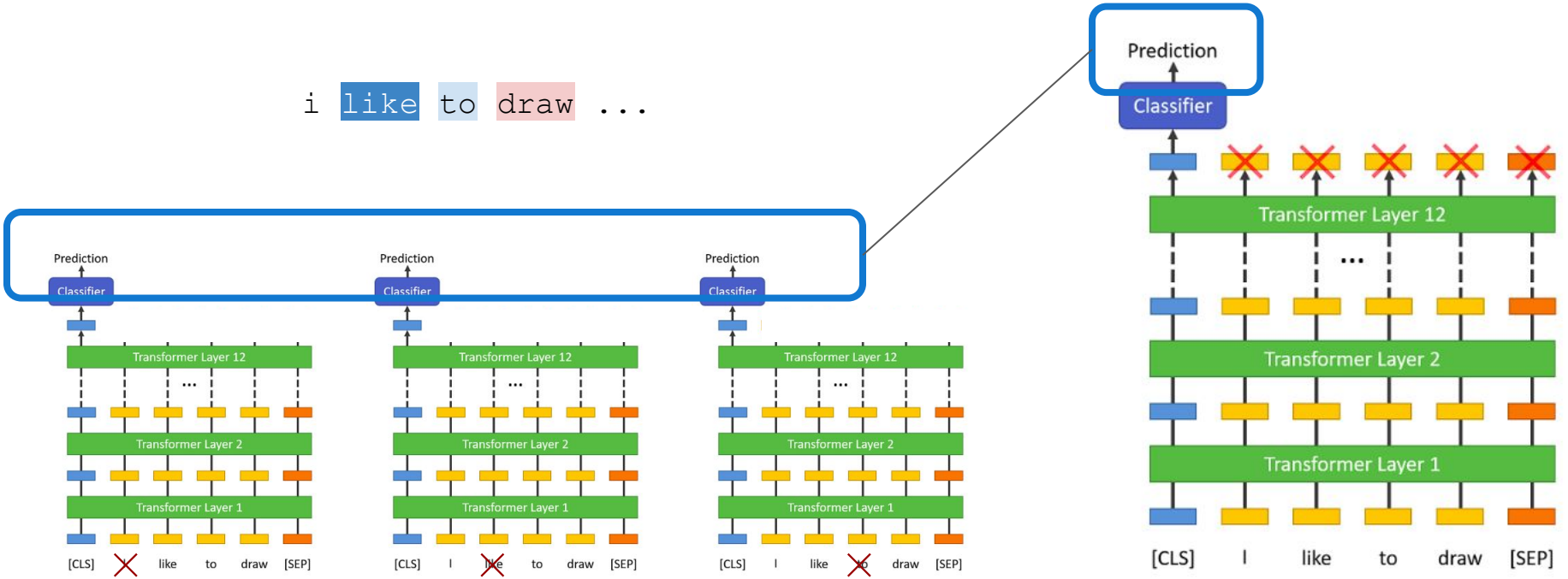
Using the norm-based analysis of BERT (Section 4), we interpreted the internal workings of the model in more detail than when weight-based analysis was used. For example, the weight-based analysis (Clark et al., 2019; Kovaleva et al., 2019) reports that specific tokens, such as periods, commas, and special tokens (e.g., separator token; `[SEP]`), tend to have high attention weights. However, our norm-based analysis found that the information collected from vectors corresponding to special tokens was considerably lesser than that reported in the weight-based analysis, and the large attention weights of these vectors were canceled by other factors. Additionally, we found that BERT controlled the levels of contribution from frequent, less informative words by controlling the norms of their vectors.

In the analysis of a Transformer-based NMT system (Section 5), we reinvestigated how accurate word alignment can be extracted from the

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 7057–7075, November 16–20, 2020.

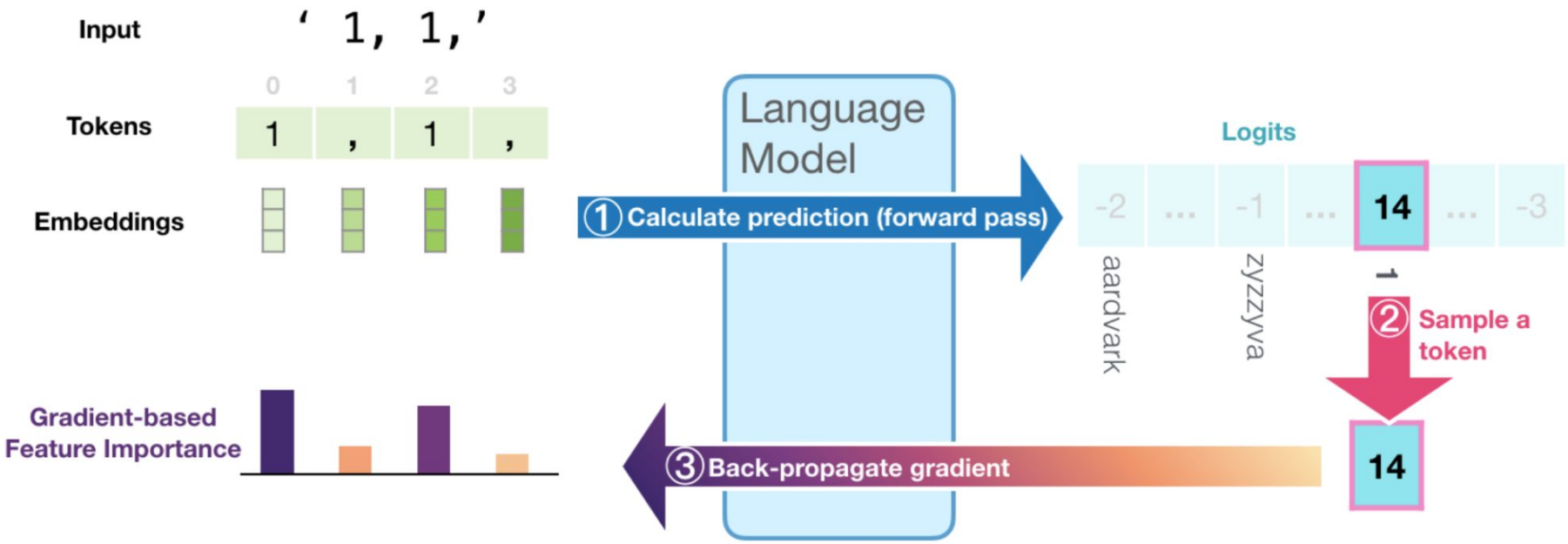
Decision Understanding

Occlusion-based



Decision Understanding

Gradient-based



Decision Understanding

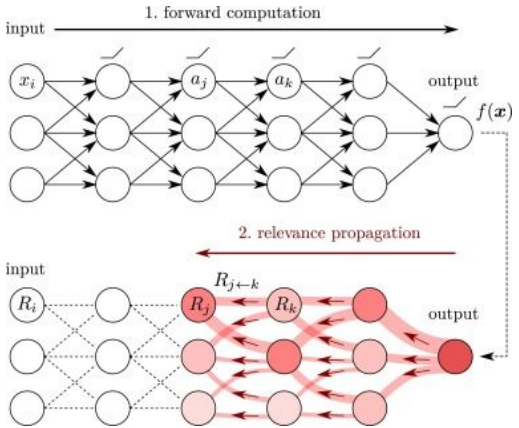
Gradient-based

$$\mathbf{R}_i^{\mathrm{GS}}(x) = \frac{\partial f_c(x)}{\partial x_i}$$

Sensitivity

$$\mathbf{R}_i^{\mathrm{GI}}(x) = x_i \cdot \mathbf{R}_i^{\mathrm{GS}}(x)$$

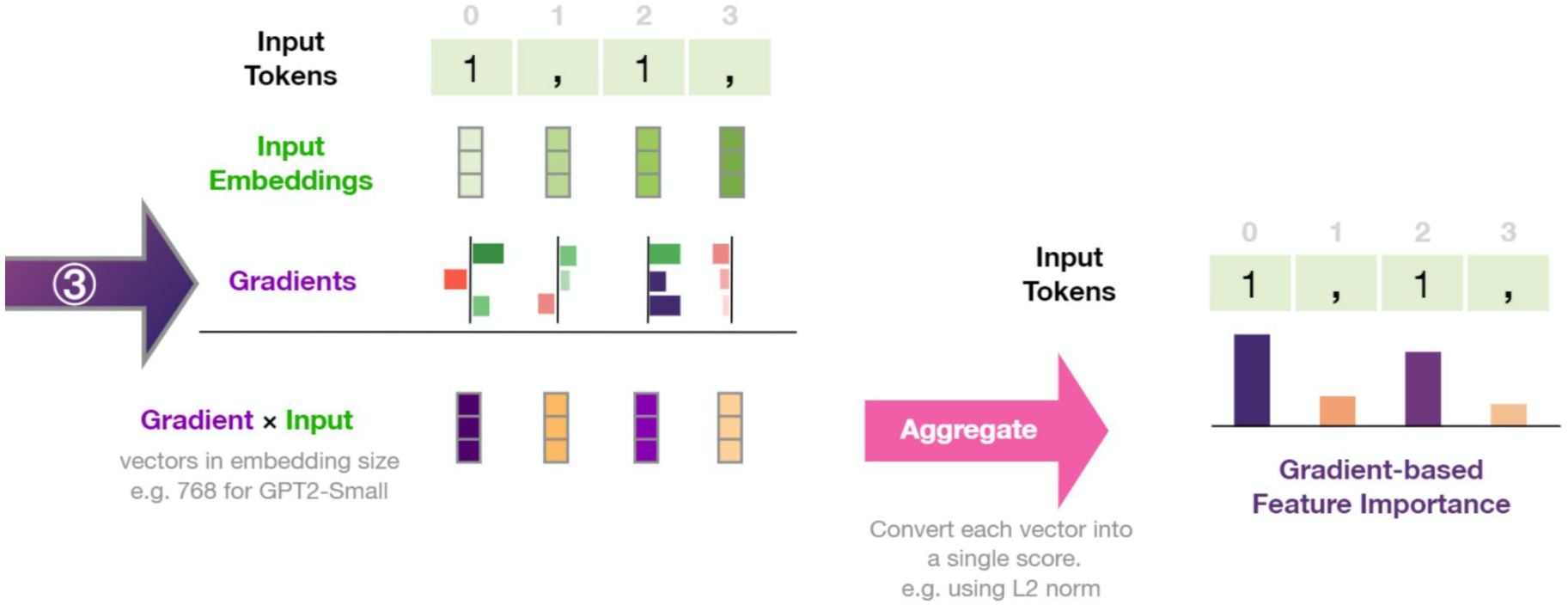
Saliency



Layerwise Relevance Propagation (LRP)

Decision Understanding

Gradient-based

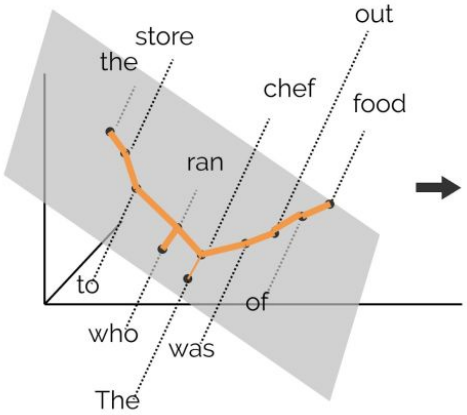


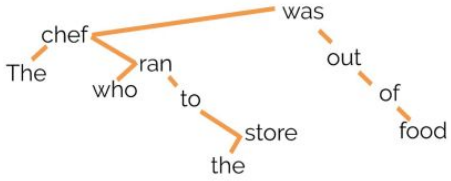
Discussion Part (1)

Representation Understanding

Probing

Structural probing





$$d_B(w_i, w_j)^2 = (h_i - h_j)^T B^T B (h_i - h_j)$$

$$\mathcal{L}(B) = \sum_{k=1}^{N} \frac{1}{|w^{(k)}|^2} \sum_{i=1}^{|w|} \sum_{j=i+1}^{|w|} |\triangle_t(\omega_i, \omega_j) - d_B(\omega_i, \omega_j)^2|$$

A Structural Probe for Finding Syntax in Word Representations

John Hewitt
Stanford University
johnhew@stanford.edu

Christopher D. Manning
Stanford University
manning@stanford.edu

Abstract

Recent work has improved our ability to detect linguistic knowledge in word representations. However, current methods for detecting syntactic knowledge do not test whether syntax trees are represented in their entirety. In this work, we propose a *structural probe*, which 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 L2 distance encodes the distance between words in the parse tree, and one in which squared L2 norm encodes depth in the parse tree. Using our probe, we show that such transformations exist for both ELMo and BERT but not in baselines, providing evidence that entire syntax trees are embedded implicitly in deep models' vector geometry.

1 Introduction

As pretrained deep models that build contextualized representations of language continue to provide gains on NLP benchmarks, understanding what they learn is increasingly important. To this end, probing methods are designed to evaluate the extent to which representations of language encode particular knowledge of interest, like part-of-speech (Belinkov et al., 2017), morphology (Peters et al., 2018a), or sentence length (Adi et al., 2017). Such methods work by specifying a *probe* (Conneau et al., 2018; Hupkes et al., 2018), a supervised model for finding information in a representation.

Of particular interest, both for linguistics and for building better models, is whether deep models' representations encode syntax (Linzen, 2018). Despite recent work (Kuncoro et al., 2018; Peters et al., 2018b; Tenney et al., 2019), open questions remain as to whether deep contextual models encode entire parse trees in their word representations.

In this work, we propose a *structural probe*, a simple model which tests whether syntax trees are consistently embedded in a linear transformation of a neural network's word representation space. Tree structure is embedded if the transformed space has the property that squared L2 distance between two words' vectors corresponds to the number of edges between the words in the parse tree. To reconstruct edge directions, we hypothesize a linear transformation under which the squared L2 norm corresponds to the depth of the word in the parse tree. Our probe uses supervision to find the transformations under which these properties are best approximated for each model. If such transformations exist, they define inner products on the original space under which squared distances and norms encode syntax trees – even though the models being probed were never given trees as input or supervised to reconstruct them. This is a structural property of the word representation space, akin to vector offsets encoding word analogies (Mikolov et al., 2013). Using our probe, we conduct a targeted case study, showing that ELMo (Peters et al., 2018a) and BERT (Devlin et al., 2019) representations embed parse trees with high consistency in contrast to baselines, and in a low-rank space.[1]

In summary, we contribute a simple structural probe for finding syntax in word representations (§2), and experiments providing insights into and examples of how a low-rank transformation recovers parse trees from ELMo and BERT representations (§3,4). Finally, we discuss our probe and limitations in the context of recent work (§5).

2 Methods

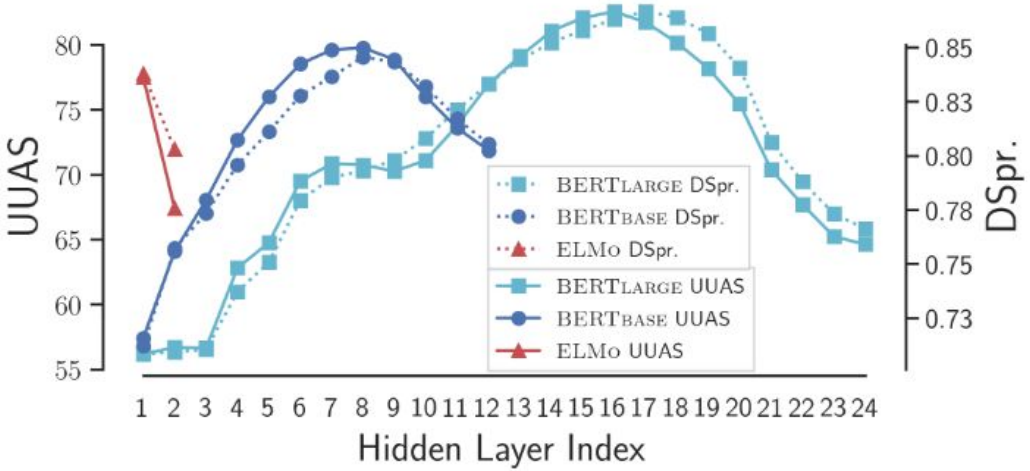
Our goal is to design a simple method for testing whether a neural network embeds each sentence's

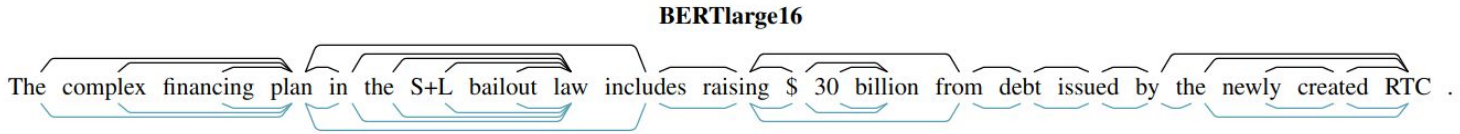
[1]We release our code at https://github.com/john-hewitt/structural-probes.

Proceedings of NAACL-HLT 2019, pages 4129–4138
Minneapolis, Minnesota, June 2 - June 7, 2019.

Probing

Structural probing

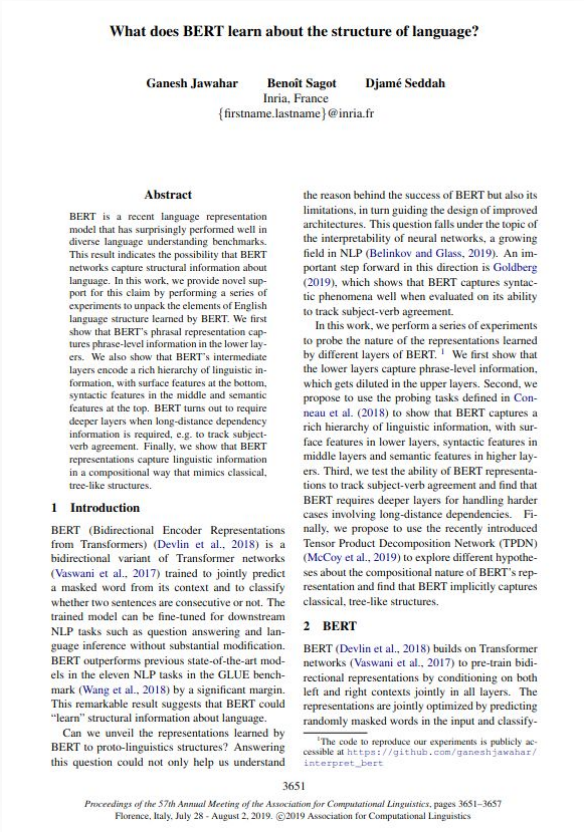




Probing

Sentence-level

Layer	SentLen (Surface)	WC (Surface)	TreeDepth (Syntactic)	TopConst (Syntactic)	BShift (Syntactic)	Tense (Semantic)	SubjNum (Semantic)	ObjNum (Semantic)	SOMO (Semantic)	CoordInv (Semantic)
1	93.9 (2.0)	24.9 (24.8)	35.9 (6.1)	63.6 (9.0)	50.3 (0.3)	82.2 (18.4)	77.6 (10.2)	76.7 (26.3)	49.9 (-0.1)	53.9 (3.9)
2	95.9 (3.4)	65.0 (64.8)	40.6 (11.3)	71.3 (16.1)	55.8 (5.8)	85.9 (23.5)	82.5 (15.3)	80.6 (17.1)	53.8 (4.4)	58.5 (8.5)
3	**96.2 (3.9)**	66.5 (66.0)	39.7 (10.4)	71.5 (18.5)	64.9 (14.9)	86.6 (23.8)	82.0 (14.6)	80.3 (16.6)	55.8 (5.9)	59.3 (9.3)
4	94.2 (2.3)	**69.8 (69.6)**	39.4 (10.8)	71.3 (18.3)	74.4 (24.5)	87.6 (25.2)	81.9 (15.0)	81.4 (19.1)	59.0 (8.5)	58.1 (8.1)
5	92.0 (0.5)	69.2 (69.0)	40.6 (11.8)	81.3 (30.8)	81.4 (31.4)	89.5 (26.7)	85.8 (19.4)	81.2 (18.6)	60.2 (10.3)	64.1 (14.1)
6	88.4 (-3.0)	63.5 (63.4)	**41.3 (13.0)**	83.3 (36.6)	82.9 (32.9)	89.8 (27.6)	**88.1 (21.9)**	82.0 (20.1)	60.7 (10.2)	71.1 (21.2)
7	83.7 (-7.7)	56.9 (56.7)	40.1 (12.0)	**84.1 (39.5)**	83.0 (32.9)	89.9 (27.5)	87.4 (22.2)	**82.2 (21.1)**	61.6 (11.7)	74.8 (24.9)
8	82.9 (-8.1)	51.1 (51.0)	39.2 (10.3)	84.0 (39.5)	83.9 (33.9)	89.9 (27.6)	87.5 (22.2)	81.2 (19.7)	62.1 (12.2)	76.4 (26.4)
9	80.1 (-11.1)	47.9 (47.8)	38.5 (10.8)	83.1 (39.8)	**87.0 (37.1)**	**90.0 (28.0)**	87.6 (22.9)	81.8 (20.5)	63.4 (13.4)	**78.7 (28.9)**
10	77.0 (-14.0)	43.4 (43.2)	38.1 (9.9)	81.7 (39.8)	86.7 (36.7)	89.7 (27.6)	87.1 (22.6)	80.5 (19.9)	63.3 (12.7)	78.4 (28.1)
11	73.9 (-17.0)	42.8 (42.7)	36.3 (7.9)	80.3 (39.1)	86.8 (36.8)	89.9 (27.8)	85.7 (21.9)	78.9 (18.6)	64.4 (14.5)	77.6 (27.9)
12	69.5 (-21.4)	49.1 (49.0)	34.7 (6.9)	76.5 (37.2)	86.4 (36.4)	89.5 (27.7)	84.0 (20.2)	78.7 (18.4)	**65.2 (15.3)**	74.9 (25.4)



What does BERT learn about the structure of language?

Ganesh Jawahar **Benoît Sagot** **Djamé Seddah**
Inria, France
{firstname.lastname}@inria.fr

Abstract

BERT is a recent language representation model that has surprisingly performed well in diverse language understanding benchmarks. This result indicates the possibility that BERT networks capture structural information about language. In this work, we provide novel support for this claim by performing a series of experiments to unpack the elements of English language structure learned by BERT. We first show that BERT's phrasal representation captures phrase-level information in the lower layers. We also show that BERT's intermediate layers encode a rich hierarchy of linguistic information, with surface features at the bottom, syntactic features in the middle and semantic features at the top. BERT turns out to require deeper layers when long-distance dependency information is required, e.g. to track subject-verb agreement. Finally, we show that BERT representations capture linguistic information in a compositional way that mimics classical, tree-like structures.

1 Introduction

BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2018) is a bidirectional variant of Transformer networks (Vaswani et al., 2017) trained to jointly predict a masked word from its context and to classify whether two sentences are consecutive or not. The trained model can be fine-tuned for downstream NLP tasks such as question answering and language inference without substantial modification. BERT outperforms previous state-of-the-art models in the eleven NLP tasks in the GLUE benchmark (Wang et al., 2018) by a significant margin. This remarkable result suggests that BERT could "learn" structural information about language.

Can we unveil the representations learned by BERT to proto-linguistics structures? Answering this question could not only help us understand the reason behind the success of BERT but also its limitations, in turn guiding the design of improved architectures. This question falls under the topic of the interpretability of neural networks, a growing field in NLP (Belinkov and Glass, 2019). An important step forward in this direction is Goldberg (2019), which shows that BERT captures syntactic phenomena well when evaluated on its ability to track subject-verb agreement.

In this work, we perform a series of experiments to probe the nature of the representations learned by different layers of BERT. [1] We first show that the lower layers capture phrase-level information, which gets diluted in the upper layers. Second, we propose to use the probing tasks defined in Conneau et al. (2018) to show that BERT captures a rich hierarchy of linguistic information, with surface features in lower layers, syntactic features in middle layers and semantic features in higher layers. Third, we test the ability of BERT representations to track subject-verb agreement and find that BERT requires deeper layers for handling harder cases involving long-distance dependencies. Finally, we propose to use the recently introduced Tensor Product Decomposition Network (TPDN) (McCoy et al., 2019) to explore different hypotheses about the compositional nature of BERT's representation and find that BERT implicitly captures classical, tree-like structures.

2 BERT

BERT (Devlin et al., 2018) builds on Transformer networks (Vaswani et al., 2017) to pre-train bidirectional representations by conditioning on both left and right contexts jointly in all layers. The representations are jointly optimized by predicting randomly masked words in the input and classify-

[1] The code to reproduce our experiments is publicly accessible at https://github.com/ganeshjawahar/interpret_bert

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3651–3657
Florence, Italy, July 28 - August 2, 2019.

Probing

Span-level --- Edge Probing

POS	The important thing about Disney is that it is a global $[\text{brand}]_1$. $\rightarrow$ NN (Noun)
Constit.	The important thing about Disney is that it $[\text{is a global brand}]_1$. $\rightarrow$ VP (Verb Phrase)
Depend.	$[\text{Atmosphere}]_1$ is always $[\text{fun}]_2$ $\rightarrow$ nsubj (nominal subject)
Entities	The important thing about $[\text{Disney}]_1$ is that it is a global brand. $\rightarrow$ Organization
SRL	$[\text{The important thing about Disney}]_2$ $[\text{is}]_1$ that it is a global brand. $\rightarrow$ Arg1 (Agent)
SPR	$[\text{It}]_1$ $[\text{endorsed}]_2$ the White House strategy... $\rightarrow$ {awareness, existed_after, ...}
Coref.$^{\text{O}}$	The important thing about $[\text{Disney}]_1$ is that $[\text{it}]_2$ is a global brand. $\rightarrow$ True
Coref.$^{\text{W}}$	$[\text{Characters}]_2$ entertain audiences because $[\text{they}]_1$ want people to be happy. $\rightarrow$ True Characters entertain $[\text{audiences}]_2$ because $[\text{they}]_1$ want people to be happy. $\rightarrow$ False
Rel.	The $[\text{burst}]_1$ has been caused by water hammer $[\text{pressure}]_2$. $\rightarrow$ Cause-Effect(e_2, e_1)

Published as a conference paper at ICLR 2019

WHAT DO YOU LEARN FROM CONTEXT? PROBING FOR SENTENCE STRUCTURE IN CONTEXTUALIZED WORD REPRESENTATIONS

Ian Tenney,[*,1] **Patrick Xia,**[2] **Berlin Chen,**[3] **Alex Wang,**[4] **Adam Poliak,**[2]
R. Thomas McCoy,[2] **Najoung Kim,**[2] **Benjamin Van Durme,**[2] **Samuel R. Bowman,**[4]
Dipanjan Das,[1] **and Ellie Pavlick**[1,5]

[1]Google AI Language, [2]Johns Hopkins University, [3]Swarthmore College,
[4]New York University, [5]Brown University

ABSTRACT

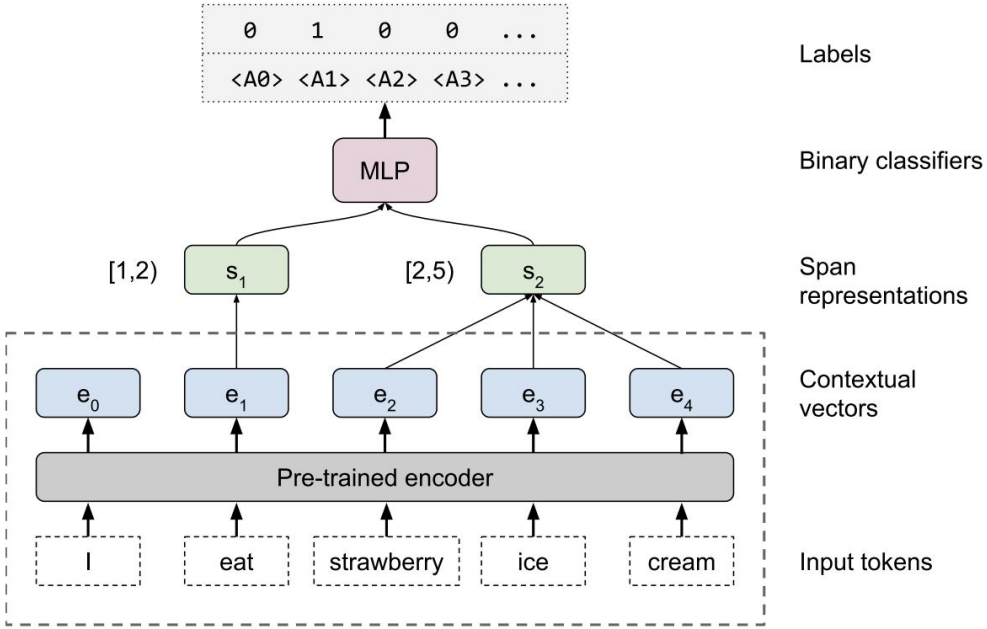
Contextualized representation models such as ELMo (Peters et al., 2018a) and BERT (Devlin et al., 2018) have recently achieved state-of-the-art results on a diverse array of downstream NLP tasks. Building on recent token-level probing work, we introduce a novel *edge probing* task design and construct a broad suite of sub-sentence tasks derived from the traditional structured NLP pipeline. We probe word-level contextual representations from four recent models and investigate how they encode sentence structure across a range of syntactic, semantic, local, and long-range phenomena. We find that existing models trained on language modeling and translation produce strong representations for syntactic phenomena, but only offer comparably small improvements on semantic tasks over a non-contextual baseline.

1 INTRODUCTION[1]

Pretrained word embeddings (Mikolov et al., 2013; Pennington et al., 2014) are a staple tool for NLP. These models provide continuous representations for word types, typically learned from co-occurrence statistics on unlabeled data, and improve generalization of downstream models across many domains. Recently, a number of models have been proposed for *contextualized* word embeddings. Instead of using a single, fixed vector per word type, these models run a pretrained encoder network over the sentence to produce contextual embeddings of each token. The encoder, usually an LSTM (Hochreiter & Schmidhuber, 1997) or a Transformer (Vaswani et al., 2017), can be trained on objectives like machine translation (McCann et al., 2017) or language modeling (Peters et al., 2018a; Radford et al., 2018; Howard & Ruder, 2018; Devlin et al., 2018), for which large amounts of data are available. The activations of this network–a collection of one vector per token–fit the same interface as conventional word embeddings, and can be used as a drop-in replacement input to any model. Applied to popular models, this technique has yielded significant improvements to the state-of-the-art on several tasks, including constituency parsing (Kitaev & Klein, 2018), semantic role labeling (He et al., 2018; Strubell et al., 2018), and coreference (Lee et al., 2018), and has outperformed competing techniques (Kiros et al., 2015; Conneau et al., 2017) that produce fixed-length

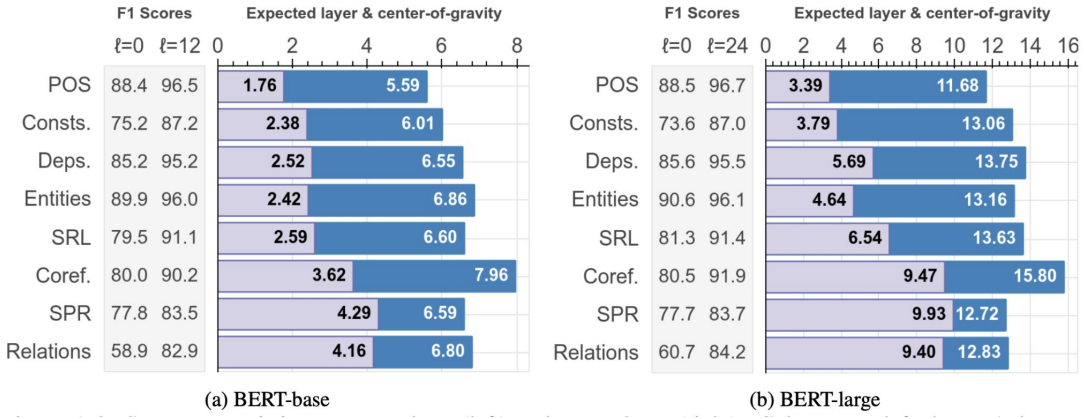
Probing

Span-level --- Edge Probing



Probing

Span-level --- Edge Probing



(a) BERT-base (b) BERT-large

BERT Rediscovers the Classical NLP Pipeline

Ian Tenney[1] **Dipanjan Das**[1] **Ellie Pavlick**[1,2]
[1]Google Research [2]Brown University
{iftenney,dipanjand,epavlick}@google.com

Abstract

Pre-trained text encoders have rapidly advanced the state of the art on many NLP tasks. We focus on one such model, BERT, and aim to quantify where linguistic information is captured within the network. We find that the model represents the steps of the traditional NLP pipeline in an interpretable and localizable way, and that the regions responsible for each step appear in the expected sequence: POS tagging, parsing, NER, semantic roles, then coreference. Qualitative analysis reveals that the model can and often does adjust this pipeline dynamically, revising lower-level decisions on the basis of disambiguating information from higher-level representations.

1 Introduction

Pre-trained sentence encoders such as ELMo (Peters et al., 2018a) and BERT (Devlin et al., 2019) have rapidly improved the state of the art on many NLP tasks, and seem poised to displace both static word embeddings (Mikolov et al., 2013) and discrete pipelines (Manning et al., 2014) as the basis for natural language processing systems. While this has been a boon for performance, it has come at the cost of interpretability, and it remains unclear whether the success of these models is due to robust implementation of compositional features of language, or simply to memorization of large amounts of data.

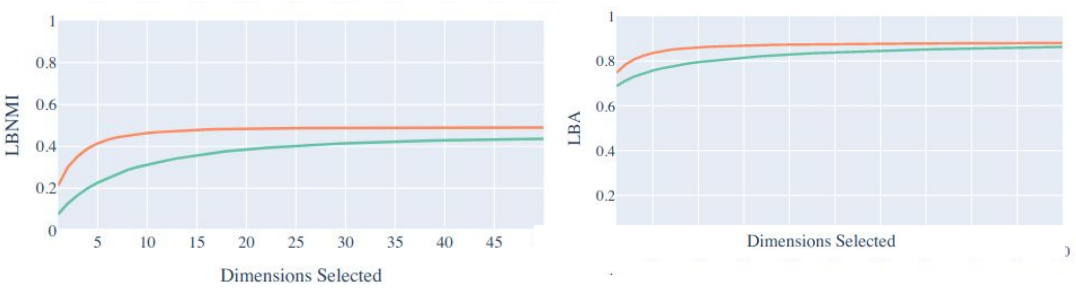
A growing body of work has begun to investigate the inner workings of large pre-trained encoders. Much of this work has found evidence that deep language models can encode a range of syntactic and semantic information (e.g. Shi et al., 2016; Belinkov, 2018; Tenney et al., 2019), and that more complex structures are represented hierarchically in the higher layers of the model (Peters et al., 2018b; Blevins et al., 2018).

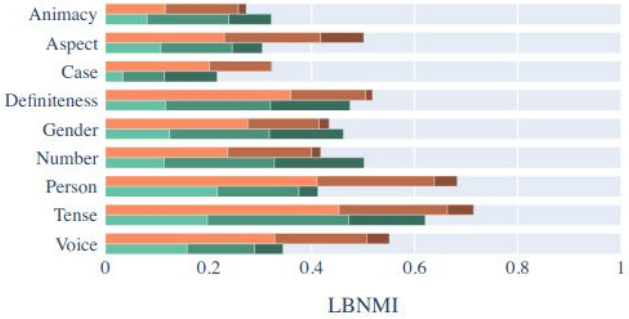
We build on this latter line of work, focusing on the BERT model (Devlin et al., 2019), and use a suite of probing tasks (Tenney et al., 2019) derived from the traditional NLP pipeline to quantify where specific types of linguistic information are encoded. Building on observations (Peters et al., 2018b) that lower layers of a language model encode more local syntax while higher layers capture more complex semantics, we present two novel contributions. First, we present an analysis that spans the common components of a traditional NLP pipeline. We show that the order in which specific abstractions are encoded reflects the traditional hierarchy of these tasks. Second, we qualitatively analyze how individual sentences are processed by the BERT network, layer-by-layer. We

Probing

Dimension-level

fastText vs BERT





Intrinsic Probing through Dimension Selection

Lucas Torroba Hennigen, Adina Williams, Ryan Cotterell
Québec Artificial Intelligence Institute (Mila), University of Cambridge
Facebook AI Research, ETH Zürich
lucas.torroba-hennigen@mila.quebec, adinawilliams@fb.com, ryan.cotterell@inf.ethz.ch

Abstract

Most modern NLP systems make use of pre-trained contextual representations that attain astonishingly high performance on a variety of tasks. Such high performance should not be possible unless some form of linguistic structure inheres in these representations, and a wealth of research has sprung up on probing for it. In this paper, we draw a distinction between intrinsic probing, which examines *how linguistic information is structured* within a representation, and the extrinsic probing popular in prior work, which only argues for the *presence* of such information by showing that it can be successfully extracted. To enable intrinsic probing, we propose a novel framework based on a decomposable multivariate Gaussian probe that allows us to determine whether the linguistic information in word embeddings is dispersed or focal. We then probe fastText and BERT for various morphosyntactic attributes across 36 languages. We find that most attributes are reliably encoded by only a few neurons, with fastText concentrating its linguistic structure more than BERT.[1]

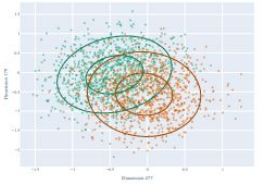


Figure 1: Scatter plot of the two most informative BERT dimensions for English present and past tense. The contours belong to our probe.

1 Introduction

Natural language processing (NLP) is enamored of contextual word representations—and for good reason! Contextual word-embedders, e.g. BERT (Devlin et al., 2019) and ELMo (Peters et al., 2018), have bolstered NLP model performance on myriad tasks, such as syntactic parsing (Kitaev et al., 2019), coreference resolution (Joshi et al., 2019), morphological tagging (Kondratyuk, 2019) and text generation (Zellers et al., 2019). Given the large empirical gains observed when they are employed, it is all but certain that word representations derived from neural networks encode some continuous analogue of linguistic structures.

Exactly *what* these representations encode about linguistic structure, however, remains little understood. Researchers have studied this question by attributing function to specific network cells with visualization methods (Karpathy et al., 2015; Li et al., 2016) and by **probing** (Alain and Bengio, 2017; Belinkov and Glass, 2019), which seeks to extract structure from the representations. Recent work has probed various representations for correlates of morphological (Belinkov et al., 2017; Giulianelli et al., 2018), syntactic (Hupkes et al., 2018; Zhang and Bowman, 2018; Hewitt and Manning, 2019; Lin et al., 2019), and semantic (Kim et al., 2019) structure.

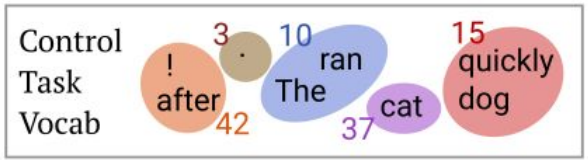
Most current probing efforts focus on what we term **extrinsic probing**, where the goal is to determine whether the posited linguistic structure is predictable from the learned representation. Generally, extrinsic probing works argue for the presence of linguistic structure by showing that it is extractable from the representations using a machine learning model. In contrast, we focus on **intrinsic probing**—whose goals are a proper superset of the

[1] Code and data are available at https://github.com/rycolab/intrinsic-probing.

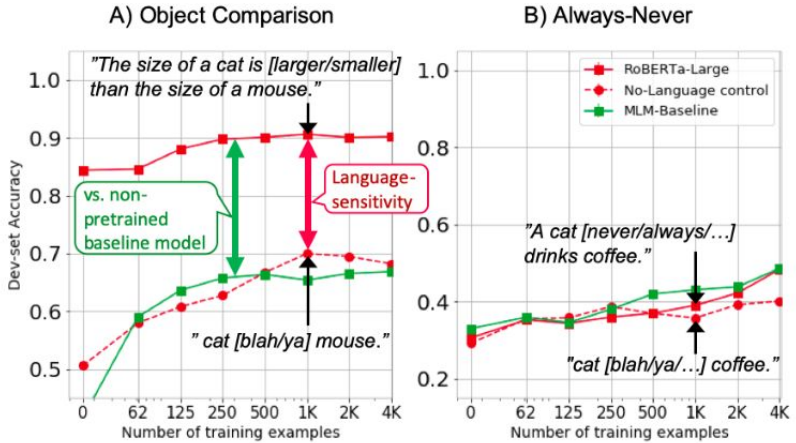
Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 197–216, November 16–20, 2020.

Probing Controls Tasks

Selectivity & Learning Curve

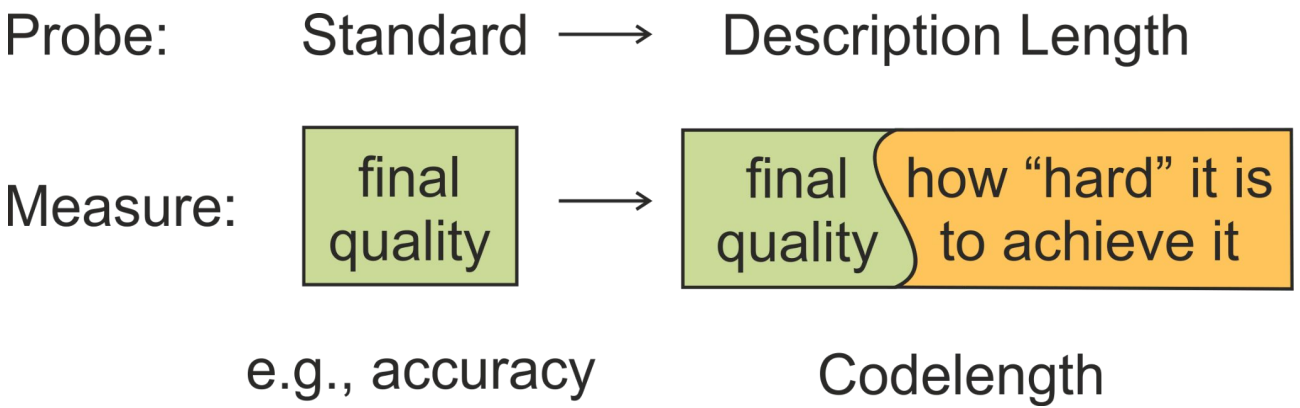


Sentence 1	The	cat	ran	quickly	.
Part-of-speech	DT	NN	VBD	RB	.
Control task	10	37	10	15	3
Sentence 2	The	dog	ran	after	!
Part-of-speech	DT	NN	VBD	IN	.
Control task	10	15	10	42	42



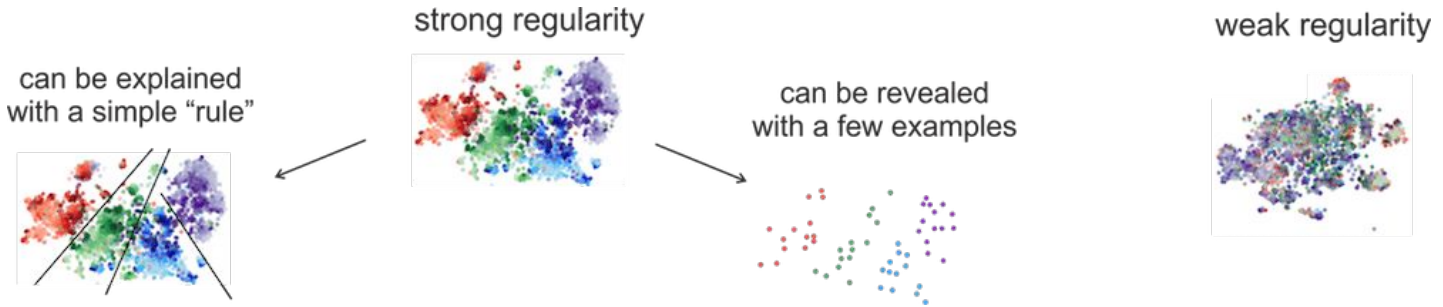
Probing

Information Theoretic Probing



Probing

Information Theoretic Probing

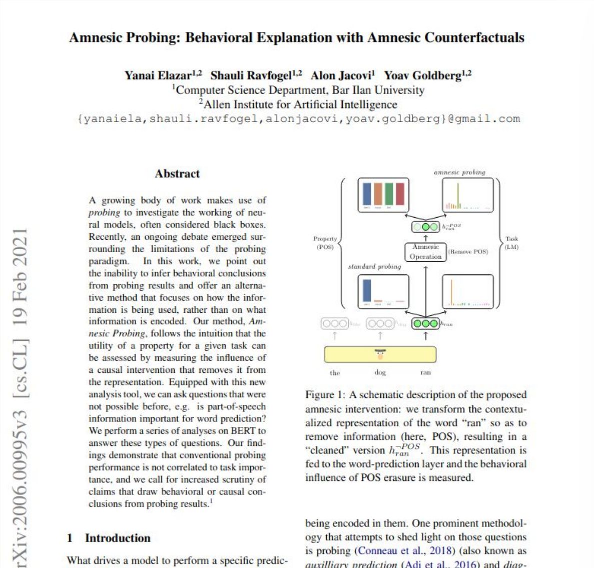


Regularity in representations with respect to labels
can be **exploited to compress** the data.

Shorter Codelength ↔ stronger regularity ↔ representations better encode labels.

Probing

Others



Amnesic Probing: Behavioral Explanation with Amnesic Counterfactuals

Yanai Elazar[1,2] Shauli Ravfogel[1,2] Alon Jacovi[1] Yoav Goldberg[1,2]
[1]Computer Science Department, Bar Ilan University
[2]Allen Institute for Artificial Intelligence
{yanaiela,shauli.ravfogel,alonjacovi,yoav.goldberg}@gmail.com

arXiv:2006.00995v3 [cs.CL] 19 Feb 2021

Abstract

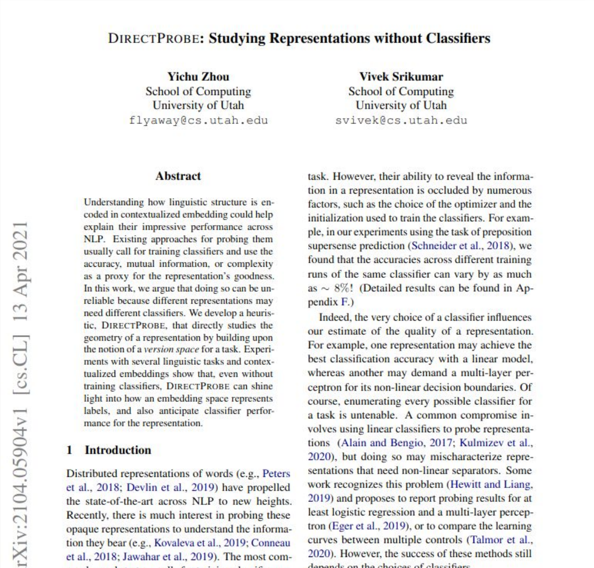
A growing body of work makes use of *probing* to investigate the working of neural models, often considered black boxes. Recently, an ongoing debate emerged surrounding the limitations of the probing paradigm. In this work, we point out the inability to infer behavioral conclusions from probing results and offer an alternative method that focuses on how the information is being used, rather than on what information is encoded. Our method, *Amnesic Probing*, follows the intuition that the utility of a property for a given task can be assessed by measuring the influence of a causal intervention that removes it from the representation. Equipped with this new analysis tool, we can ask questions that were not possible before, e.g. is part-of-speech information important for word prediction? We perform a series of analyses on BERT to answer these types of questions. Our findings demonstrate that conventional probing performance is not correlated to task importance, and we call for increased scrutiny of claims that draw behavioral or causal conclusions from probing results.[1]

Figure 1: A schematic description of the proposed amnesic intervention: we transform the contextualized representation of the word "ran" so as to remove information (here, POS), resulting in a "cleaned" version $h_{ran}^{\neg POS}$. This representation is fed to the word-prediction layer and the behavioral influence of POS erasure is measured.

1 Introduction

What drives a model to perform a specific predic-

being encoded in them. One prominent methodology that attempts to shed light on those questions is probing (Conneau et al., 2018) (also known as *auxilliary prediction* (Adi et al., 2016) and *diag-*



DirectProbe: Studying Representations without Classifiers

Yichu Zhou
School of Computing
University of Utah
flyaway@cs.utah.edu

Vivek Srikumar
School of Computing
University of Utah
svivek@cs.utah.edu

arXiv:2104.05904v1 [cs.CL] 13 Apr 2021

Abstract

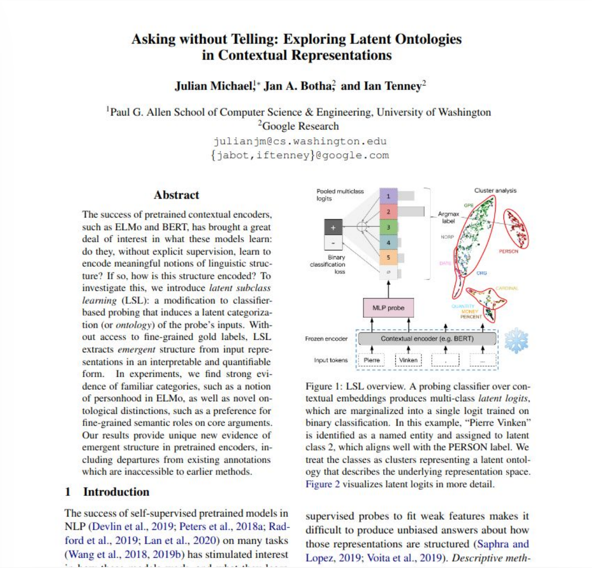
Understanding how linguistic structure is encoded in contextualized embedding could help explain their impressive performance across NLP. Existing approaches for probing them usually call for training classifiers and use the accuracy, mutual information, or complexity as a proxy for the representation's goodness. In this work, we argue that doing so can be unreliable because different representations may need different classifiers. We develop a heuristic, DirectProbe, that directly studies the geometry of a representation by building upon the notion of a *version space* for a task. Experiments with several linguistic tasks and contextualized embeddings show that, even without training classifiers, DirectProbe can shine light into how an embedding space represents labels, and also anticipate classifier performance for the representation.

1 Introduction

Distributed representations of words (e.g., Peters et al., 2018; Devlin et al., 2019) have propelled the state-of-the-art across NLP to new heights. Recently, there is much interest in probing these opaque representations to understand the information they bear (e.g., Kovaleva et al., 2019; Conneau et al., 2018; Jawahar et al., 2019). The most com-

task. However, their ability to reveal the information in a representation is occluded by numerous factors, such as the choice of the optimizer and the initialization used to train the classifiers. For example, in our experiments using the task of preposition supersense prediction (Schneider et al., 2018), we found that the accuracies across different training runs of the same classifier can vary by as much as $\sim 8\%$! (Detailed results can be found in Appendix F.)

Indeed, the very choice of a classifier influences our estimate of the quality of a representation. For example, one representation may achieve the best classification accuracy with a linear model, whereas another may demand a multi-layer perceptron for its non-linear decision boundaries. Of course, enumerating every possible classifier for a task is untenable. A common compromise involves using linear classifiers to probe representations (Alain and Bengio, 2017; Kulmizev et al., 2020), but doing so may mischaracterize representations that need non-linear separators. Some work recognizes this problem (Hewitt and Liang, 2019) and proposes to report probing results for at least logistic regression and a multi-layer perceptron (Eger et al., 2019), or to compare the learning curves between multiple controls (Talmor et al., 2020). However, the success of these methods still



Asking without Telling: Exploring Latent Ontologies in Contextual Representations

Julian Michael[1,*] Jan A. Botha[2] and Ian Tenney[2]

[1]Paul G. Allen School of Computer Science & Engineering, University of Washington
[2]Google Research
julianjm@cs.washington.edu
{jabot,iftenney}@google.com

Abstract

The success of pretrained contextual encoders, such as ELMo and BERT, has brought a great deal of interest in what these models learn: do they, without explicit supervision, learn to encode meaningful notions of linguistic structure? If so, how is this structure encoded? To investigate this, we introduce *latent subclass learning* (LSL): a modification to classifier-based probing that induces a latent categorization (or *ontology*) of the probe's inputs. Without access to fine-grained gold labels, LSL extracts *emergent* structure from input representations in an interpretable and quantifiable form. In experiments, we find strong evidence of familiar categories, such as a notion of personhood in ELMo, as well as novel ontological distinctions, such as a preference for fine-grained semantic roles on core arguments. Our results provide unique new evidence of emergent structure in pretrained encoders, including departures from existing annotations which are inaccessible to earlier methods.

Figure 1: LSL overview. A probing classifier over contextual embeddings produces multi-class *latent logits*, which are marginalized into a single logit trained on binary classification. In this example, "Pierre Vinken" is identified as a named entity and assigned to latent class 2, which aligns well with the PERSON label. We treat the classes as clusters representing a latent ontology that describes the underlying representation space. Figure 2 visualizes latent logits in more detail.

1 Introduction

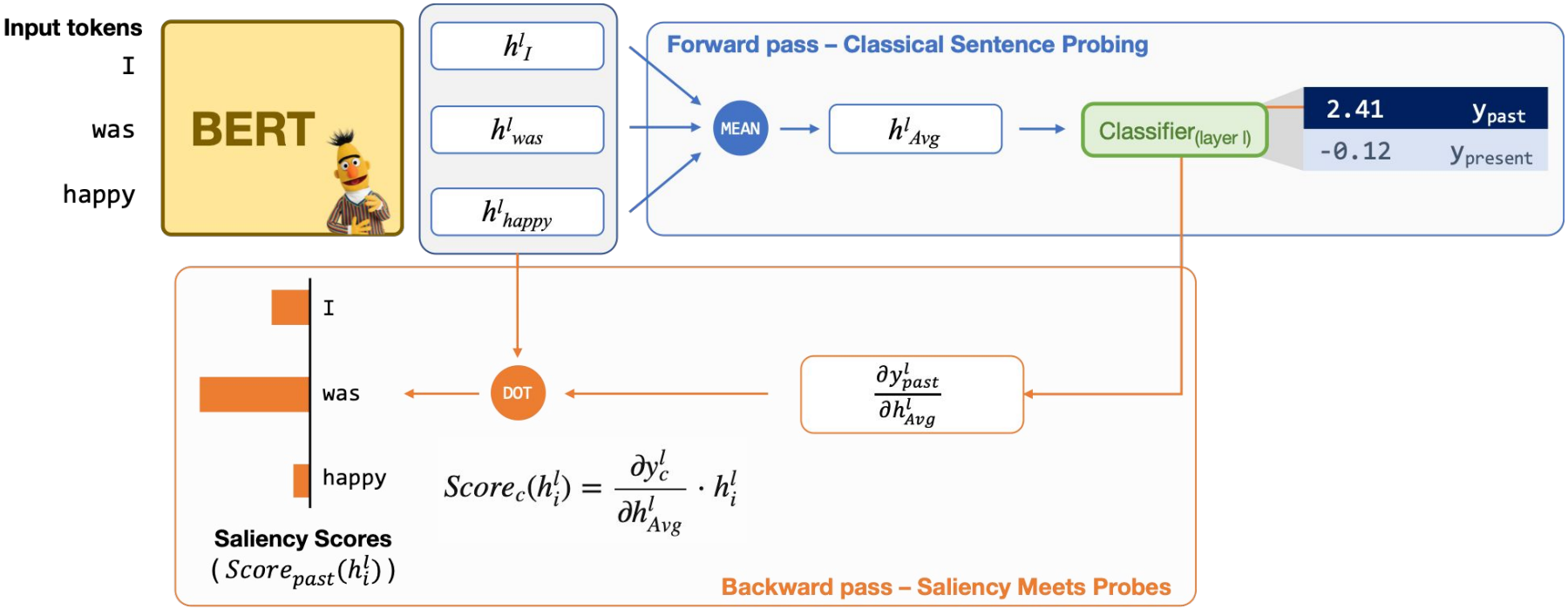
The success of self-supervised pretrained models in NLP (Devlin et al., 2019; Peters et al., 2018a; Radford et al., 2019; Lan et al., 2020) on many tasks (Wang et al., 2018, 2019b) has stimulated interest

supervised probes to fit weak features makes it difficult to produce unbiased answers about how those representations are structured (Saphra and Lopez, 2019; Voita et al., 2019). *Descriptive meth-*

Discussion Part (2)

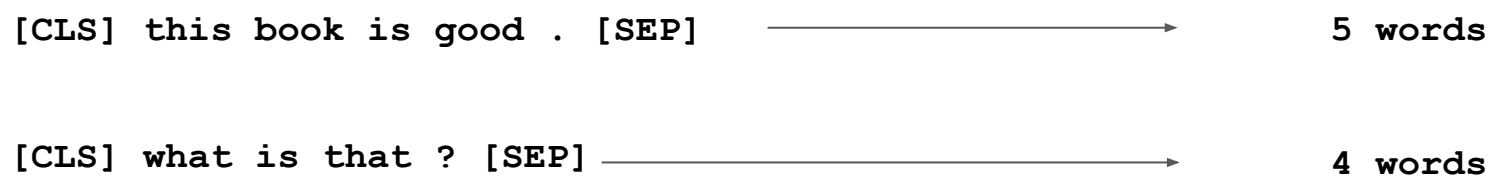
Exploring the Role of BERT Token Representations to Explain Sentence Probing Results

Saliency Meets Probes



Probing Explanation

Sentence Length

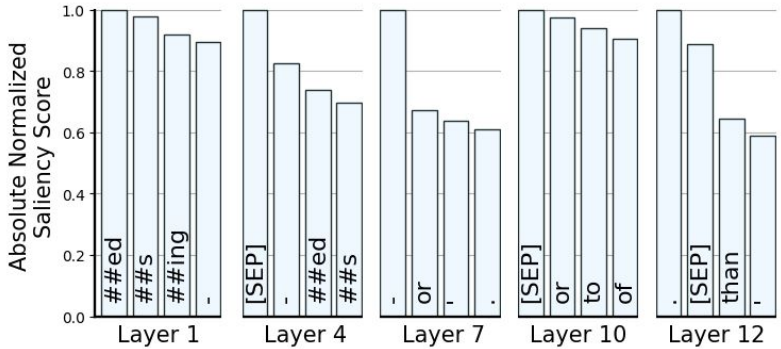


Probing Explanation

Sentence Length

`[CLS] this book is good . [SEP]` → 5 words

`[CLS] what is that ? [SEP]` → 4 words



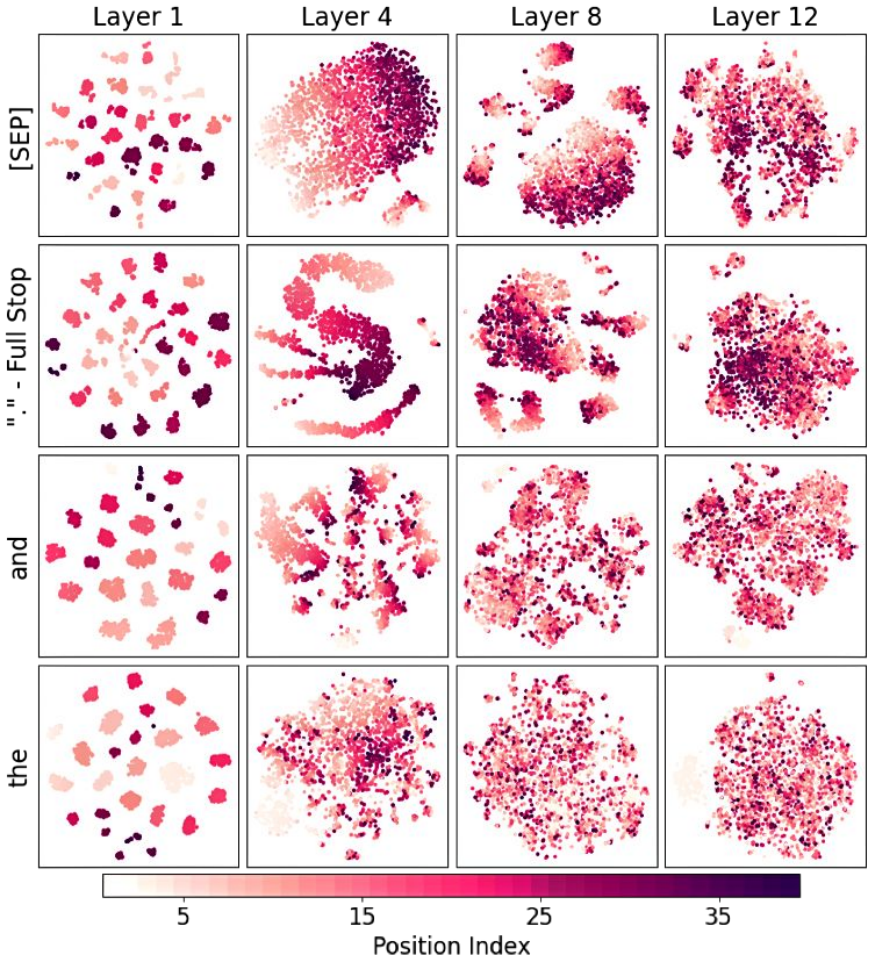
Layer	SentLen
1	0.91
2	0.89
3	0.89
4	0.85
5	0.81
6	0.78
7	0.70
8	0.68
9	0.59
10	0.49
11	0.43
12	0.44

Probing Explanation

Sentence Length

Sentence ending tokens

retain positional information.



Probing Explanation

Obj. Number / Verb Tense

ObjNum:

I wasn't chasing rainbows but perhaps she had spotted my struggle . --> NNS

Tense:

In her view , reading the bible fixes everything . --> PRESENT

Tense	ObjNum
0.87	0.82
0.88	0.83
0.88	0.84
0.89	0.85
0.89	0.86
0.89	0.86
0.89	0.86
0.89	0.85
0.89	0.85
0.89	0.84
0.89	0.83
0.89	0.83

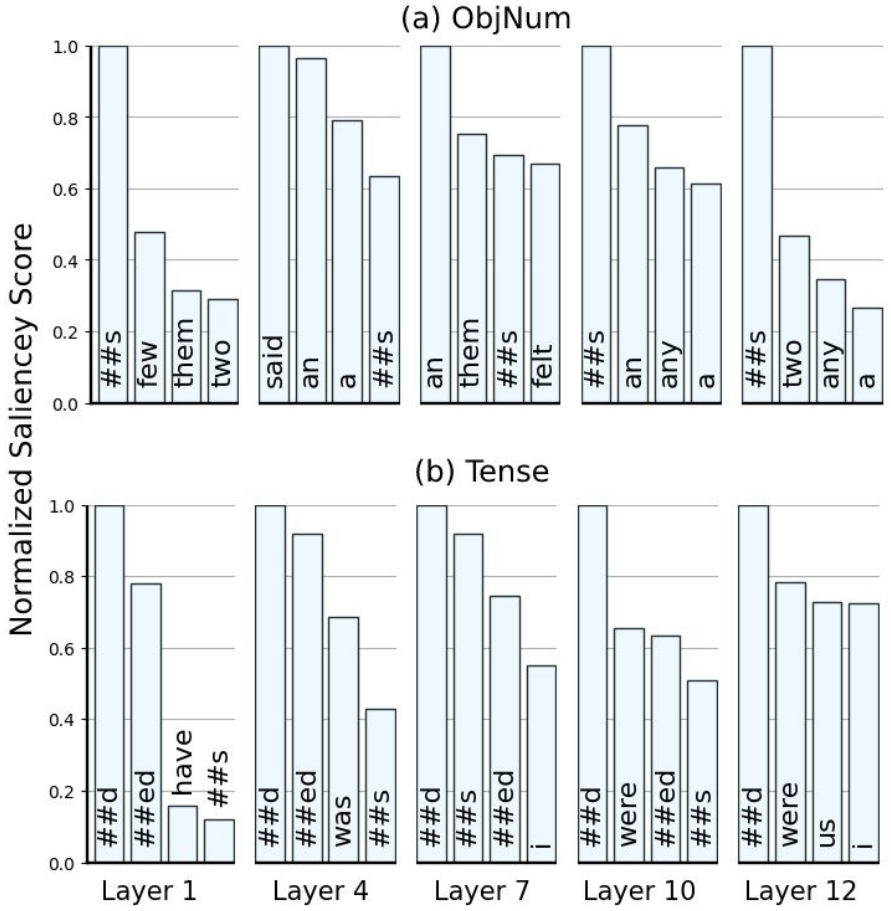
Probing Explanation

Verb Tense / Obj. Number

Articles and ending tokens

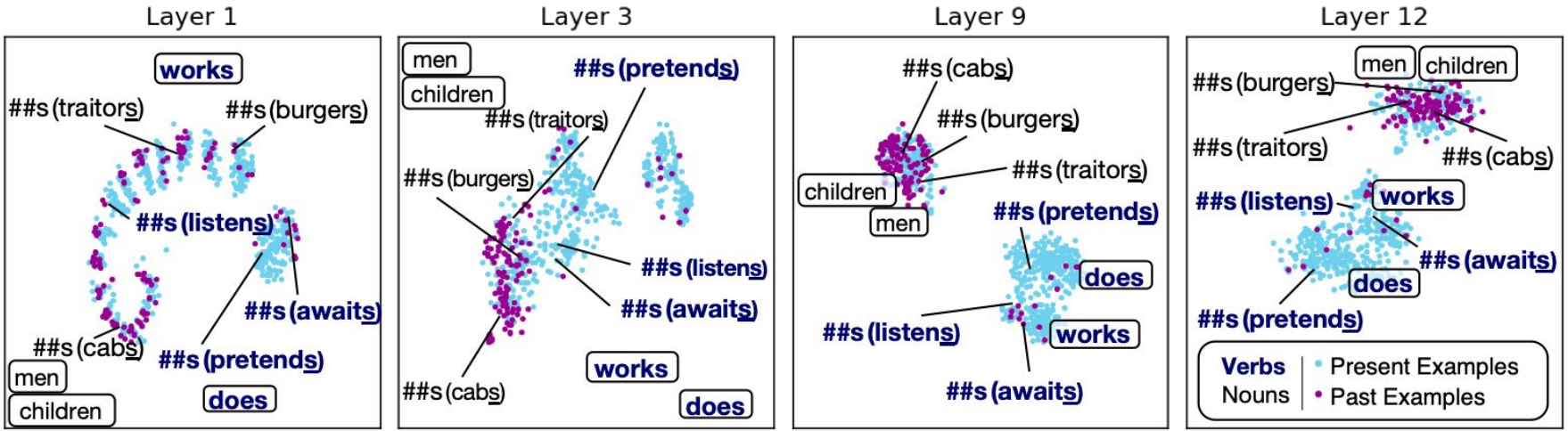
(e.g., ##s and ##ed)

are key playmakers.



Probing Explanation

Verb Tense / Obj. Number

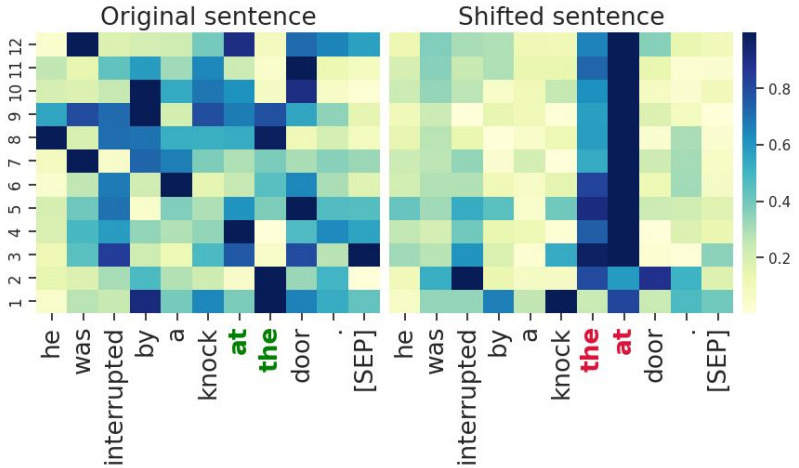


##s – Plural or Present?

Probing Explanation

Bi-gram Shift -- Word-level Abnormality

He was interrupted by a knock at the door . --> Original
I felt like time my here was getting short . --> Inverted

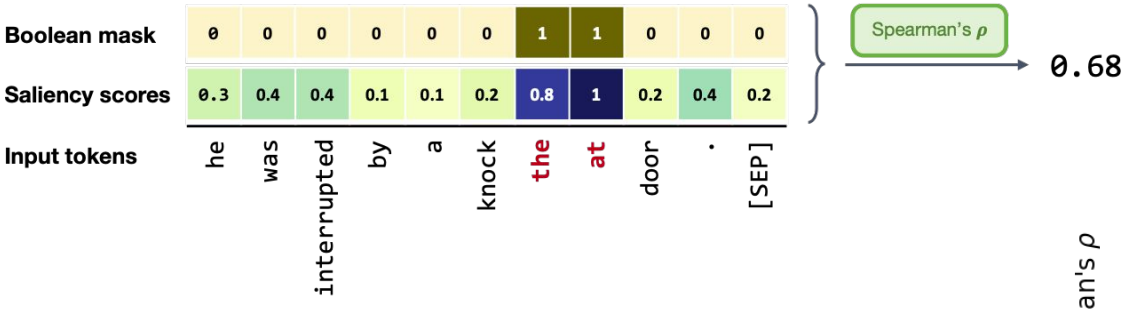


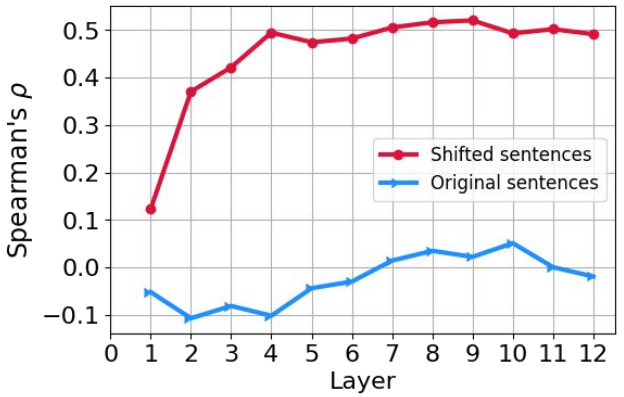
BShift

0.57
0.72
0.84
0.87
0.88
0.90
0.91
0.91
0.91
0.91
0.90
0.89

Probing Explanation

Bi-gram Shift -- Word-level Abnormality

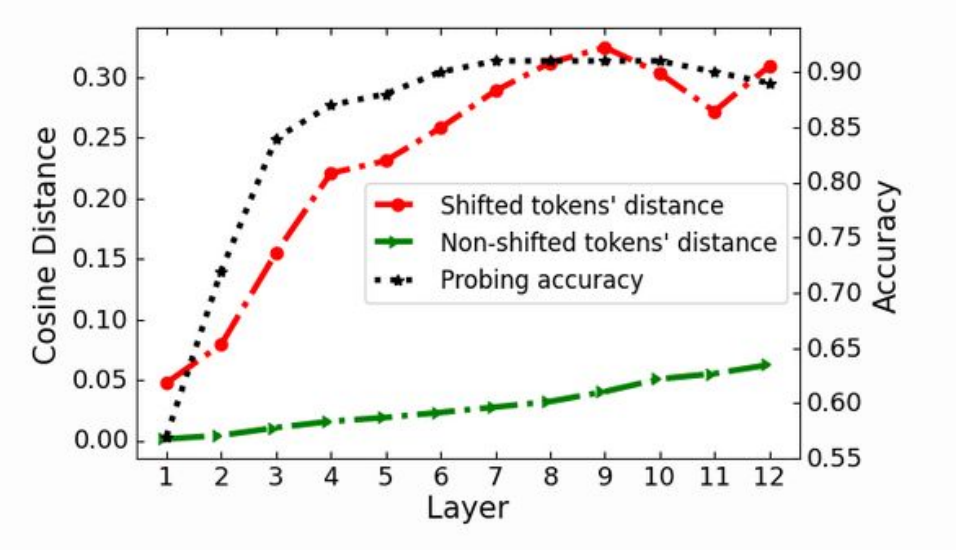




Probing Explanation

Bi-gram Shift -- Word-level Abnormality

BERT encoding abnormalities in the **shifted tokens**.



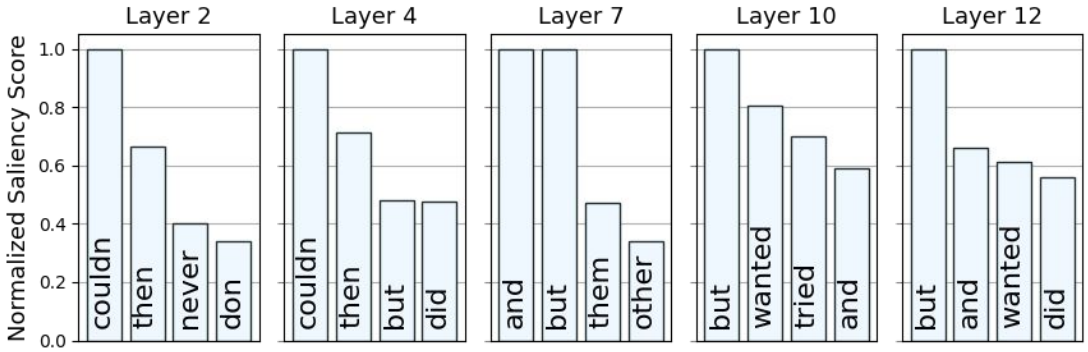
Probing Explanation

Coordination Inversion -- Phrasal-level Abnormality

There was something to consider but he might be a prince . --> Inverted

I cut myself and the glass broke . --> Inverted

Both sentences would be correct if we just swap the blue and the purple clauses.



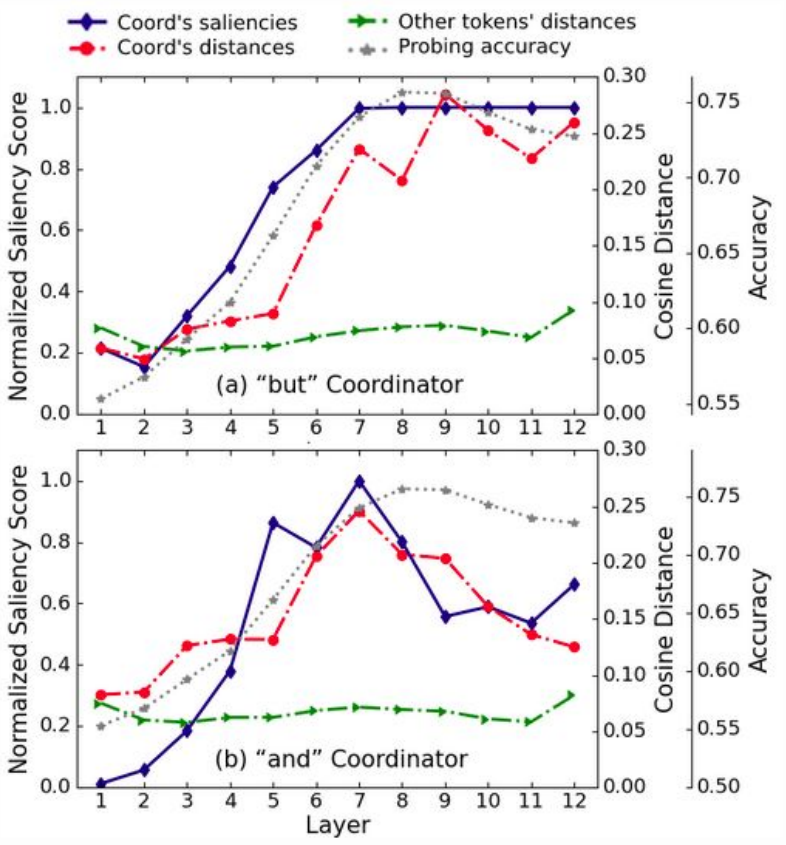
CoordInv

0.55
0.57
0.60
0.62
0.66
0.71
0.74
0.75
0.76
0.74
0.73
0.72

Probing Explanation

Coordination Inversion -- Phrasal-level Abnormality

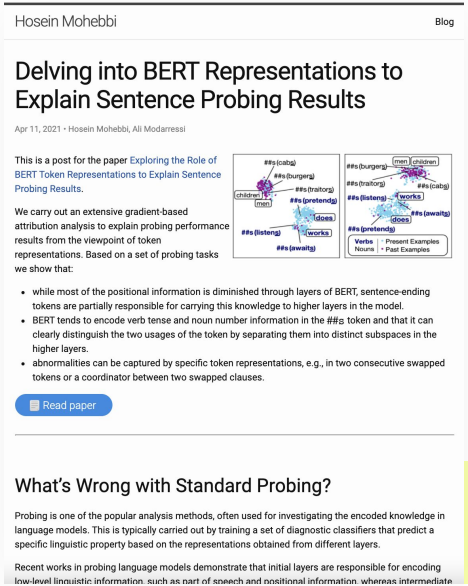
This observation implies that BERT somehow **encodes oddity** in the **coordinator** representations.



Discussion Part (3)

For further information...

Paper:
https://arxiv.org/pdf/2104.01477.pdf



Hosein Mohebbi Blog

Delving into BERT Representations to Explain Sentence Probing Results

Apr 11, 2021 • Hosein Mohebbi, Ali Modarressi

This is a post for the paper Exploring the Role of BERT Token Representations to Explain Sentence Probing Results.

We carry out an extensive gradient-based attribution analysis to explain probing performance results from the viewpoint of token representations. Based on a set of probing tasks we show that:

- while most of the positional information is diminished through layers of BERT, sentence-ending tokens are partially responsible for carrying this knowledge to higher layers in the model.
- BERT tends to encode verb tense and noun number information in the ##s token and that it can clearly distinguish the two usages of the token by separating them into distinct subspaces in the higher layers.
- abnormalities can be captured by specific token representations, e.g., in two consecutive swapped tokens or a coordinator between two swapped clauses.

Read paper

What's Wrong with Standard Probing?

Probing is one of the popular analysis methods, often used for investigating the encoded knowledge in language models. This is typically carried out by training a set of diagnostic classifiers that predict a specific linguistic property based on the representations obtained from different layers.

Recent works in probing language models demonstrate that initial layers are responsible for encoding low-level linguistic information, such as part of speech and positional information, whereas intermediate

Blog:
https://hmohebbi.github.io/blog/explain-probing-results

Exploring the Role of BERT Token Representations to Explain Sentence Probing Results

Hosein Mohebbi[*♡] **Ali Modarressi**[*♡] **Mohammad Taher Pilehvar**[♠]
[♡] Iran University of Science and Technology
[♠] Tehran Institute for Advanced Studies, Iran
[♡]{hosein_mohebbi, m_modarressi}@comp.iust.ac.ir
[♠]mp792@cam.ac.uk

Abstract

Several studies have been carried out on revealing linguistic features captured by BERT. This is usually achieved by training a diagnostic classifier on the representations obtained from different layers of BERT. The subsequent classification accuracy is then interpreted as the ability of the model in encoding the corresponding linguistic property. Despite providing insights, these studies have left out the potential role of token representations. In this paper, we provide an analysis on the representation space of BERT in search for distinct and meaningful subspaces that can explain probing results. Based on a set of probing tasks and with the help of attribution methods we show that BERT tends to encode meaningful knowledge in specific token representations (which are often ignored in standard classification setups), allowing the model to detect syntactic and semantic abnormalities, and to distinctively separate grammatical number and tense subspaces.

1 Introduction

Recent years have seen a surge of interest in pre-trained language models, highlighted by extensive research around BERT (Devlin et al., 2019) and

specific linguistic property based on the representations obtained from different layers. Recent works in probing language models demonstrate that initial layers are responsible for encoding low-level linguistic information, such as part of speech and positional information, whereas intermediate layers are better at syntactic phenomena, such as syntactic tree depth or subject-verb agreement, while in general semantic information is spread across the entire model (Lin et al., 2019; Peters et al., 2018; Liu et al., 2019a; Hewitt and Manning, 2019; Tenney et al., 2019). Despite elucidating the type of knowledge encoded in various layers, these studies do not go further to investigate the reasons behind the layer-wise behavior and the role played by token representations. Analyzing the shortcomings of pre-trained language models requires a scrutiny beyond the mere performance in a given probing task. This is particularly important as recent studies suggest that the final classifier (applied to model's outputs) might itself play a significant role in learning nuances of the task (Hewitt and Liang, 2019; Voita and Titov, 2020; Pimentel et al., 2020).

We extend the layer-wise analysis to the token level in search for distinct and meaningful subspaces in BERT's representation space that can

References (1)

- Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, pages 276–286, Florence, Italy. Association for Computational Linguistics.
- Samira Abnar and Willem Zuidema. 2020. Quantifying attention flow in transformers. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4190–4197, Online. Association for Computational Linguistics.
- Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. 2020. Attention is not only a weight: Analyzing transformers with vector norms. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 7057–7075, Online. Association for Computational Linguistics.
- J Alammar. 2020. Interfaces for explaining transformer language models.
- 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.
- Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

References (2)

- Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.
- Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R Bowman, Dipanjan Das, et al. 2018. What do you learn from context? probing for sentence structure in contextualized word representations. In International Conference on Learning Representations.
- Lucas Torroba Hennigen, Adina Williams, and Ryan Cotterell. 2020. Intrinsic probing through dimension selection. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 197–216, Online. 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.
- Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2020. oLMpics-on what language model pre-training captures. Transactions of the Association for Computational Linguistics, 8:743–758.
- Elena Voita and Ivan Titov. 2020. Information theoretic probing with minimum description length. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 183–196, Online. Association for Computational Linguistics.

References (3)

- Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. 2021. Amnesic probing: Behavioral explanation with amnesic counterfactuals. Transactions of the Association for Computational Linguistics, 9:160–175.
- Yichu Zhou and Vivek Srikumar. 2021. Directprobe: Studying representations without classifiers. arXiv preprint arXiv:2104.05904.
- Julian Michael, Jan A. Botha, and Ian Tenney. 2020. Asking without telling: Exploring latent ontologies in contextual representations. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 6792–6812, Online. Association for Computational Linguistics.
- Hosein Mohebbi, Ali Modarressi, and Mohammad Taher Pilehvar. 2021. Exploring the role of bert token representations to explain sentence probing results. arXiv preprint arXiv:2104.01477.