oLMpics - On what Language Model Pre-training Captures

Alon Talmor[1,2] **Yanai Elazar**[1,3] **Yoav Goldberg**[1,3] **Jonathan Berant**[1,2]

[1]The Allen Institute for AI
[2]Tel-Aviv University
[3]Bar-Ilan University
{alontalmor@mail,joberant@cs}.tau.ac.il,
{yanaiela,yoav.goldberg}@gmail.com

Abstract

Recent success of pre-trained language models (LMs) has spurred widespread interest in the language capabilities that they possess. However, efforts to understand whether LM representations are useful for symbolic reasoning tasks have been limited and scattered. In this work, we propose eight reasoning tasks, which conceptually require operations such as comparison, conjunction, and composition. A fundamental challenge is to understand whether the performance of a LM on a task should be attributed to the pre-trained representations or to the process of fine-tuning on the task data. To address this, we propose an evaluation protocol that includes both zero-shot evaluation (no fine-tuning), as well as comparing the learning curve of a fine-tuned LM to the learning curve of multiple controls, which paints a rich picture of the LM capabilities. Our main findings are that: (a) different LMs exhibit qualitatively different reasoning abilities, e.g., ROBERTA succeeds in reasoning tasks where BERT fails completely; (b) LMs do not reason in an abstract manner and are *context-dependent*, e.g., while ROBERTA can compare ages, it can do so only when the ages are in the typical range of human ages; (c) On half of our reasoning tasks all models fail completely. Our findings and infrastructure can help future work on designing new datasets, models and objective functions for pre-training.

1 Introduction

Large pre-trained language models (LM) have revolutionized the field of natural language processing in the last few years (Peters et al., 2018a; Devlin et al., 2019; Liu et al., 2019; Yang et al., 2019; Radford et al., 2019), leading to undeniable empirical gains in almost every benchmark. This has instigated research exploring what is captured by
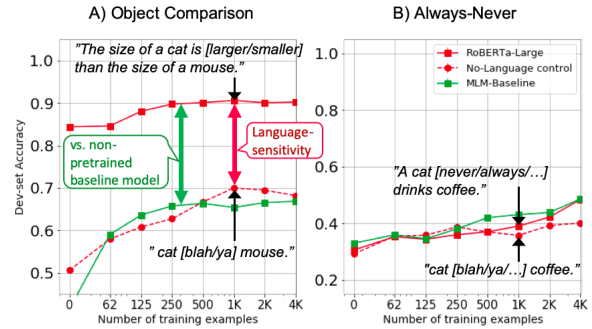

Figure 1: Overview of our experimental design. Two probes are evaluated using learning curves (including zero-shot). ROBERTA-L's (red squares, upper text in black) accuracy is compared to a NO LANGUAGE control (red circles, lower text in black), and MLM-BASELINE, which is not pre-trained (green squares). Here, we conclude that the LM representations are well-suited for task A), whereas in task B) the model is adapting to the task during fine-tuning.

the contextualized representations that these LM compute, revealing that they encode substantial amounts of syntax and semantics (Linzen et al., 2016b; Peters et al., 2018b; Tenney et al., 2019b; Goldberg, 2019; Hewitt and Manning, 2019; Tenney et al., 2019a; Shwartz and Dagan, 2019; Lin et al., 2019; Coenen et al., 2019).

Despite these efforts, it remains unclear *what skills are difficult to learn from a LM objective only?* In this paper, we propose a diverse set of probing tasks for types of symbolic reasoning that are potentially difficult to capture using a LM objective (see Table 1). Our intuition is that since a LM objective focuses on word co-occurrence, it will struggle with tasks that are considered to involve symbolic reasoning such as determining whether a *conjunction* of properties is held by an object, and *comparing* the sizes of different objects. Understanding what is missing from current LMs may help design datasets and objectives that will endow models with the missing capabilities.

Probe name	Setup	Example
ALWAYS-NEVER	MC-MLM	*A chicken [MASK] has horns.* **A.** never **B.** rarely **C.** sometimes **D.** often **E.** always
AGE COMPARISON	MC-MLM	*A 21 year old person age is [MASK] than a 35 year old person.* **A.** younger **B.** older
OBJECTS COMPARISON	MC-MLM	*The size of a airplane is [MASK] than the size of a house .* **A.** larger **B.** smaller
ANTONYM NEGATION	MC-MLM	*It was [MASK] hot, it was really cold .* **A.** not **B.** really
PROPERTY CONJUNCTION	MC-QA	*What is usually located at hand and usually used for writing?* **A.** pen **B.** spoon **C.** computer
TAXONOMY CONJUNCTION	MC-QA	*A ferry and a floatplane are both a type of [MASK].* **A.** vehicle **B.** airplane **B.** boat
ENCYC. COMPOSITION	MC-QA	*When did the band where Junior Cony played first form?* **A.** 1978 **B.** 1977 **C.** 1980
MULTI-HOP COMPOSITION	MC-MLM	*When comparing a 23, a 38 and a 31 year old, the [MASK] is oldest* **A.** second **B.** first **C.** third

Table 1: Examples for all our reasoning probes. We use two types of experimental setups, explained in §2. The first answer (A) is the correct answer.

However, how does one verify whether pre-trained representations hold information that is useful for a particular task? Past work mostly resorted to fixing the representations and *fine-tuning* a simple, often linear, randomly initialized probe, to determine whether the representations hold relevant information (Ettinger et al., 2016; Adi et al., 2016; Belinkov and Glass, 2019; Hewitt and Manning, 2019; Wallace et al., 2019; Rozen et al., 2019; Warstadt et al., 2019; Richardson et al., 2019). However, it is difficult to determine whether success is due to the pre-trained representations or due to fine-tuning itself (Hewitt and Liang, 2019). To handle this challenge, we include multiple controls that improve our understanding of the empirical results.

Our "purest" setup is zero-shot: we cast tasks in the *masked LM* format, and use a pre-trained LM without any fine-tuning. For example, given the statement *"A cat is [MASK] than a mouse"*, a LM can decide if the probability of *"larger"* is higher than *"smaller"* for the a masked word (Figure 1). If a model succeeds without pre-training over many pairs of objects, then its representations are useful for this task. However, if it fails, it could be due to a mismatch between the language it was pre-trained on and the language of the task (which might be automatically generated and with grammatical errors). Thus, we also compute the learning curve (Figure 1), by fine-tuning on increasing amounts of data the already pre-trained masked language modeling (MLM) head, a 1-hidden layer MLP on top of the model's contextualized representations. A model that can adapt from fewer examples arguably has better representations for it.

Moreover, to diagnose whether model performance is related to pre-trained representations or fine-tuning, we add controls to every experiment (Figures 1,3). First, we add a control that makes minimal use of language tokens, i.e., *"cat [MASK] mouse"* (NO LANGUAGE in Figure 1). If a model

is agnostic to the presence of language, then performance can be attributed to fine-tuning rather than to pre-training. Similar logic is used to compare against baseline models that are not pre-trained at all. (except for non-contextualize word embeddings). Overall, our setup provides a rich picture of whether LM representations are useful for solving a wide range of reasoning tasks.

We introduce eight tasks that test different types of reasoning, as shown in Table 1. We run experiments using several pre-trained LMs, based on BERT (Devlin et al., 2019) and ROBERTA (Liu et al., 2019). We find that there are clear qualitative differences between different LMs with similar architecture. For example, ROBERTA-LARGE (ROBERTA-L) can perfectly solve reasoning tasks, such as comparing numbers, even in a zero-shot setup, while other models' performance is close to random. However, good performance is highly *context-dependant*. Specifically, we repeatedly observe that even when a model solves a task, small changes to the input quickly derail it to low performance. For example, ROBERTA-L can almost perfectly compare people's ages, when the numeric values are in the expected range (15-105), but miserably fails if the values are outside this range. Interestingly, it is able to reliably answer, and reverse the order of younger/older, when ages are specified through the year of birth in the range 1920-2000. This highlights that the LMs ability to solve this task is strongly tied to the specific values and linguistic context and does not generalize to arbitrary scenarios. Last, we find that in four out of eight tasks, all LMs perform poorly compared to the controls.

Our contributions are summarized as follows:

- A set of probes that test whether specific reasoning skills are encoded in the representations of pre-trained LMs.
- An evaluation protocol for understanding whether a capability is encoded in pre-traind

representations or is learned during fine-tuning.

- An analysis of reasoning skills that current LMs possess. We find that LMs with similar architectures are qualitatively different, that their success is highly context-dependent, and that in many cases, all LMs fail.
- Code and infrastructure for designing new probes and testing them on a large set of pre-trained language models.

The code and models will be available at `http://github.com/alontalmor/oLMpics`.

2 Models

We now turn to the architectures and loss functions used throughout the different probing tasks.

2.1 Pre-trained Language Models

All models in this paper take a sequence of tokens $x = (x_1, \ldots, x_n)$, and compute contextualized representations with a pre-trained LM, that is, $h = \text{ENCODE}(x) = (h_1, \ldots, h_n)$. Specifically, we consider (a) BERT: (Devlin et al., 2019), a transformer-based architecture (Vaswani et al., 2017), trained with a *masked-language modeling* objective (MLM), i.e., the model is trained to predicts words that are masked from the input; including BERT-WHOLE-WORD-MASKING (BERT-WWM), that was trained using *whole-word-masking* (b) ROBERTA (Liu et al., 2019), which has the same architecture as BERT, but was trained on more data and optimized carefully.

2.2 Probing setups

We probe the pre-trained LMs using two setups: multi-choice masked LM (MC-MLM) and multi-choice question answering (MC-QA). The MC-MLM setup is our default one, and is used for tasks where the answer-set is small, consistent across the different questions, and each answer appears as a single item in the word-piece vocabulary. The MC-QA setup is used when the answer-set substantially varies between questions, and many of the answers consist of more than one word, or expected not to appear in the word-piece vocabulary.

MC-MLM We adapt the original MLM setup to arbitrary probing tasks, by converting it to a multi-choice setup (MC-MLM). Specifically, the input to the LM is the sequence $x = ([CLS], \ldots, x_{i-1}, [MASK], x_{i+1}, \ldots, [SEP])$, where a single token x_i is masked. Then, the contextualized representation h_i is passed through a MC-MLM *head*:

$$l = FF_{\text{MLM}}(h_i) \in \mathbb{R}^{|\mathcal{V}|}, p = \text{softmax}(m \odot l),$$

where $\mathcal{V}$ is the vocabulary, and FF_{MLM} is a 1-hidden layer MLP. Applying the mask $m \odot l$, guarantees that the support of the probability distribution will be over exactly K candidate tokens: the correct one and $K - 1$ distractors. This is done to speed-up the adaptation rate of the model, and allows for reasonable accuracy to be achieved from limited amounts of data. Training minimizes cross-entropy loss given the gold masked token.

Figure 2 illustrates this setup. The input "[CLS] *Cats* [MASK] *drink coffee* [SEP]" is passed through the model, the contextualized representation of the masked token is passed through the MC-MLM head, and the final distribution is over the vocabulary words *"always"*, *"sometimes"* and *"never"*, where the gold token is *"never"*, in this case.

A compelling advantage of this setup, is that reasonable performance can be obtained without training, using the original LM representations and the already pre-trained MLM head weights.

MC-QA Here, we use the standard setup for answering multi-choice questions with pre-trained LMs (Talmor et al., 2019; Mihaylov et al., 2018; Zellers et al., 2018). Given a question q and candidate answers $a_1, \ldots, a_K$, we compute for each candidate answer a_k representations $h^{(k)}$ from the input tokens "[CLS] q [SEP] a_k [SEP]". Then the probability over answers is obtained using the *multi-choice QA head*:

$$l^{(k)} = FF_{\text{QA}}(h_1^{(k)}), p = \text{softmax}(l^{(1)}, \ldots, l^{(K)}),$$

where FF_{QA} is a 1-hidden layer MLP that is run over the [CLS] (first) token of an answer candidate and outputs a single logit. Note that in this setup that parameters of FF_{QA} cannot be initialized using the original pre-trained LM.

2.3 Baseline Models

To provide a lower bound on the performance of pre-trained LMs, we introduce two baseline models with only non-contextualized representations.

MLM-BASELINE This serves as a lower-bound for the MC-MLM setup. The input to $FF_{\text{QA}}(\cdot)$ is the hidden representation $h \in \mathbb{R}^{1024}$ (for large models). To obtain a similar architecture with
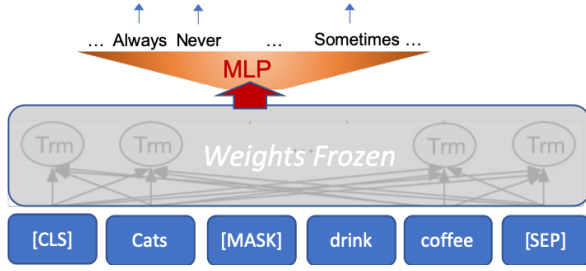
Figure 2: MC-MLM setup. A few vocabulary tokens are possible outputs; the MLP is initialized with pre-trained weights.

non-contextualized representations, we concatenate the first 20 tokens of each example, representing each token with a 50-dimensional GLOVE vector (Pennington et al., 2014), and pass this 1000-dimensional representation of the input through FF_{QA}, exactly like in MC-MLM. In all probes, phrases are limited to 20 tokens. If there are less than 20 tokens in the input, we zero-pad the input.

MC-QA baseline This serves as a lower-bound for MC-QA. We use the ESIM architecture over GLOVE representations, which is known to provide a strong model when the input is a pair of text fragments (Chen et al., 2017). We adapt the architecture to the multi-choice setup using the procedure proposed by Zellers et al. (2018).

3 Controlled Experiments

We now elaborate on our experimental design, and the controls used to better interpret the empirical results. We will use the results of our first task, AGE-COMPARISON, as a running example, where we test for models' ability to compare the numeric value of ages.

3.1 Zero-shot Experiments with MC-MLM

Fine-tuning pre-trained LMs makes it difficult to disentangle what is captured by the original representations and what was learned during fine-tuning (Hewitt and Liang, 2019). Thus, ideally, one should test pre-trained LMs using the original weights directly without *any* fine-tuning (Linzen et al., 2016a; Goldberg, 2019). The MC-MLM setup (§2) uses the pre-trained MLM head and thus achieves exactly that. One only needs to design the task as a statement with a single masked token and K possible output tokens. In AGE-COMPARISON, this is done using the phrase

"A AGE-1 *year-old person age is* [MASK] *than a* AGE-2 *year-old person."*, where AGE-1 and AGE-2 are replaced with different integers, and the $K = 2$ possible answers are *"younger"* and *"older"*. Other than that, no training is needed, and the original representations are tested.

Figure 3A provides an example of such zero-shot evaluation. Different values are assigned to AGE-1 and AGE-2, and the pixel is colored when the model predicts *"younger"*. Accuracy is measured as the proportion of cases where the model outputs the correct token. The performance of BERT-WWM, is on the left, and of ROBERTA-L on the right (green). The results in Figure 3A and Table 2 show that ROBERTA-L consistently represents numbers and compares them in (96% accuracy), BERT-WWM achieves higher than random accuracy (63% accuracy), while BERT-LARGE (BERT-L) is roughly random (49% accuracy). ROBERTA-L errs when the difference between numerical values is small, while BERT-WWM errors are more scattered. The performance of MLM-BASELINE is random as expected, as the MLP_{MLM} weights are randomly initialized.

3.2 Learning Curves

Despite the advantages of zero-shot evaluation, performance of a model might be adversely affected by mismatches between the language the pre-trained LM was trained on and the language of the examples in our tasks. For example, one task that focuses on negation uses a the template *"It was [MASK] fat, it was really slim"*, with candidate answers *"not"*, *"very"*. Such sentences may derail a LM due to the unnatural language rather than the ability to interpret negation.

To tackle this, we fine-tune models in the MC-MLM setup with a limited number of examples. We assume that if the LM representations are useful for a task, then it will require few examples to overcome the language mismatch described above and achieve high performance. In almost all cases, we train with N examples, where $N \in \{62, 125, 250, 500, 1K, 2K, 4K\}$. To account for optimization instabilities, we fine-tune multiple times with different random seeds for each value of N,[1] and report average accuracy Dodge et al. (2019). The representations h are fixed at fine-tuning time, and we only train the already pre-

---

[1] six times when the number of examples is small, and three times when it is large.

Model	Zero shot	MLP$_{\text{MLM}}$		Linear		LangSense	
	shot	WS	Max	WS	Max	pert	nolang
RoBERTa-L	96	96	100	96	100	43	47
BERT-WWM	63	78	100	63	86	11	22
BERT-L	49	55	91	48	51	1	3
BERT-B	49	51	62	50	58	0	2
RoBERTa-B	37	56	99	45	50	5	4
Baseline	48	61	74	-	-	1	0

Table 2: AGE-COMPARISON results. LANGSENSE are the Language Sensitivity controls, pert is PERTRUBED LANGUAGE and nolang is NO LANGUAGE. The baseline row is MLM-BASELINE. Random accuracy is 50%.
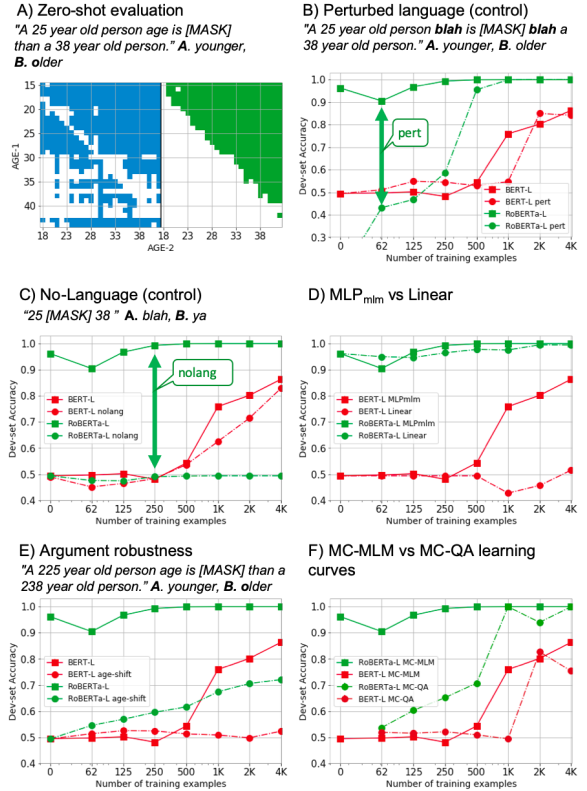

Figure 3: An illustration of our evaluation protocol that relies on learning curves. We compare ROBERTA-L (red), BERT-WWM (blue), and BERT-L (green), controls are in dashed lines and circles. Zero-shot evaluation on the top left, does the model predicts AGE-1 is *"younger"* (in color) than AGE-2, or *"older"* (in white).

trained parameters of MLP$_{\text{MLM}}$.

Learning-curve metrics Learning curves are informative, but inspecting a many learning curves could be difficult to digest. Thus, we summarize them using two aggregate statistics. We report: (a) MAX, i.e., the maximal accuracy on the learning curve, and (b) the metric WS, which is a weighted average of accuracies on the learning curve, where higher weights are given to points where N is small.[2] WS emphasizes our focus on performance given little training data, as it highlights what was encoded by the model *before* fine-tuning. WS is related to the area under the accuracy curve, and to the online code metric, proposed by Yogatama et al. (2019); Blier and Ollivier (2018).

For AGE-COMPARISON, the solid lines in Figure 3B illustrate the learning curves of ROBERTA-L and BERT-L, and Table 2 shows the aggregate statistics. We fine-tune the model by replacing AGE-1 and AGE-2 with values between 43 and 120, but test with values between 15 and 38, to guarantee that the model *generalizes* to values unseen at training time. Again, we see that the representations learned by ROBERTA-L are already equipped with the knowledge necessary for solving this task from few examples.

3.3 Controls

Comparing learning curves tells us which model learns from fewer examples. However, since MLPs can approximate any function, it is difficult to determine whether the performance is tied to the knowledge acquired at pre-training time, or to the process of fine-tuning itself. We present controls that attempt to disentangle these two factors.

Are LMs sensitive to the language input? We are interested in whether pre-trained representations handle reasoning tasks over language examples. Thus, a natural control is to present the reasoning task *without* language and check whether performance drops. If the learning curve of a model does not change when the input is perturbed or even mostly deleted, then the model shows low *language sensitivity* and the pre-trained representations do not explain the probe performance. This approach is related to recent work by Hewitt and Liang (2019), who proposed a control task, where the learning curve of a model is compared to a learning curve when words are associated with random behaviour. We propose two control tasks:

NO LANGUAGE *control* We remove all input tokens, except for [MASK] and the *arguments* of the task, i.e., the tokens that are necessary for computing the desired output. In AGE-COMPARISON, an example is reduced to the phrase *"24 [MASK]*

---

[2]We use the decreasing weights $W = (0.23, 0.2, 0.17, 0.14, 0.11, 0.08, 0.07)$.

55", where the candidate answers are the words *"blah"*, for *"older"*, and *"ya"*, for *"younger"*. If the learning curve is similar to when the full example is given (low language sensitivity), then the LM is not strongly using the language input.

The dashed lines in Figure 3C illustrate the learning curves in the NO LANGUAGE control. ROBERTA-L (green) shows high language sensitivity, while BERT-L (red) has low language sensitivity. This suggests it handles this task mostly through examples provided during fine-tuning. Table 2 paints a similar picture, where the metric we use is identical to WS, except that instead of averaging accuracies, we average the *difference* in accuracies between the standard model and the NO LANGUAGE control (rounding negative numbers to zero). For ROBERTA-L the value is almost 50, because ROBERTA-L gets almost 100% accuracy in the presence of language, and is random (50% accuracy) without language.

PERTRUBED LANGUAGE *control* A more targeted control, is to replace words that are central for the reasoning task with nonsense words. For example, in a PROPERTY CONJUNCTION task we replace the word *"and"* with the word *"blah"*, resulting in examples such as *"What is located at hand **blah** used for writing?"*. If the learning curve of the PERTRUBED LANGUAGE control is similar to the original example, this indicates that the model does not utilize the pre-trained representation of *"and"* to solve the task, and may not capture its effect on the semantics of the statement.

Targeted words change from probe to probe. In AGE-COMPARISON, the targeted words are *"A AGE-1 year old person **blah** is [MASK] **blah** a AGE-2 year old person."* In this case the word *"age"* and the word *"than"*, which add context and suggest a comparison operation, are replaced. Figure 3B shows the learning curves for ROBERTA-L and BERT-L, where solid lines corresponds to the original examples and dashed lines are the PERTRUBED LANGUAGE control. Despite this minor perturbation, the performance of ROBERTA-L substantially decreases, implying that the model requires the language input to achieve high accuracy, while BERT-L is not influenced by this perturbation.

Does a linear transformation suffice? In MC-MLM, the representations h are fixed, and only the pre-trained parameters of MLP$_{MLM}$ are fine-tuned. As a proxy for measuring "how far" are

the representations from solving a task, we fix the weights of the first layer of MLP$_{MLM}$, and only train the final layer. Succeeding in this setup means that only a linear transformation of the representations is required for this probe. Table 2 and Figure 3D show (dashed lines) the performance of the linear setup (LINEAR), compared to MLP$_{MLM}$. We observe that ROBERTA-L can reach high performance in this setup, while BERT-L remains random.

Are LMs sensitive to the input distribution? In probes where the *arguments* of the symbolic reasoning can take a range of values, we can test whether models are robust to changes in the input distribution. In AGE-COMPARISON, we shift ages to values that are not within a human life span: $215 - 230$. Figure 3E shows that models are substantially affected by shift the age values. ROBERTA-L partially recover and achieve fair accuracy, but the drop in zero-shot performance illustrates that the ability of LMs to predict *"younger"* or *"older"* is tied to the natural distribution of ages, and the models cannot just abstractly reason about numbers in any context.

3.4 Multi-Choice Question Answering

Constructing a MC-MLM probe limits the answer candidates to a single token from the word-piece vocabulary. To relax this setup we also explore the MC-QA setup from §2.

In MC-QA, we phrase the task as a question, letting answer candidates be arbitrary strings, which provides ample expressivity (Gardner et al., 2019). In Table 1, PROPERTY CONJUNCTION and ENCYC. COMPARISON serve as examples for this setup. For AGE-COMPARISON we use the same task in MC-QA setup, Figure 3F shows the learning curves. Because in MC-QA, the network MLP$_{QA}$ cannot be initialized by pre-trained weights, it is impossible to obtain meaningful zero-shot results, and more training examples are needed to train MLP$_{QA}$. Still, the trends observed in MC-MLM remain, with ROBERTA-L achieving best performance with the fewest examples.

4 The oLMpic Games

We now present a series of challenges aimed at probing the symbolic reasoning abilities of pre-trained LMs.

4.1 Can LMs perform robust comparison?

Comparing two numeric values, requires representing the values and performing the comparison operations. In §3 we saw the AGE-COMPARISON task, in which the ages of two people were compared. We found that ROBERTA-L and to some extent BERT-WWM were able to handle this task, performing well under the controls. We expand on this to related comparison tasks and perturbations that assess the sensitivity of LMs to the particular context and to the numerical value.

Is ROBERTA-L comparing numbers or ages?
ROBERTA-L obtained zero-shot accuracy of 96% in AGE-COMPARISON. But is it robust? We test this using perturbations to the task and present the results in Figure 4. Figure 4A corresponds to the experiment from §3, where we observed that ROBERTA-L predicts *"younger"* (blue pixels) and *"older"* (white pixels) almost perfectly.

To test whether ROBERTA-L can compare ages given the birth year rather than the age, we use the statement *"A person born in `YEAR-1` is [MASK] than me in age, If i was born in `YEAR-2`."* Figure 4B shows that ROBERTA-L correctly flips *"younger"* to *"older"* (76% accuracy), reasoning that a person born in 1980 is older than a person born in 2000.

However, when evaluated on the exact same statement, but with values corresponding to typical *ages* instead of years (Figure 4D), ROBERTA-L obtains an accuracy of 12%, consistently outputting the wrong prediction. ==It seems that since the values are typical ages and not years, it disregards the statement, performing the comparison based on the values only and not the language.== We will revisit this tendency in §4.4.

Symmetrically, Figure 4C shows results when numeric values of ages are swapped with typical years of birth. ROBERTA-L is unable to handle this, always predicting *"older"*. This emphasizes that the model is sensitive to the argument values.

Can Language Models compare object sizes?
Comparing physical properties of objects requires knowledge of the numeric value of the property and the ability to perform comparison. Previous work has shown that such knowledge can be extracted from text and images (Bagherinezhad et al., 2016; Forbes and Choi, 2017; Yang et al., 2018a; Elazar et al., 2019). Can LMs do the same?
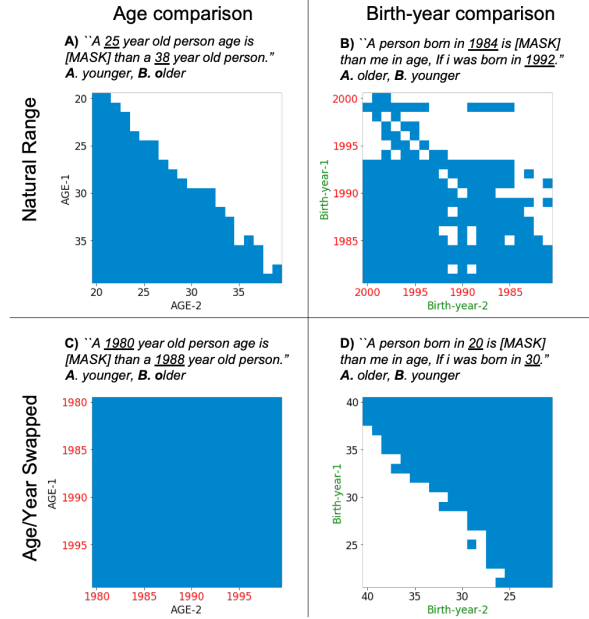***Probe Construction*** We construct statements of

Figure 4: AGE COMPARISON perturbations. Left side graphs are age-comparison, right side graphs are age comparison by birth-year. In the bottom row, the values of ages are swapped with birth-years and vice versa. In blue pixels the model predicts *"older"*, in white pixels *"younger"*. The first answer (A) is the correct answer.

the form *"The size of a `OBJ-1` is usually much [MASK] than the size of a `OBJ-2`."*, where the candidate answers are *"larger"* and *"smaller"*. To instantiate the two objects, we manually sample a list of objects from two domains: animals (e.g., *"camel"*, *"dinosaur"*) and general objects (e.g., *"pen"*, *"sun"*), and use the first domain for training and the second for evaluation. We bucket different objects based on the numerical value of their *size* based on their median value from DOQ (Elazar et al., 2019), and then manually fix any errors. Overall, we collected 127 and 35 objects for training and development respectively. We automatically instantiate object slots using objects that are in the same bucket.

Results ==ROBERTA-L excels in this task, starting from 84% accuracy in the zero-shot setup and reaching MAX of 91% (Table 3). Other models start with random performance and are roughly on par with MLM-BASELINE. ROBERTA-L shows sensitivity to the language, suggesting that the ability to compare object sizes is encoded in it.==

Analysis Table 4 shows results of running ROBERTA-L in the zero-shot setup over pairs of objects, where we sampled a single object from each bucket. Objects are ordered by their size

from small to large. Overall, ROBERTA-L correctly predicts *"larger"* below the diagonal, and *"smaller"* above it. Interestingly, errors are concentrated around the diagonal, due to the more fine-grained differences in sizes, and in the last column where we compare objects to *"sun"*. A possible explanation is that the model ignores the other object compared, stating that the *sun* is large.

Model	Zero	MLP_MLM		LINEAR		LANGSENSE	
	shot	WS	MAX	WS	MAX	pert	nolang
RoBERTa-L	84	87	91	86	90	11	27
BERT-WWM	52	64	81	62	74	2	8
BERT-L	49	54	71	51	57	4	4
BERT-B	53	57	71	55	59	8	3
RoBERTa-B	50	61	74	55	64	7	0
Baseline	41	61	73	-	-	0	2

Table 3: Results for the SIZE COMPARISON probe. Task example: *"The size of a shoe is usually much [MASK] than the size of a bench."*, Answer: smaller, Distractor: larger.

	nail	pen	laptop	table	house	airplane	city	sun
nail	-	smaller	smaller	smaller	smaller	smaller	smaller	smaller
pen	smaller	-	smaller	smaller	smaller	smaller	smaller	smaller
laptop	larger	larger	-	larger	smaller	smaller	smaller	smaller
table	larger	larger	larger	-	smaller	larger	smaller	larger
house	larger	larger	larger	larger	-	larger	smaller	larger
airplane	larger	larger	larger	larger	larger	-	larger	larger
city	larger	larger	larger	larger	larger	larger	-	larger
sun	larger	larger	larger	larger	larger	larger	larger	-

Table 4: Zero-shot performance of ROBERTA-L in SIZE COMPARISON.

4.2 Do LMs know *"always"* from *"often"*?

Adverbial modifiers such as *"always"*, *"sometimes"* or *"never"*, tell us about the frequency of various world facts. Anecdotally, when ROBERTA-L predicts a completion for the phrase *"Cats usually drink [MASK]."*, the top completion is *"coffee"* rather than *"water"*. This is since coffee is a frequent drink in literature, which the model was trained on. However, humans know that *"Cats NEVER drink coffee"*.

The *"Always-Never"* task We present statements, such as *"rhinoceros [MASK] have fur"*, with answer candidates, such as *"never"* or *"always"*. To succeed, the model should represent the frequency of the fact, and map the appropriate meaning of the adverbial modifier to that representation.

Probe Construction We manually craft templates that contain one slot for a `subject` and another for an `object`. Two example templates are *"`FOOD-TYPE` is [MASK] part of a `ANIMAL`'s diet."* and *"A `ANIMAL` [MASK] has*

a `BODY-PART`." (more examples available in Table 6). The `subject` slot is instantiated with concepts of the correct semantic type, according to the `isA` predicate in CONCEPTNET. In the example above we will find concepts that are of type `FOOD-TYPE` and `ANIMAL`. The `object` slot is then instantiated by forming masked templates of the form *"meat is part of a [MASK]'s diet."* and *"cats have [MASK]."* and letting BERT-L produce the top-20 completions. We filter out completions that do not have the correct semantic type according to the `isA` predicate. Finally, we crowdsource gold answers using Amazon Mechanical Turk. Annotators were presented with an instantiated template (with the masked token removed), such as *"Chickens have horns."* and chose the correct answer from 5 candidates: *"never"*, *"rarely"*, *"sometimes"*, *"often"* and *"always"*. We collected 1,300 examples with 1,000 used for training and 300 for evaluation.

Results Table 5 shows the results, where random accuracy is 20%, and the accuracy when taking a majority vote over the training set is 35.5%. In the zero-shot setup, accuracy is less than random. In the MLP_MLM and LINEAR setup accuracy reaches a maximum of 57% in BERT-L, but ==MLM-BASELINE obtains similar accuracy, implying that the task was mostly tackled at fine-tuning time, and the pre-trained representations did not contribute much.== Language controls strengthen this hypothesis, where performance hardly drops in the PERTRUBED LANGUAGE control (1-2 tokens per templates were replaced with *"blah"*), and slightly drops in the NO LANGUAGE control. Figure 5A compares the learning curve of the best performing model, BERT-WWM, and the NO LANGUAGE control. MLM-BASELINE consistently outperforms the LMs, which display only minor language sensitivity, suggesting that pre-training is not effective for solving this task.

Model	Zero	MLP_MLM		LINEAR		LANGSENSE	
	shot	WS	MAX	WS	MAX	pert	nolang
RoBERTa-L	14	42	54	23	39	1	6
BERT-WWM	11	44	56	27	47	2	6
BERT-L	14	42	57	25	46	2	3
BERT-B	15	41	55	28	47	1	8
RoBERTa-B	18	42	53	23	46	0	7
Baseline	14	46	57	-	-	0	1

Table 5: Results for the ALWAYS-NEVER probe. Task example: *"A dish with strawberries [MASK] contains tuna. "*, Answer: never, Distractos: (rarely, sometimes, often, always).

Analysis We generated predictions from the best model, BERT-WWM, and show analysis results in Table 6. For reference, the majority vote accuracy for human annotators is near 100%. Although the answers *"often"* and *"rarely"* are the gold answer in 19% of the training data, the LMs predict these answers in less than 1% of examples. In the template *"A dish with FOOD-TYPE [MASK] contains FOOD-TYPE."* the LM always predicts *"sometimes"*. Overall we find models do not perform well. Reporting bias (Gordon and Van Durme, 2013) may play a roll in LMs inability to correctly determine cases such as *"A rhinoceros NEVER has fur."* Interestingly, behavioral research conducted on blind humans shows they exhibit a similar bias (Kim et al., 2019).

Question	Answer	Distractor	Acc.
A dish with pasta [MASK] contains pork .	**sometimes**	sometimes	75
A dish with cheese [MASK] contains honey .	rarely	**sometimes**	
lamp is [MASK] placed in the room .	often	**sometimes**	68
stool is [MASK] placed in the box .	never	**sometimes**	
A hawk [MASK] has a beak .	**always**	sometimes	61
A lizard [MASK] has a wing .	never	**always**	
A mouse [MASK] has a horn .	never	**always**	
A pig is [MASK] smaller than a cat .	rarely	**always**	47
herring is [MASK] part of a fish's diet .	sometimes	**never**	41
meat is [MASK] part of a elephant's diet .	never	**sometimes**	
A calf is [MASK] larger than a dog .	**sometimes**	often	30
A raccoon is [MASK] larger than a dog .	sometimes	**never**	

Table 6: Error analysis for ALWAYS-NEVER. Model predictions are in bold, and Acc. shows accuracy per template.

4.3 Do LMs Capture Negation?

Ideally, the presence of the word *"not"* should affect the prediction of a masked token. However, it has been shown that the completion for both *"A cat is a [MASK]."* and *"A cat is **not** a [MASK] ."*, is *"cat"*. Several recent works have shown that indeed LMs do not take into account the presence of negation in sentences (Ettinger, 2019; Nie et al., 2019; Kassner and Schütze, 2019). Richardson et al. (2019) showed that LMs are able to successfully apply knowledge of antonyms in a natural language inference setup. Here, we add to this literature, by probing whether LMs can properly use negation in the context of *synonyms* vs. *antonyms*.

Do LMs Capture the Semantics of Antonyms? We check whether LMs use modifiers appropriately in the presence of synonyms vs. antonyms. In the statement *"He was [MASK] fast, he was very slow."*, [MASK] should be replaced with *"not"*, since *"fast"* and *"slow"* are antonyms. Conversely, in *"He was [MASK] fast, he was*
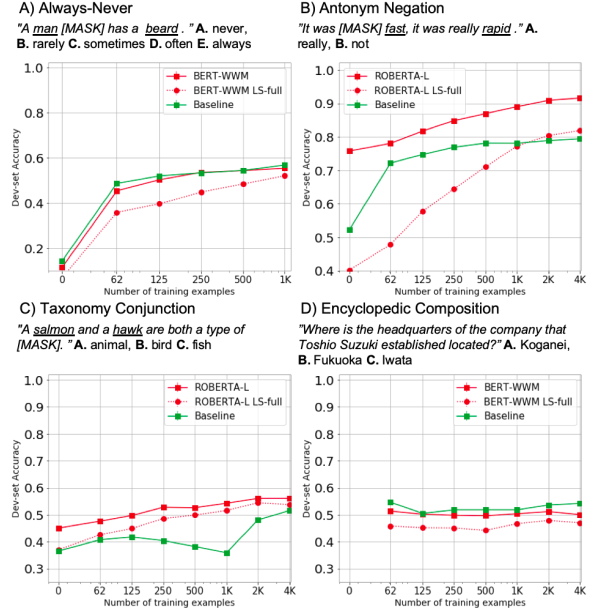

Figure 5: Learning curves in four tasks. For each task the best performing LM is shown alongside the NO LANGUAGE control and baseline model. The first answer (A) is the correct answer.

very rapid", the LM should choose a word like *"very"* in the presence of the synonyms *"fast"* and *"rapid"*. A LM that can correctly distinguish between *"not"* and *"very"*, demonstrates knowledge of the taxonomic relations antonym and synonym, as well as the ability to reason about how negation should be used in this context.

Probe Construction We sample synonym and antonym pairs from CONCEPTNET (Speer et al., 2017) and WORDNET (Fellbaum, 1998), and use Google Books Corpus to choose pairs that occur frequently in language. We make use of the statements introduced above. Half of the examples are synonym pairs and half antonyms, generating 4,000 training examples and 500 for evaluation.

Results ROBERTA-L shows higher than chance (50%) accuracy of 75% in the zero-shot setting, as well as high sensitivity to the linguistic input (Table 7). MLM-BASELINE, equipped with GloVe word embeddings, is able to reach a WS of 73 and MAX of 80%, which is comparable to the other LMs, suggesting they do not have a large advantage on this task. A learning curve of ROBERTA-L vs. controls is shown in Figure 5B.

Model	Zero shot	MLP$_{MLM}$		LINEAR		LANGSENSE	
		WS	MAX	WS	MAX	pert	nolang
RoBERTa-L	75	83	91	77	84	14	24
BERT-WWM	57	66	82	59	72	1	1
BERT-L	45	60	78	48	66	0	0
BERT-B	54	67	83	59	73	6	12
RoBERTa-B	58	69	86	62	76	10	15
Baseline	52	73	80	-	-	2	3

Table 7: Results for the ANTONYM NEGATION probe. Task example: *"It was [MASK] <u>tall</u> , it was really <u>short</u>."*, Answer: not, Distractor: really.

4.4 How well do LMs handle conjunctions of facts?

We present two probes where a model should understand the reasoning expressed by the word *and*.

Property conjunction CONCEPTNET is a Knowledge-Base that describes the properties of millions of concepts through its (`subject`, `predicate`, `object`) triples. We use CON-CEPNET to test whether LMs can find concepts for which a conjunction of properties holds. For example, we will create a question like *"What is located in a street and is related to octagon?"*, where the correct answer is *"street sign"*. Because answers are drawn from CONCEPTNET, they often consist of more than one word-piece, thus examples are generated in the MC-QA setup.
Probe Construction CONCEPTNET contains more than 34 million (`subject`, `predicate`, `object`) triples. To construct an example, we first choose a concept that has two properties in CONCEPTNET, where a property is a (`subject`, `predicate`) pair, or a (`predicate`, `object`) pair. For example, `street sign` has the properties (`atLocation`, `street`) and (`relatedTo`, `octagon`). Then, we create two distractor concepts, for which only one property holds: `car` has the property (`atLocation`, `street`), and `math` has the property (`relatedTo`, `octagon`). Given the answer concept, the distractors and the properties, we can automatically generate pseudo-langauge questions and answers by mapping 15 CONCEPT-NET predicates to natural language questions. We split examples such that concepts are disjoint between training and evaluation.
Results In MC-QA, we fine-tune the entire network and do not freeze any representations. Zero-shot cannot be applied since the weights of MLP$_{QA}$ are untrained. All LMs consistently improve as the number of examples increases, reaching a MAX of 66-80% (Table 8), and a WS of 37-47, substantially higher than the baselines (56% MAX and 37 WS). Language Sensitivity is slightly higher than zero in some models. Overall, results suggest the LMs do have some capability in this task, but it is hard to clearly determine it existed before fine-tuning.

Model	LEARNCURVE		LANGSENSE	
	WS	MAX	pert	nolang
RoBERTa-L	37	70	0	0
BERT-WWM	47	80	0	3
BERT-L	47	78	4	6
BERT-B	45	78	2	4
RoBERTa-B	40	68	0	2
XLNet	38	66	1	2
Baseline	37	56	0	0

Table 8: Results for the PROPERTY CONJUNCTION probe. Task example: *"What <u>has caffeine</u> and usu-<u>ally used for</u> waking up in morning?"*, Answer: coffee, Distractos: (coke, hearing music). The baseline row is the ESIM baseline.

Taxonomy conjunction A different operation is to find properties that are shared by two concepts. Specifically, we test whether LMs can find the mutual hypernym of a pair of concepts. For example, *"A germ and a human are both a type of [MASK]."*, where the answer is *"organism"*.
Probe Construction We use CONCEPTNET and WORDNET to find pairs of concepts and their hypernyms, keeping only pairs that frequently appear in the GOOGLE BOOK CORPUS. The example template is *"A `ENT-1` and a `ENT-2` are both a type of [MASK]."*, where `ENT-1` and `ENT-2` are replaced with entities that have a common hypernym, which is the gold answer. Distractors are concepts that are hypernyms of `ENT-1`, but not `ENT-2`, or vice versa. For evaluation, we keep all examples related to food and animal taxonomies, e.g., *"A beer and a ricotta are both a type of [MASK]."*, where the answer is *"food"* and the distractors are *"cheese"* and *"alcohol"*. For training, we use examples from different taxonomic trees, such that the concepts in the training and evaluation sets are disjoint.
Results Table 9 shows that models' zero-shot accuracy is substantially higher than random (33%) (Richardson et al., 2019), but overall even after fine-tuning accuracy is at most 57%. However, we do observe language sensitivity in the NO LAN-GUAGE and PERTRUBED LANGUAGE controls, suggesting that some models have pre-existing ca-

Next, we characterize when do the models err. A learning curve of the best performing model, ROBERTA-L is compared to controls in Figure 5C.

Model	Zero shot	MLP$_{MLM}$		LINEAR		LANGSENSE	
		WS	MAX	WS	MAX	pert	nolang
RoBERTa-L	45	49	56	45	46	0	9
BERT-WWM	47	48	52	47	48	0	17
BERT-L	45	47	51	45	46	6	9
BERT-B	45	46	52	45	47	3	10
RoBERTa-B	45	47	57	45	46	0	13
Baseline	36	34	51	-	-	0	1

Table 9: Results for the TAXONOMY CONJUNCTION probe. Task example: *"A beer and a ricotta are both a type of [MASK]."*, Answer: food, Distractos: (alcohol, cheese).

Analysis Analyzing the errors of ROBERTA-L, we found that a typical error is predicting for *"A crow and a horse are both a type of [MASK]."* that the answer is *"bird"*, rather than *"animal"*. Specifically, LMs prefer hypernyms that are closer in terms of edge distance on the taxonomy tree. Thus, a crow is first a bird, and then an animal. We find that when distractors are closer in the taxonomy tree to one of the entities in the statement than the gold answer, the models will consistently (80%) choose the distractor, ignoring the second entity in the phrase.

4.5 Can language models perform multi-hop reasoning?

Questions that require multi-hop reasoning, such as *"Who is the director of the movie about a WW2 pacific medic?"*, have recently drawn attention (Yang et al., 2018b; Welbl et al., 2017; Talmor and Berant, 2018) as a challenging task for contemporary models. But do pre-trained LMs have some internal mechanism to handle such questions?

To address this question, we create two probes, one for compositional question answering, and the other uses a multi-hop setup, building upon our observation (§3) that some LMs can compare ages.

Encyclopedic composition We construct questions such as *"When did the band where John Lennon played first form?"*. Because answers require more than one word-piece, we use the MC-QA setup.

Probe Construction We use the following three templates: (1) *"when did the band where ENT played first form?"*, (2) *"who is the spouse of the actor that played in ENT?"* and (3) *"where is the headquarters of the company that ENT established located?"*. We instantiate ENT using information from WIKIDATA (Vrandečić and Krőtzsch, 2014), choosing challenging distractors. For example, for template 1, the distractor will be a year close to the gold answer, and for template 3, it will be a city in the same country as the gold answer city.

To solve the complex question, the model must know all single-hop facts required for answering it. Thus, the model is first fine-tuned on all such facts (*"What company did Bill Gates establish? Microsoft"*, *"Where is the headquarters of Microsfot? Seattle"*) from the training and evaluation set, and then fine-tuned on multi-hop composition.

Results Results are summarized in Table 10. All models achieve low accuracy in this task, and the baseline performs best with a MAX of 54%. Language sensitivity of all models is small, suggesting that the LMs are unable to resolve compositional questions, but also struggle to learn it with some supervision. A learning curve of the best performing LM, BERT-WWM is compared to the controls in Figure 5D.

Model	LEARNCURVE		LANGSENSE	
	WS	MAX	pert	nolang
RoBERTa-L	42	50	0	2
BERT-WWM	47	53	1	4
BERT-L	45	51	1	4
BERT-B	43	48	0	3
RoBERTa-B	41	46	0	0
Baseline	49	54	3	0

Table 10: Results for the ENCYCLOPEDIC COMPOSITION probe. Task example: *"Who is the spouse of the actor that played in Friends & Lovers?"*, Answer: Matthew Vaughn, Distractors: (Candice King, Katy Perry). The baseline row is the ESIM baseline.

Multi-hop Comparison Multi-hop reasoning can be found in many common structures in natural language. In the phrase *"When comparing a 83 year old, a 63 year old and a 56 year old, the [MASK] is oldest"* one must first solve the question of which person is oldest, and then refer to its ordering: first, second or third.

Probe Construction We use the template above, treating the ages as arguments. The LM must choose between the candidates *"first"*, *"second"*, and *"third"*. Age arguments are in the same ranges as in AGE-COMPARISON.

Results All three possible answers appear in ROBERTA-L's top-10 zero-shot predictions, in-

	RoBERTa-L	BERT-WWM	BERT-L	RoBERTa-B	BERT-B
ALWAYS-NEVER					
AGE COMPARISON	✓	✓			
OBJECTS COMPARISON	✓	✗			
ANTONYM NEGATION	✓		✗	✗	
PROPERTY CONJUNCTION		✗			
TAXONOMY CONJUNCTION	✗	✗		✗	
ENCYC. COMPOSITION					
MULTI-HOP COMPARISON					

Table 11: The oLMpic games medals' table, summarizing per-task success. ✓ indicate the LM has achieved high accuracy considering controls and baselines, ✗ indicates partial success.

Model	Zero	MLP$_{MLM}$		LINEAR		LANGSENSE	
	shot	WS	MAX	WS	MAX	pert	nolang
RoBERTa-L	30	36	49	31	36	1	0
BERT-WWM	33	41	74	33	34	3	1
BERT-L	25	31	35	29	34	0	0
BERT-B	33	34	37	34	38	0	1
RoBERTa-B	33	32	40	32	34	0	1
Baseline	33	38	52	0	33	0	0

Table 12: Results for the COMPOSITIONAL COMPARISON probe. Task example: *"When comparing a 32 year old, a 24 year old and a 16 year old, the [MASK] is oldest"*, Answer: first, Distractors: (second, third).

dicating that the model sees the answers as viable choices. However, although successful in AGE-COMPARISON, the performance of ROBERTA-L is poor in this probe (Table 12), With zero-shot accuracy that is almost random, WS slightly above random, MAX lower than MLM-BASELINE (52%), and close to zero language sensitivity. All LMs seem to be learning the task during probing. Although BERT-WWM was able to partially solve the task with a MAX of 74% when approaching 4,000 training examples, the models do not appear to show multi-step capability in this task.

5 Medals

We summarize the results of the oLMpic Games in Table 11. For each task and LM, we summarize its success taking into account baseline models and controls. Interestingly, the table is mostly empty. The LMs, generally, have not been able to demonstrate strong pre-training capabilities in these symbolic reasoning tasks. BERT-WWM has shown partial success in a few tasks, whereas ROBERTA-L shows high performance in ALWAYS-NEVER, OBJECTS COMPARISON and ANTONYM NEGATION, and emerges as the most promising LM in these probes. However, when perturbed, ROBERTA-L has failed to demon-

strates consistent generalization and abstraction.

6 Discussion

In this work we presented eight different tasks for evaluating the reasoning abilities of models, alongside an evaluation protocol for disentangling pre-training from fine-tuning.

Interestingly, we found that even models that have identical structure and objective functions differ not only quantitatively but also qualitatively. Specifically, ROBERTA-L has shown reasoning abilities that are absent from other models. Thus, with enough data and careful optimization, models can acquire from a LM objective skills that might be surprising intuitively.

Another insight is that when current LMs succeed in a reasoning task, they do not seem to do so through abstraction and operation composition as humans perceive it. The abilities are strongly context-dependent, if ages are compared – then the numbers should be typical ages. Discrepancies from the training distribution quickly lead to large drops in performance. Last, the performance of LM in many reasoning tasks is poor.

Our work sheds some light on some of the blind spots of current LMs. We will release our code and data and hope that this work will help researchers evaluate the reasoning abilities of models, aid the design of new probes, and will guide future work on pre-training, objective functions and model design for endowing models with capabilities they are currently lacking.

References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2016. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. *arXiv preprint arXiv:1608.04207*.

Hessam Bagherinezhad, Hannaneh Hajishirzi, Yejin Choi, and Ali Farhadi. 2016. Are elephants bigger than butterflies? reasoning about sizes of objects. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72.

Léonard Blier and Yann Ollivier. 2018. The description length of deep learning models. In *Advances in Neural Information Processing Systems*, pages 2216–2226.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1657–1668, Vancouver, Canada. Association for Computational Linguistics.

Andy Coenen, Emily Reif, Ann Yuan, Been Kim, Adam Pearce, Fernanda Viégas, and Martin Wattenberg. 2019. Visualizing and measuring the geometry of bert. *arXiv preprint arXiv:1906.02715*.

J. Devlin, M. Chang, K. Lee, and K. Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Association for Computational Linguistics (NAACL)*.

Jesse Dodge, Suchin Gururangan, Dallas Card, Roy Schwartz, and Noah A Smith. 2019. Show your work: Improved reporting of experimental results. 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 2185–2194.

Yanai Elazar, Abhijit Mahabal, Deepak Ramachandran, Tania Bedrax-Weiss, and Dan Roth. 2019. How large are lions? inducing distributions over quantitative attributes. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3973–3983, Florence, Italy. Association for Computational Linguistics.

Allyson Ettinger. 2019. What bert is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *arXiv preprint arXiv:1907.13528*.

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

C. Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Maxwell Forbes and Yejin Choi. 2017. Verb physics: Relative physical knowledge of actions and objects. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 266–276.

Matt Gardner, Jonathan Berant, Hannaneh Hajishirzi, Alon Talmor, and Sewon Min. 2019. Question answering is a format; when is it useful? *arXiv preprint arXiv:1909.11291*.

Yoav Goldberg. 2019. Assessing bert's syntactic abilities. *arXiv preprint arXiv:1901.05287*.

Jonathan Gordon and Benjamin Van Durme. 2013. Reporting bias and knowledge acquisition. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 25–30. ACM.

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.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT*, pages 4129–4138.

Nora Kassner and Hinrich Schütze. 2019. Negated lama: Birds cannot fly. *arXiv preprint arXiv:1911.03343*.

Judy S Kim, Giulia V Elli, and Marina Bedny. 2019. Knowledge of animal appearance among sighted and blind adults. *Proceedings of the National Academy of Sciences*, 116(23):11213–11222.

Yongjie Lin, Yi Chern Tan, and Robert Frank. 2019. Open sesame: Getting inside bert's linguistic knowledge. *arXiv preprint arXiv:1906.01698*.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016a. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *TACL*, 4:521–535.

Tal Linzen, D. Emmanuel, and G. Yoav. 2016b. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics (TACL)*, 4.

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

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2019. Adversarial nli: A new benchmark for natural language understanding. *arXiv preprint arXiv:1910.14599*.

J. Pennington, R. Socher, and C. D. Manning. 2014. GloVe: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. 2018a. Deep contextualized word representations. In *North American Association for Computational Linguistics (NAACL)*.

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

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Kyle Richardson, Hai Hu, Lawrence S Moss, and Ashish Sabharwal. 2019. Probing natural language inference models through semantic fragments. *arXiv preprint arXiv:1909.07521*.

Ohad Rozen, Vered Shwartz, Roee Aharoni, and Ido Dagan. 2019. Diversify your datasets: Analyzing generalization via controlled variance in adversarial datasets. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 196–205.

Vered Shwartz and Ido Dagan. 2019. Still a pain in the neck: Evaluating text representations on lexical composition. In *Transactions of the Association for Computational Linguistics (TACL)*.

Robert Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *AAAI*, pages 4444–4451.

A. Talmor and J. Berant. 2018. The web as knowledge-base for answering complex questions. In *North American Association for Computational Linguistics (NAACL)*.

A. Talmor, J. Herzig, N. Lourie, and J. Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *North American Association for Computational Linguistics (NAACL)*.

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

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

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

D. Vrandečić and M. Krőtzsch. 2014. Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, 57.

Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. 2019. Do nlp models know numbers? probing numeracy in embeddings. 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 5310–5318.

Alex Warstadt, Yu Cao, Ioana Grosu, Wei Peng, Hagen Blix, Yining Nie, Anna Alsop, Shikha Bordia, Haokun Liu, Alicia Parrish, et al. 2019. Investigating bertâĂŹs knowledge of language: Five analysis methods with npis. 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 2870–2880.

J. Welbl, P. Stenetorp, and S. Riedel. 2017. Constructing datasets for multi-hop reading comprehension across documents. *arXiv preprint arXiv:1710.06481*.

Yiben Yang, Larry Birnbaum, Ji-Ping Wang, and Doug Downey. 2018a. Extracting commonsense properties from embeddings with limited human guidance. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 644–649.

Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. W. Cohen, R. Salakhutdinov, and C. D. Manning. 2018b. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

D. Yogatama, C. de M. d'Autume, J. Connor, T. Kocisky, M. Chrzanowski, L. Kong, A. Lazaridou, W. Ling, L. Yu, C. Dyer, et al. 2019. Learning and evaluating general linguistic intelligence. *arXiv preprint arXiv:1901.11373*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.