Probing Linguistic Information
For Logical Inference In Pre-trained Language Models

Zeming Chen   **Qiyue Gao**

Rose-Hulman Institute of Technology
{chenz16, gaoq}@rose-hulman.edu

Abstract

Progress in pre-trained language models has led to a surge of impressive results on downstream tasks for natural language understanding. Recent work on probing pre-trained language models uncovered a wide range of linguistic properties encoded in their contextualized representations. However, it is unclear whether they encode semantic knowledge that is crucial to symbolic inference methods. We propose a methodology for probing linguistic information for logical inference in pre-trained language model representations. Our probing datasets cover a list of linguistic phenomena required by major symbolic inference systems. We find that (i) pre-trained language models do encode several types of linguistic information for inference, but there are also some types of information that are weakly encoded, (ii) language models can effectively learn missing linguistic information through fine-tuning. Overall, our findings provide insights into which aspects of linguistic information for logical inference do language models and their pre-training procedures capture. Moreover, we have demonstrated language models' potential as semantic and background knowledge bases for supporting symbolic inference methods.

Introduction

Pre-trained language models have replaced traditional symbolic-based natural language processing systems on a variety of language understanding tasks, mainly because symbolic-based NLP systems often rely on linguistic properties as features. Those features are hard to acquire. Many types of linguistic information are either hand-written rules or background knowledge extracted from traditional knowledge base, which make symbolic-based systems hard to scale up on large benchmarks such as GLUE (Wang et al. 2018). On the other hand, many recent probing studies have revealed that sentence representations of pre-trained language models encode a large amount of linguistic information and background knowledge (Tenney et al. 2019; Petroni et al. 2019; Bouraoui, Camacho-Collados, and Schockaert 2020). However, it remains unknown if these representations also encode implicit linguistic information for inference crucial to symbolic inference systems.
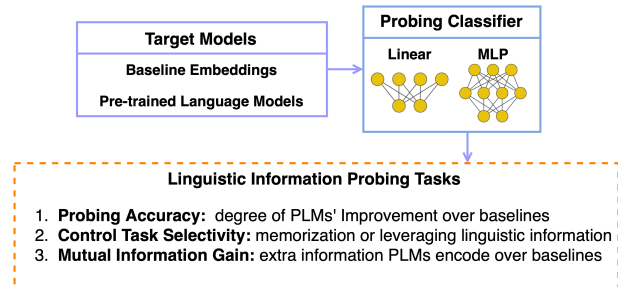
Figure 1: Given pre-trained language models, the probing classifier extracts linguistic infromation for a given probing task. The amount of intimation is measured by the probing accuracy and the information gain, compared with baselines.

In this paper, we propose an inference information probing framework (Figure 1). We define a set of probing tasks that focus on different types of linguistic information required by symbolic systems. In particular, we cover linguistic information for simple and complex semantic phenomena. Simple semantic phenomena often relies on partial or no context and does not require advanced linguistic skills like contextual understanding and reasoning. Our simple phenomena include word-to-word semantic relations, lexical semantics, and contradiction signatures. Complex phenomena depends on multiple types of reasoning skills like reasoning on event context, monotonicity, coreference, and commonsense knowledge. For complex phenomena, we probe sentiment, relational knowledge, anaphora resolution, and monotonicity reasoning. We are interested in answering two questions: (1) Do pre-trained language models encode linguistic information essential to symbolic inference systems? (2) Do pre-trained language models acquire new linguistic information for inference during the fine-tuning process for the NLI task? For each task, we conducted probing experiments on multiple contextualized language models and compared results to several strong baselines.

Our analysis shows that language models encode diverse types of linguistic information for inference. In particular, they encode more information on simple semantic phenomena than complex semantic phenomena. Our label-wise qualitative analysis revealed that the amount of information encoded by language models for each task is different across

labels which justifies our previous findings. Moreover, we found that pre-trained language models can obtain some types of the missing linguistic information through fine-tuning for the NLI task. Overall, our findings show that pre-trained language models can be potential linguistic knowledge bases supporting symbolic inference systems.

Contributions Our contributions are as follows:

1. Our work expands on prior probing studies by studying a wider range of linguistic information, including simple and complex semantic phenomena.

2. Our experiments allow classifier expressiveness to be analyzed in a more complex setting covering syntactic and semantic linguistic properties beyond prior works.

3. Our study provides insights into what types of new linguistic information pre-trained language models obtain during fine-tuning on large NLI datasets. This contributes to the interpretability of NLI models.

Related Work

Recent studies have reported the existence of linguistic properties encoded in the self-attention weights of language models' contextualized representations. These linguistic properties include syntactic structure, semantic knowledge, and some world knowledge (Rogers, Kovaleva, and Rumshisky 2020). Several studies train and evaluate a probing classifier on top of different language models' contextualized representations to explore the existence of information about linguistic properties. These studies have shown that pre-trained language models encode some levels of syntactic and semantic knowledge. Hewitt and Manning (2019) recovered syntactic dependencies from BERT's embeddings by learning transformation matrices. Tenney et al. (2019), which is more directly related to our work, proposed the edge probing framework and found that contextualized embeddings encode information about named entity types, relations, semantic roles and proto roles based on the high accuracy of the probing classifier.

Some probing studies focus on inducing factual knowledge captured in pre-trained language models. A majority of the studies rely on the Masked Language Modeling (MLM) component of the model which can be adapted to induce knowledge easily, since the model only needs to fill in the blanks. Petroni et al. (2019) showed that pre-trained BERT encodes relational knowledge competitive with those that are accessed from knowledge bases using traditional NLP methods. They also found that BERT has strong ability to recall factual knowledge prior to any fine-tuning, making it a good candidate for open-domain QA systems (unsupervised). Bouraoui, Camacho-Collados, and Schockaert (2020) proposed a method to induce relations from pretrained language models. They first found potential sentences that express a relation in a large corpus. A subset of sentences was used as templates. They then fine-tuned a language model to predict whether a given word pair form some relation. They found strong evidence that relations can be obtained from language models.

Compared to existing work, we extend common syntactic and semantic tasks to a range of tasks that focus on more complex linguistic phenomenons. Some of our tasks, such as semantic graph construction and monotonicity polarization, require both syntactic and semantic information. Probing for more complex linguistic tasks allows us to diagnose the particular advantages of language models over conventional NLP systems. It also allows us to study the expressiveness of probing classifiers in a more complex setting beyond syntactic tasks. Moreover, our experiments on fine-tuned NLI language models provides insights into the type of linguistic information they capture through fine-tuning.

Probing Methodology

Edge Probing and Vertex Probing

Edge probing is a simple and useful probing framework proposed by Tenney et al. (2019). It can provide a uniform set of metrics and architectures across diverse task types. Formally, a sentence is defined as a list of tokens $[t_0, t_1, t_2, ..., t_n]$ and an edge target as $\{s_1, s_2, \mathcal{L}\}$ where s_1 and s_2 are two end exclusive spans with $s_1 = \{i^{s1}, j^{s1})$ and $s_2 = \{i^{s2}, j^{s2})$. $\mathcal{L}$ is a label assigned to the pair of spans which the classifier needs to accurately predict. The label set for $\mathcal{L}$ is different across tasks including both binary labels and multi labels. Each sentence $[t_0, t_1, t_2, ..., t_n]$ is encoded by a language model into contextualized sentence representation $[e_0, e_1, e_2, ..., e_n]$. A projection layer concatenated with a self-attention pooling operator will be applied to the representation to extract span representations according to the index position of two spans $s_1 = \{i^{s1}, j^{s1})$ and $s_2 = \{i^{s2}, j^{s2})$. As Tenney et al. (2019) mentioned, the pooling is fixed-length and only operates within the bounds of a span to ensure that the classifier can only access information of the rest of the sentence from the contextual sentence representation. The two span representations are concatenated and passed to the classifier to predict the label. To ensure we only probe a pre-trained language model without modifying its parameters, we freeze its parameters to not allow for gradient updates.

The vertex probing framework has the same settings and formulations as the edge probing framework, except that vertex probing operates on every token in a sentence. The classifier receives only a single span representation as input. Formally, the definition is very similar to that of the sequence tagging task. With a list of tokens $[t_0, t_1, t_2, ..., t_n]$, we define each token as a single span target $s = \{(i^s, j^s), \mathcal{L}\}$. The vertex probing is used to predict which words belong to a category in the label set.

Classifier Selection

Selecting a good probing classifier is essential to the probing process. We first choose the linear classifier. According to Hewitt and Liang (2019), the linear classifier is less expressive and thus is prevented from memorizing the task. However, Pimentel et al. (2020) uses probing to estimate the mutual information between a representation-valued and a linguistic-property–valued random variable. They argue that the most optimal probe should be used to minimize the chance of misinterpreting a representation's encoded information, and therefore achieve the optimal estimate of mu-

Task	Probe Type	Example	Split	# S
Semantic-Graph (SemGraph)	edge probing	A [tall] [boy] is [running] [quickly] to catch a [soccer] [ball] . [boy] [running] $\longrightarrow$ **Concept-Relation**; [tall] [boy] $\longrightarrow$ **Concept-Modifier**; [quickly] [running] $\longrightarrow$ **Relation-Modifier**	train test	10000 5000
Monotonicity	vertex probing	Some$^\uparrow$ people$^\uparrow$ in$^\uparrow$ the$^\uparrow$ White$^=$ House$^=$ does$^\uparrow$ not$^\uparrow$ know$^\downarrow$ if$^\downarrow$ any$^\downarrow$ dog$^\downarrow$ in$^\downarrow$ Ohio$^\downarrow$ ate$^\downarrow$ bananas$^\downarrow$ yesterday$^\downarrow$	train test	5000 500
Lexical (SA-Lex)	edge probing	P: The $[\text{man}]_{s^1}$ is holding a $[\text{saxophone}]_{s^2}$ H: The man is holding an $[\text{instrument}]_{s^3}$. $(s_1, s_3) \longrightarrow$ **Unaligned**; $(s_2, s_3) \longrightarrow$ **Aligned**	train test	1000 500
Anaphora (SA-AP)	edge probing	The $[\text{technician}]_{s^1}$ told the $[\text{customer}]_{s^2}$ that $[\text{he}]_{s^3}$ could pay with cash. $(s_1, s_3) \longrightarrow$ **Unaligned**; $(s_2, s_3) \longrightarrow$ **Aligned**	train test	500 220
Sentiment (SA-ST)	vertex probing	P: When asked about the restaurant, Brielle said, " $[\text{It}]_{t^1}$ $[\text{was}]_{t^2}$ $[\text{terrible!}]_{t^3}$ $[\text{I}]_{t^4}$ $[\text{found}]_{t^5}$ $[\text{this}]_{t^6}$ $[\text{product}]_{t^7}$ $[\text{to}]_{t^8}$ $[\text{be}]_{t^9}$ $[\text{way}]_{t^{10}}$ $[\text{too}]_{t^{11}}$ $[\text{big}]_{t^{12}}$ " H: Brielle $[\text{did}]_{t^{13}}$ $[\text{not}]_{t^{14}}$ $[\text{like}]_{t^{15}}$ $[\text{the}]_{t^{16}}$ $[\text{restaurant}]_{t^{17}}$ $\{t^1, \ldots, t^3\} \longrightarrow$ **Align$_1$**; $\{t^4, \ldots, t^{12}\} \longrightarrow$ **Unaligned**; $\{t^{13}, \ldots, t^{17}\} \longrightarrow$ **Align$_2$**	train test	1000 600
Relational-Knowledge (SA-RK)	vertex probing	P: $[\text{Dirk}]_{t^1}$ $[\text{Nowitski}]_{t^2}$ $[\text{is}]_{t^3}$ $[\text{a}]_{t^4}$ $[\text{current}]_{t^5}$ $[\text{NBA}]_{t^6}$ $[\text{star}]_{t^7}$ playing with the $[\text{Dallas}]_{t^8}$ $[\text{Mavericks}]_{t^9}$ $[\text{as}]_{t^{10}}$ $[\text{an}]_{t^{11}}$ $[\text{all-purpose}]_{t^{12}}$ $[\text{forward}]_{t^{13}}$ H: $[\text{Dirk}]_t^{14}$ $[\text{Nowitski}]_t^{15}$ $[\text{plays}]_t^{16}$ $[\text{in}]_t^{17}$ $[\text{the}]_t^{18}$ $[\text{NBA}]_t^{19}$ $\{t^1, \ldots, t^7\} \longrightarrow$ **Align$_1$**; $\{t^8, \ldots, t^{13}\} \longrightarrow$ **Unaligned**; $\{t^{14}, \ldots, t^{19}\} \longrightarrow$ **Align$_2$**	train test	1000 500
Contradiction Signature (ContraSig)	vertex probing	P: Italy and Germany $[\text{have}]_{t^1}$ $[\text{each}]_{t^2}$ $[\text{played}]_{t^3}$ $[\text{twice}]_{t^4}$, and they $[\text{haven't}]_{t^5}$ $[\text{beaten}]_{t^6}$ $[\text{anybody}]_{t^7}$ $[\text{yet}]_{t^8}$ H: Italy $[\text{defeats}]_{t^9}$ $[\text{Germany}]_{t^{10}}$ $\{t^1, \ldots, t^4\} \longrightarrow$ **None$_1$**; $\{t^5, \ldots, t^8\} \longrightarrow$ **Contra-sig$_1$**; $\{t^9, t^{10}\} \longrightarrow$ **Contra-sig$_2$**	train test	1000 500

Figure 2: Here we list examples of our probing tasks. For semantic graph construction task, the red, green and blue boxes denote modifiers, concepts and relations respectively. For semantic alignment tasks and the contradiction signature detection task, the red boxes are spans that are semantically aligned. The green boxes are spans that form a contradiction signature. The blue boxes are spans that are irrelevant to semantic alignment or contradiction. Here P stands for a premise, and H stands for a hypothesis. The anaphora-based alignment only uses a single sentence. For the labels, $(s^1, s^3) \longrightarrow$ Aligned means that s^1 and s^3 are aligned. $\{t^1, \ldots, t^7\} \longrightarrow$ Align$_1$ means that tokens t^1 to t^7 belongs to the first phrase in a semantically aligned pair.

tual information. To lessen the chance of misinterpretation, we conducted probing using a Multi-layer Perceptron (MLP) classifier with one hidden layer.

Experiment Setup

To answer both questions 1 and 2 in the introduction, we experiment with five pre-trained language models. We selected BERT-base and BERT-large (Devlin et al. 2019), RoBERTa-base and RoBERTa-large (Liu et al. 2019), and DeBERTa (He et al. 2021). All five models can provide contextualized sentence representations and have shown impressive performance on the GLUE (Wang et al. 2018) benchmark. Our experiment setup follows three types of evaluation methods to interpret the probing performance.

Probing Accuracy We probe pre-trained language models and the baseline word embeddings using linear and MLP classifiers. Then, we compare their performance to determine if the pre-trained models improve over the baselines.

If such improvement is significant, the pre-trained models contain more information about a task than the baseline. Otherwise, they do not contain enough information to benefit a task. We select four uncontextualized word embeddings as our baselines, including random embedding, Fast-Text (Joulin et al. 2017), Glove (Pennington, Socher, and Manning 2014), and Word2Vec (Mikolov et al. 2013). We also conduct a label-wise qualitative analysis for each task to explore if the amount of information is encoded differently across the task-specific labels. Finally, to determine if language models can learn the missing linguistic information for inference, we evaluate NLI models fine-tuned on MultiNLI (Williams, Nangia, and Bowman 2018), using probing tasks that do not benefit from the pre-trained models.

Control Task Hewitt and Liang (2019) argue that accuracy cannot fully validate that a representation encodes linguistic information since a highly expressive classifier could

have memorized the task. They proposed the use of control tasks to complement probings. The main idea is to validate if a classifier could predict task outputs independently of a representation's linguistic properties. If the classifier can achieve high accuracy in this setting, the accuracy does not necessarily reflect the properties of the representation. A control task has the same input as the associated linguistic task, but it assigns random labels to the input data. The selectivity, which is the difference between the linguistic task accuracy and the control task accuracy, is used to measure the quality of a probe. A good probe should have high selectivity meaning that it has low control task accuracy but high linguistic task accuracy.

Information-theoretic Probing   Pimentel et al. (2020) argue that the task of supervised probing is an attempt to measure how much information a neural representation can provide for a task. They operationalized probing as estimating the mutual information between a representation and a probing task. The mutual information from a target representation is compared to the information estimated from a control function's representation, which serves as a baseline for comparison. In our experiments, we use uncontextualized baselines as control functions. We estimate the information gain between a contextualized embedding and a baseline. Information gain measures how much more information about a task a contextualized embedding has over a baseline. In addition, we transform the gain into a percentage measurement to make the results more interpretable.

Inference Information Probes

In this section, we introduce a list of edge and vertex probing tasks for probing implicit linguistic information for symbolic inference methods in pre-trained language model representations. To discover potential tasks that can provide essential linguistic information for symbolic inferences, we studied four major logical systems for NLI, all with high accuracy on SICK (Marelli et al. 2014), and several challenge datasets for the NLI task. They include NLI systems based on natural logic (Abzianidze 2020), monotonicity reasoning (Hu et al. 2020; Chen, Gao, and Moss 2021), and theorem proving (Yanaka et al. 2018).

Semantic Graph Construction (SemGraph)

This task probes the graph-based abstract meaning representation for sentences, a type of knowledge found effective in symbolic systems for acquiring paraphrase pairs and selecting correct inference steps (Yanaka et al. 2018; Chen, Gao, and Moss 2021). The task is to construct a semantic graph that captures connections between concepts, modifiers, and relations in a sentence. Relations are words that form a connection, including verbs and prepositions. Concepts are arguments connected by a relation such as objects and subjects. Each concept connects to a set of modifiers that attribute to it. An example semantic graph is shown in Table 2. We define this as an edge probing task and assign a label to a pair of tokens. A label is selected from the label set: concept-to-relation, concept-to-modifier, relation-to-concept, relation-to-modifier, relation-to-relation, modifier-to-relation, modifier-to-concept. To construct the dataset, we use dependency parsing and semantic role labeling tools to identify concepts, modifiers, and relations in a sentence and the connection between them. We selected premises from the SNLI test set as our inputs and split them into training and testing sets.

Semantic Alignment (SA)

This set of tasks probes the linguistic information for inference involving semantically aligned phrase or word pairs. These aligned pairs can often serve as an explanation of the entailment gold-label (Abzianidze 2020; Chen, Gao, and Moss 2021). We cover a wide range of semantic phenomena common in natural language inference including lexical (SA-Lex), anaphora (SA-AP), sentiment (SA-ST), and relational knowledge (SA-RK). Table 2 lists each type of semantic alignment with associated examples. Probing data are first collected from multiple challenge datasets for NLU and then are manually annotated for the edge and vertex probing framework. For the sentiment task, we noticed that the aligned phrases are always part of a person's saying, leading a model to solve the task quickly by memorization. To avoid this, we concatenated each premise with speech fragments from another randomly selected premise to build a more complex premise. For instance, in the example in Table 2, *I found this product to be way too big* is a speech fragment from another premise sample.

We formulate each task as either an edge probing or a vertex probing task during annotation. For edge probing tasks, we assign either **Aligned** or **Unaligned** to a pair of spans. For example, in the Lexical example in Table 2, (s^2: [*saxophone*], s^3: [*instrument*]) are aligned, and (s^1: [*man*], s^3: [*instrument*]) are unaligned. In a vertex probing task, we label a token as either **Aligned**$_1$ (the token belongs to the first phrase of the aligned pair), **Aligned**$_2$ (the token belongs to the second phrase of the aligned pair), or **Unaligned** (the token is not in any aligned phrases). For example, in the relational-knowledge example in Table 2, {*Dirk*, *Nowitski*, *is*, *a*, *current*, *NBA*, *star*} are tokens in the first phrase of the aligned pair, {*Dirk*, *Nowitski*, *plays*, *in*, *the*, *NBA*} are tokens in the second phrase of the aligned pair, and {*Dallas*, *Mavericks*, *as*, *an*, *all-purpose*, *forward*} are unaligned tokens. We apply edge probing to tasks with single word spans (lexical and anaphora) and vertex probing to tasks involving multi-word spans (sentiment and relational knowledge). In general, vertex probing adds more distractors into the task to increase the difficulty level, ensuring that the models are using the correct types of reasoning when making a decision.

Contradiction Signature (ContraSig)

Being able to reason about contradictions between a pair of sentences is a fundamental requirement of Natural Language Inference. To determine a contradictory relationship, systems often rely on contradiction signatures, or possible rationales of contradiction in the sentence pair. Contradiction signature detection tests for both syntax and fundamental semantics. We define this task as vertex probing and manually annotated one dataset for detecting contradiction in text (Marelli et al. 2014) by labeling tokens in the first phrase of

Model	SemGraph	ContraSig	Monot	SA-Lex	SA-ST	SA-AP	SA-RK
Group 1: **Baselines** (direct probing, no fine-tuning)							
Random	68.5	29.7	48.8	45.5	46.8	48.9	45.7
Word2Vec	68.8	42.4	43.4	59.7	31.4	32.3	42.3
Glove	71.3	40.2	41.3	60.6	33.4	35.8	50.5
FastText	69.4	31.7	51.9	50.1	51.3	51.3	52.2
Group 2: **Language Models** (**Linear** classifier probing, no fine-tuning; **selectivity** is shown in parenthesis)							
BERT-base	91.8 (42.0)	58.5 (35.1)	48.5 (35.8)	57.4 (7.30)	56.3 (34.7)	49.8 (0.1)	62.8 (50.3)
BERT-large	88.9 (39.5)	51.6 (37.9)	44.9 (37.4)	55.5 (4.80)	51.2 (33.1)	50.2 (0.1)	62.1 (49.5)
RoBERTa-base	88.9 (39.2)	52.2 (33.0)	42.8 (42.3)	61.4 (11.3)	47.5 (38.6)	49.4 (0.6)	62.3 (49.7)
RoBERTa-large	89.6 (40.0)	48.4 (27.6)	38.1 (43.0)	66.2 (16.2)	46.1 (36.6)	49.8 (0.3)	62.9 (50.1)
DeBERTa	**93.4** (43.7)	78.5 (32.3)	54.9 (42.2)	83.8 (33.2)	42.8 (35.7)	51.8 (2.3)	65.6 (52.8)
Group 3: **Language Models** (**MLP** classifier probing, no fine-tuning; **selectivity** is shown in parenthesis)							
BERT-base	90.7 (40.7)	91.6 (31.7)	58.1 (41.3)	89.0 (38.8)	67.1 (34.3)	61.5 (17.6)	70.9 (54.7)
BERT-large	89.3 (38.0)	91.0 (33.0)	57.4 (34.6)	88.8 (38.4)	**67.4** (35.5)	77.6 (31.3)	69.0 (53.1)
RoBERTa-base	91.5 (39.5)	92.5 (27.8)	50.1 (49.1)	87.2 (36.6)	66.1 (37.3)	87.0 (38.6)	70.9 (52.9)
RoBERTa-large	90.1 (40.6)	91.9 (35.5)	55.4 (46.8)	88.9 (37.2)	65.7 (38.2)	**88.6** (39.3)	70.2 (49.1)
DeBERTa	92.3 (42.2)	**92.9** (32.5)	**65.7** (58.6)	**91.0** (41.5)	63.0 (36.9)	85.2 (37.3)	**72.9** (58.2)

Table 1: This table lists results from the probing and fine-tuning experiments We include in group 1 probing accuracy (%) from two baselines. Group 2 and 3 shows probing accuracy (%) of the linear classifier (group 2) and the MLP classifier (group 3). For each probe, we also record the selectivity score (%) from control tasks.

a contradiction signature as **Contra-sig**$_1$, tokens in the second phrase of a contradiction signature as **Contra-sig**$_2$, and irrelevant tokens as **None**. Table 2 shows an example, with {*have*, *each*, *played*, *twice*} as irrelevant tokens, {*defeats*, *Germany*} as tokens in the first phrase of the contradiction signature, and {*haven't*, *beaten*, *anybody*, *yet*} as tokens in the second phrase of the contradiction signature.

Monotonicity Polarization (Monot)

Monotonicity information supports word-replacement-based logical inferences that NLI systems can use. For each token, we assign a monotonicity mark that is either **Monotone** ($\uparrow$), **Antitone** ($\downarrow$), or **None** ($=$). To construct our dataset, we annotated monotonicity information on all sentences in the MED dataset (Yanaka et al. 2019) as training examples using a monotonicity annotation tool called Udep2Mono (Chen and Gao 2021). For testing, we extended a challenging gold-label monotonicity dataset used by Udep2Mono that includes multiple levels of monotonicity changes from different quantifiers and logical operators. For each sentence, we replicate ten sentences following the same syntactic format. Vertex probing is used for monotonicity polarization because a model must predict every token's monotonicity information.

Experiment Results and Findings

Do LMs encode information for inference?

Here we evaluate the degree to which pre-trained language models encode implicit information of linguistic properties that is essential to logical inference systems. We conducted probes on the five pre-trained language models. Table 1 shows results from the probing experiments.

Semantic Graph Construction With the linear classifier, all language models achieve high probing accuracy that outperforms the baselines. Together, with high selectivity, this is strong evidence that the information in these models' representations allows the classifier to recover the connectives of concepts, relations, and modifiers. Interestingly, the MLP classifier did not improve on the linear classifier significantly, suggesting that the information is easy to interpret. The performance here is consistent with language models' good performance on dependency parsing, semantic role labeling, and part-of-speech tagging (Tenney et al. 2019) which are related to semantic graph construction.

Semantic Alignment We observe that on lexical and anaphora based alignments (SA-Lex, SA-AP), language models show high probing accuracy that improves over the baselines significantly when using the MLP Classifier. This is evidence that language models encode linguistic information of these two types of semantic alignment since they also show high selectivity. Language models do not improve over the baselines when using the linear classifier, suggesting that these types of linguistic information might be hard to interpret. Language models only improved trivially over the baselines for alignments involving sentiment and relational knowledge (SA-ST, SA-RK). The insignificant improvement suggests that models weakly capture the information on these complex semantic alignment. Overall, the trend is that pre-trained models tend to show poor performance on complex semantic alignment that requires understanding and reasoning. This behavior is consistent with Tenney et al. (2019)'s finding that language models only encode a small amount of information on semantic reasoning.

Contradiction Signature and Monotonicity For the task on contradiction signature detection, all language models show poor performance with linear classifier except DeBERTa, which has relatively high accuracy (78.5%), validating that information on contradiction signatures is more accessible from DeBERTa than the other four models. After using the MLP classifier, all models' accuracy increased sig-

nificantly (above 90%) while maintaining very high selectivity. We attribute this partly to the fact that many contradictions are simple morphological negation and antonyms, which can be largely detected by using lexical semantics and syntax. The high accuracy is thus strong evidence that language models do encode a good amount of information on syntax and lexical semantics. For the monotonicity polarization task, language models show low accuracy with both the linear classifier and the MLP classifier. This suggests that these language models may not encode much monotonicity information that can support the polarization. Again, the results here show that pre-trained models encode more information on simple semantic phenomena (contradiction signature) than complex ones (monotonicity).

Label-wise Qualitative Analysis

To further understand the amount of linguistic information that pre-trained language models capture, we analyze label-wise probing quality for each task. Label-wise accuracy per task are shown in figure 3. We first observe that on some tasks (SA-Lex, SA-AP, ContraSig, SemGraph), pre-trained language models show high and balanced accuracy across labels. These behaviors are strong evidence that these models encode rich information on simple semantic phenomena. On the semantic graph construction task, the accuracy distribution is similar across models. The relatively low accuracy on modifier-to-relation (m-r) and modifier-to-concept (m-c) show the incompleteness of information in language models that support the linking of modifiers to words being modified. Language models seem to encode information for linking concepts to corresponding words since the accuracy is consistently high. Note that although anaphora (SA-AP) is a complex phenomena, models show decent performance which validates the finding from the previous experiment.

For other complex semantic alignment (SA-ST, SA-RK), the heatmaps show highly imbalanced label-wise accuracy. The models have higher accuracy on words in the hypothesis of an aligned pair than words in the premise, which has a much more complicated context. Since vertex-probing needs to locate phrases in a premise contributing to an entailment, the low accuracy on predicting span locations in a premise suggests that language models only encode very little linguistic information on complex semantic phenomena. For monotonicity polarization, the accuracy on each label is very different. Across language models, the accuracy on monotone polarity is higher than that on antitone and neutral polarity. This is consistent with other findings from other probing studies on monotonicity reasoning. (Yanaka et al. 2019; Geiger, Richardson, and Potts 2020). The results from this analysis validates our main finding from the previous experiment: models tend to encode less information on complex semantic phenomena.

Discussions

Information-Theoretic Probing We conducted additional experiments on information-theoretical probing to validate our findings based on probing accuracy. Recall that here we want to estimate the information gain between a pre-trained representation and a baseline representation. We

SA-AP

	LM1	LM2	LM3	LM4	LM5
α	65	89	88	91	89
⊘	58	67	86	86	82

SA-Lex

	LM1	LM2	LM3	LM4	LM5
α	84	88	75	78	86
⊘	85	85	84	84	88

SA-RK

	LM1	LM2	LM3	LM4	LM5
α_1	33	37	31	27	37
α_2	58	50	64	51	62
⊘	76	73	68	81	79

SA-ST

	LM1	LM2	LM3	LM4	LM5
α_1	17	1.3	6.6	0.8	0
α_2	92	87	85	91	75
$\perp_1$	11	1.2	3.3	0.8	0
$\perp_2$	0	0	0.7	0	0
⊘	72	57	58	56	53

Monotonicity

	LM1	LM2	LM3	LM4	LM5
←	75	79	59	72	78
→	47	44	52	48	63
‖	24	8.9	10	9.7	27

ContraSig

	LM1	LM2	LM3	LM4	LM5
$\perp_1$	91	87	89	87	90
$\perp_2$	87	86	90	88	88
⊘	94	94	95	96	96

SemGraph

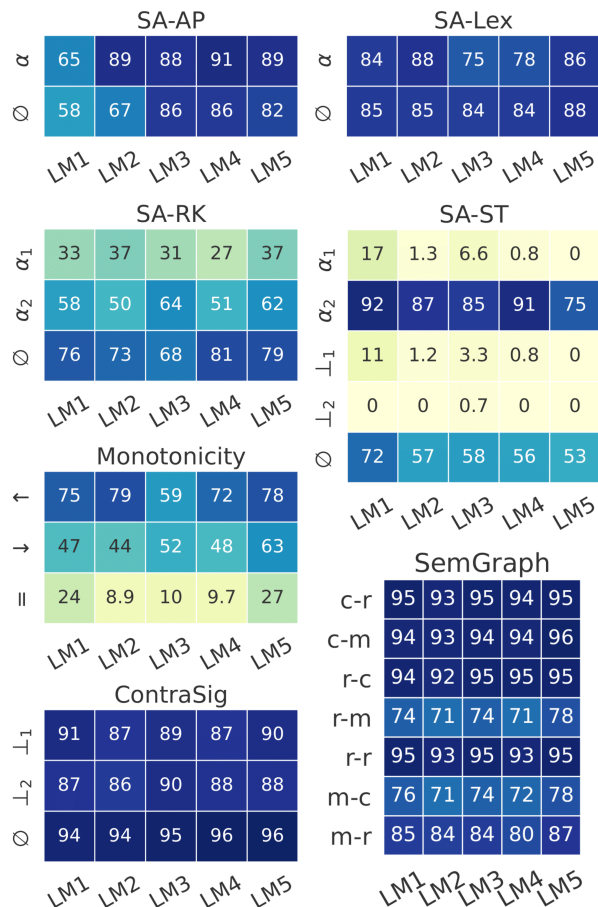|	LM1	LM2	LM3	LM4	LM5
c-r	95	93	95	94	95
c-m	94	93	94	94	96
r-c	94	92	95	95	95
r-m	74	71	74	71	78
r-r	95	93	95	93	95
m-c	76	71	74	72	78
m-r	85	84	84	80	87

Figure 3: Plots here shows label-wise accuracy across models for each inference information probing task. Here LM1-5 stands for the five language models in order (BERT-base, BERT-large, RoBERTa-base, RoBERTa-large, DeBERTa).

followed Pimentel et al. (2020)'s practice by using a more powerful probing classifier (MLP). We compared each language model to the best-performed baseline embedding. As table 2 shows, pre-trained language models on average encode more than 50% more information on the eight probing tasks. Overall, pre-trained language models encode more information than baseline embedding consistently across all tasks. Among these, the highest information gain of language models is on lexical alignment (more than 100% of increase). This is surprising since baseline word embeddings are a representation of word semantics. We hypothesize that this is due to the proximity of words that contradicts each other in the embedding space. Based on the results, we conclude that pre-trained language models encode significantly more information on linguistic information of logical inference than conventional word embeddings.

Classifier Expressiveness Some of our findings contradict several statements made by Hewitt and Liang (2019) regarding classifier expressiveness. First, they claim that one should choose a less expressive classifier over a highly ex-

Model	SemGraph	ContraSig	Monot	SA-Lex
LM1	0.10 (6%)	1.16 (65%)	0.82 (58%)	1.16 (116%)
LM2	0.08 (5%)	1.15 (64%)	0.8 (56%)	1.14 (114%)
LM3	0.10 (6%)	1.16 (65%)	0.85 (60%)	1.14 (114%)
LM4	0.09 (5%)	1.14 (64%)	0.9 (63%)	1.15 (115%)
LM5	0.11 (7%)	1.16 (65%)	0.78 (55%)	1.16 (116%)
	SA-ST	**SA-AP**	**SA-RK**	**Avg**
LM1	0.39 (39%)	1.07 (54%)	0.74 (80%)	0.78 (59%)
LM2	0.41 (41%)	1.02 (51%)	0.74 (80%)	0.76 (59%)
LM3	0.74 (74%)	1.04 (52%)	0.80 (86%)	0.83 (65%)
LM4	0.62 (62%)	1.05 (53%)	0.78 (84%)	0.82 (64%)
LM5	0.67 (67%)	1.02 (51%)	0.67 (72%)	0.79 (62%)

Table 2: Results on infromation-theoretic probing. Here we show the amount of extra information each language model shares, compared to the best baseline word embedding. The percentage of information gain is listed in the parenthesis.

Model	Monot	SA-ST	SA-RK
LM1	60.9 (▲ 2.8)	79.8 (▲ 12.7)	81.4 (▲ 14.2)
LM2	60.2 (▲ 2.8)	77.6 (▲ 10.2)	80.5 (▲ 14.6)
LM3	59.4 (▲ 9.3)	78.0 (▲ 11.9)	88.0 (▲ 22.7)
LM4	58.0 (▲ 2.6)	76.1 (▲ 10.4)	89.7 (▲ 27.7)
LM5	68.7 (▲ 3.0)	78.9 (▲ 14.9)	89.6 (▲ 19.1)

Table 3: Probing accuracy (%) of fine-tuned NLI models on probing tasks that did not benefit from pre-trained models.

pressive one (linear over MLP) since the prior one has higher selectivity. However, based on the accuracy, we observe that the linear classifier has worse performance for semantically-oriented tasks than the MLP classifier. We also found that the MLP classifier can achieve similar selectivity as a linear classifier for these tasks while achieving higher accuracy. These findings suggest that linear classifiers could misinterpret semantic information from a representation, which supports Pimentel et al. (2020)'s arguments on using a more powerful classifier. Secondly, they claim that a probing classifier with sufficient expressiveness can learn any task on top of a lossless representation with enough training examples. However, we found that even high expressive classifiers like MLP fail to perform well on tasks with monotonicity and higher-level semantic reasoning.

Can LMs learn missing information? Here we evaluate whether pre-trained language models can obtain linguistic information for inference, missing from their pre-trained representations, through fine-tuning for the NLI task. We select a version fine-tuned on the MultiNLI dataset for each language model. We probed these fine-tuned models for three tasks that did not benefit from the pre-trained language models. Our probing results are shown in Table 3, and we only record probings with the best performance here. We observe that all language models' fine-tuned representations improve over their pre-trained representations significantly on the sentiment (SA-ST) and relational-knowledge-based (SA-RK) semantic alignment. In contrast, they do not improve the performance on monotonicity polarization (Monotonicity). The results show that language models can capture linguistic information on some types of semantic reasoning

but not on monotonicity. Possible explanations are that these models could not obtain monotonicity information during fine-tuning, or the training data of MultiNLI does not contain enough examples about monotonicity.

Conclusions and Future Work

We presented a systematic study on determining if pretrained language models encode implicit linguistic information essential to symbolic inference methods. For each probing task, we constructed associating datasets. We then conducted probings on pre-trained language models using both linear and MLP classifiers for each task. In general, we first found that baseline word embeddings do not contain much linguistic information for inference and contextualized representations of language models encode some levels of linguistic information. However, they encode more information on simple semantic phenomena than on complex phenomena. Moreover, we found that linear classifiers can predict correctly under syntactical information but often misinterpret information on semantics resulting in low classification accuracy. Our label-wise qualitative analysis found that the amount of linguistic information being encoded is different across task-specific labels. In particular, language models encode more linguistic information for one label than other labels. This label-wise information difference again justifies the absence and incompleteness of some linguistic information for inference in language models. Furthermore, we found that language models can effectively learn some types of missing information on complex semantic reasoning through fine-tuning for the NLI task. Overall, language models show potential to serve as knowledge bases of linguistic information that supports robust symbolic reasoning.

We believe that our probing and analysis provide an adequate picture of whether critical linguistic information for inference exists in pre-trained language models. Moreover, our probing can inspire future systems on combining neural and symbolic NLP methods. For future work, one could conduct further analysis on each type of linguistic information in language models by constructing more detailed probing datasets. One could also design logical systems that can access linguistic information from pre-trained language models and apply them in the inference process for improved performance on large benchmarks.

Acknowledgments

References

Abzianidze, L. 2020. Learning as Abduction: Trainable Natural Logic Theorem Prover for Natural Language Inference. In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, 20–31. Barcelona, Spain (Online): Association for Computational Linguistics.

Bouraoui, Z.; Camacho-Collados, J.; and Schockaert, S. 2020. Inducing Relational Knowledge from BERT. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05): 7456–7463.

Chen, Z.; and Gao, Q. 2021. Monotonicity Marking from Universal Dependency Trees. In *Proceedings of the 14th International Conference on Computational Semantics (IWCS)*, 121–131. Groningen, The Netherlands (online): Association for Computational Linguistics.

Chen, Z.; Gao, Q.; and Moss, L. S. 2021. NeuralLog: Natural Language Inference with Joint Neural and Logical Reasoning. In *Proceedings of *SEM 2021: The Tenth Joint Conference on Lexical and Computational Semantics*, 78–88. Online: Association for Computational Linguistics.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.

Geiger, A.; Richardson, K.; and Potts, C. 2020. Neural Natural Language Inference Models Partially Embed Theories of Lexical Entailment and Negation. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, 163–173. Online: Association for Computational Linguistics.

He, P.; Liu, X.; Gao, J.; and Chen, W. 2021. DE-BERTA: DECODING-ENHANCED BERT WITH DISENTANGLED ATTENTION. In *International Conference on Learning Representations*.

Hewitt, J.; and Liang, P. 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)*, 2733–2743. Hong Kong, China: Association for Computational Linguistics.

Hewitt, J.; and Manning, C. D. 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)*, 4129–4138. Minneapolis, Minnesota: Association for Computational Linguistics.

Hu, H.; Chen, Q.; Richardson, K.; Mukherjee, A.; Moss, L. S.; and Kuebler, S. 2020. MonaLog: a Lightweight System for Natural Language Inference Based on Monotonicity. In *Proceedings of the Society for Computation in Linguistics 2020*, 334–344. New York, New York: Association for Computational Linguistics.

Joulin, A.; Grave, E.; Bojanowski, P.; and Mikolov, T. 2017. Bag of Tricks for Efficient Text Classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 427–431. Association for Computational Linguistics.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *ArXiv*, abs/1907.11692.

Marelli, M.; Menini, S.; Baroni, M.; Bentivogli, L.; Bernardi, R.; and Zamparelli, R. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *LREC*.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient Estimation of Word Representations in Vector Space. In Bengio, Y.; and LeCun, Y., eds., *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Pennington, J.; Socher, R.; and Manning, C. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. Doha, Qatar: Association for Computational Linguistics.

Petroni, F.; Rocktäschel, T.; Riedel, S.; Lewis, P.; Bakhtin, A.; Wu, Y.; and Miller, A. 2019. 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)*, 2463–2473. Hong Kong, China: Association for Computational Linguistics.

Phang, J.; Yeres, P.; Swanson, J.; Liu, H.; Tenney, I. F.; Htut, P. M.; Vania, C.; Wang, A.; and Bowman, S. R. 2020. `jiant` 2.0: A software toolkit for research on general-purpose text understanding models. http://jiant.info/.

Pimentel, T.; Valvoda, J.; Hall Maudslay, R.; Zmigrod, R.; Williams, A.; and Cotterell, R. 2020. Information-Theoretic Probing for Linguistic Structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 4609–4622. Online: Association for Computational Linguistics.

Rogers, A.; Kovaleva, O.; and Rumshisky, A. 2020. A Primer in BERTology: What We Know About How BERT Works. *Transactions of the Association for Computational Linguistics*, 8: 842–866.

Tenney, I.; Xia, P.; Chen, B.; Wang, A.; Poliak, A.; McCoy, R. T.; Kim, N.; Durme, B. V.; Bowman, S. R.; Das, D.; and Pavlick, E. 2019. 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.

Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. R. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. *CoRR*, abs/1804.07461.

Williams, A.; Nangia, N.; and Bowman, S. 2018. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proceedings of the 2018 Conference*

of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), 1112–1122. New Orleans, Louisiana: Association for Computational Linguistics.

Yanaka, H.; Mineshima, K.; Bekki, D.; Inui, K.; Sekine, S.; Abzianidze, L.; and Bos, J. 2019. Can Neural Networks Understand Monotonicity Reasoning? In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, 31–40. Florence, Italy: Association for Computational Linguistics.

Yanaka, H.; Mineshima, K.; Martínez-Gómez, P.; and Bekki, D. 2018. Acquisition of Phrase Correspondences Using Natural Deduction Proofs. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 756–766. New Orleans, Louisiana: Association for Computational Linguistics.