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 (LMs) have revolutionized the field of natural language processing in the last few years (Dai and Le, 2015; Peters et al., 2018a; Yang et al., 2019; Radford et al., 2019; Devlin et al., 2019). This has instigated research exploring what is captured by the contextualized representations that these LMs compute, revealing that they encode substantial amounts of syntax and semantics (Linzen et al., 2016b; Tenney et al., 2019b, a; Shwartz and Dagan, 2019; Lin et al., 2019; Coenen et al., 2019).

Despite these efforts, it remains unclear *what symbolic reasoning capabilities are difficult to learn from an 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 because 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.

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; Peters et al., 2018b; Warstadt 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 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

Transactions of the Association for Computational Linguistics, vol. 8, pp. 743–758, 2020. https://doi.org/10.1162/tacl_a_00342
Action Editor: Hinrich Schütze. Submission batch: 2/2020; Revision batch: 7/2020; Published 12/2020.

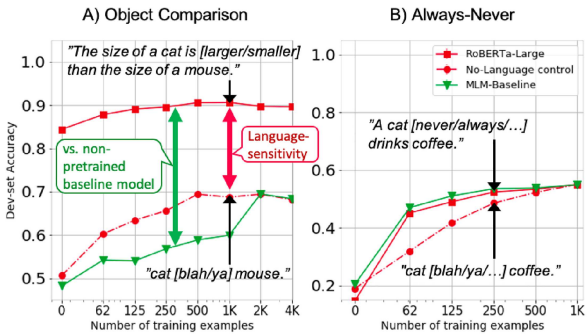

Figure 1: Overview of our experimental design. Two probes are evaluated using learning curves (including zero-shot). RoBERTa-L's (red squares, upper text in black) accuracy is compared with a No Language (No Lang.) control (red circles, lower text in black), and MLM-Baseline, which is not pre-trained (green triangles). 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 statement *''A cat is [MASK] than a mouse''*, an 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 probing task (which might be automatically generated, containing grammatical errors). Thus, we also compute the learning curve (Figure 1), by fine-tuning with increasing amounts of data on the already pre-trained masked language modeling (MLM) output ''head'', a 1-hidden layer multilayer perceptron (MLP) on top of the model's contextualized representations. A model that adapts from fewer examples arguably has better representations for it.

Moreover, to diagnose whether model performance is related to pre-training or fine-tuning, we add controls to every experiment (Figures 1, 2). First, we add a control that makes minimal use of language tokens, that is, *''cat [MASK] mouse''* (No Lang. in Figure 1). If a model succeeds given minimal use of language, the performance can be mostly attributed to fine-tuning rather than to the pre-trained language representations. Similar logic is used to compare against baselines that are not pre-trained (except for non-contextualized word embeddings). Overall, our setup provides a rich picture of whether LM representations help in solving a wide range of tasks.

We introduce eight tasks that test different types of reasoning, as shown in Table 1.[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 some reasoning tasks, such as comparing numbers, even in a zero-shot setup, whereas other models' performance is close to random. However, good performance is highly *context-dependent*. 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 when ages are specified through the birth year 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 with the controls.

Our contributions are summarized as follows:

- A set of probes that test whether specific reasoning skills are captured by pre-trained LMs.
- An evaluation protocol for understanding whether a capability is encoded in pre-trained representations or is learned during fine-tuning.
- An analysis of skills that current LMs possess. We find that LMs with similar architectures are qualitatively different, that their success is context-dependent, and that often all LMs fail.
- Code and infrastructure for designing and testing new probes on a large set of pre-trained LMs. The code and models are available at `http://github.com/alontalmor/oLMpics`.

[1] Average human accuracy was evaluated by two of the authors. Overall inter-annotator agreement accuracy was 92%.

Probe name	Setup	Example	Human[1]
ALWAYS-NEVER	MC-MLM	*A chicken [MASK] has horns.* **A.** never **B.** rarely **C.** sometimes **D.** often **E.** always	91%
AGE COMPARISON	MC-MLM	*A 21 year old person is [MASK] than me in age, If I am a 35 year old person.* **A.** younger **B.** older	100%
OBJECTS COMPARISON	MC-MLM	*The size of a airplane is [MASK] than the size of a house .* **A.** larger **B.** smaller	100%
ANTONYM NEGATION	MC-MLM	*It was [MASK] hot, it was really cold .* **A.** not **B.** really	90%
PROPERTY CONJUNCTION	MC-QA	*What is usually located at hand and used for writing?* **A.** pen **B.** spoon **C.** computer	92%
TAXONOMY CONJUNCTION	MC-MLM	*A ferry and a floatplane are both a type of [MASK].* **A.** vehicle **B.** airplane **C.** boat	85%
ENCYC. COMPOSITION	MC-QA	*When did the band where Junior Cony played first form?* **A.** 1978 **B.** 1977 **C.** 1980	85%
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	100%

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

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 $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$, and compute contextualized representations with a pre-trained LM, that is, $\boldsymbol{h} = \text{ENCODE}(\boldsymbol{x}) = (\boldsymbol{h}_1, \ldots, \boldsymbol{h}_n)$. Specifically, we consider: (a) BERT (Devlin et al., 2019), a pre-trained LM built using the Transformer (Vaswani et al., 2017) architecture, which consists of a stack of Transformer layers, where each layer includes a multi-head attention sublayer and a feed-forward sub-layer. BERT is trained on large corpora using the MLM, that is, the model is trained to predict words that are masked from the input; including BERT-WHOLE-WORD-MASKING (BERT-WWM), which was trained using *whole-word-masking*; (b) ROBERTA (Liu et al., 2019), which has the same architecture as BERT, but was trained on 10x more data and optimized carefully.

2.2 Probing Setups

We probe the pre-trained LMs using two setups: multichoice MLM (MC-MLM) and multichoice question answering (MC-QA). The default setup is MC-MLM, 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.[2] The MC-QA setup is used when the answer set substantially varies between questions, and many of the answers have more than one word piece.

[2]Vocabularies of LMs such as BERT and ROBERTA contain *word-pieces*, which are sub-word units that are frequent in the training corpus. For details see Sennrich et al. (2016).

MC-MLM Here, we convert the MLM setup to a multichoice setup (MC-MLM). Specifically, the input to the LM is the sequence $\boldsymbol{x} = (\texttt{[CLS]}, \ldots, \boldsymbol{x}_{i-1}, \texttt{[MASK]}, \boldsymbol{x}_{i+1}, \ldots, \texttt{[SEP]})$, where a single token $\boldsymbol{x}_i$ is masked. Then, the contextualized representation $\boldsymbol{h}_i$ is passed through a MC-MLM *head* where $\mathcal{V}$ is the vocabulary, and FF_{MLM} is a 1-hidden layer MLP:

$$l = FF_{\text{MLM}}(\boldsymbol{h}_i) \in \mathbb{R}^{|\mathcal{V}|}, p = \text{softmax}(m \oplus l),$$

where $\oplus$ is element-wise addition and $m \in \{0, -\infty\}^{|\mathcal{V}|}$ is a mask that guarantees that the support of the probability distribution will be over exactly $K \in \{2, 3, 4, 5\}$ candidate tokens: the correct one and $K - 1$ distractors. Training minimizes cross-entropy loss given the gold masked token. An input, e.g., *''[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 (Petroni et al., 2019).

MC-QA Constructing a MC-MLM probe limits the answer candidates to a single token from the word-piece vocabulary. To relax this we use in two tasks the standard setup for answering multichoice questions with pre-trained LMs (Talmor et al., 2019; Mihaylov et al., 2018). Given a question $\boldsymbol{q}$ and candidate answers $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_K$, we compute for each candidate answer $\boldsymbol{a}_k$ representations $\boldsymbol{h}^{(k)}$ from the input tokens *''[CLS]* $\boldsymbol{q}$ *[SEP]* $\boldsymbol{a}_k$ *[SEP]''*. Then the probability over answers is obtained using the *multichoice QA head*:

$$l^{(k)} = FF_{\text{QA}}(\boldsymbol{h}_1^{(k)}), p = \text{softmax}(l^{(1)}, \ldots, l^{(K)}),$$

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 $FF_{\text{MLM}}(\cdot)$ is the hidden representation $\boldsymbol{h} \in \mathbb{R}^{1024}$ (for large models). To obtain a similar architecture with 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_{MLM}, 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 multichoice setup using the procedure proposed by Zellers et al. (2018). Each phrase and candidate answer are passed as a list of token '`[CLS] phrase [SEP] answer [SEP]`' to the LM. The contextualized representation of the `[CLS]` token is linearly projected to a single logit. The logits for candidate answers are passed through a softmax layer to obtain probabilities, and the argmax is selected as the model prediction.

3 Controlled Experiments

We now describe the experimental design and controls used to interpret the results. We use the Age-Compare task as a running example, where models need to compare the numeric value of ages.

3.1 Zero-shot Experiments with MC-MLM

Fine-tuning pre-trained LMs makes it hard to disentangle what is captured by the original representations and what was learned during fine-tuning. Thus, ideally, one should test LMs using the pre-trained weights *without* fine-tuning (Linzen et al., 2016a; Goldberg, 2019). The MC-MLM setup, which uses a pre-trained MLM head, achieves exactly that. One only needs to design the task as a statement with a single masked token and K possible output tokens. For example, in Age-Compare, we chose the phrasing *''A `AGE-1` year old person is `[MASK]` than me in age, If I am a `AGE-2` year old person.''*, where `AGE-1` and `AGE-2` are replaced with different integers, and possible answers are *''younger''* and *''older''*. Otherwise, no training is needed, and the original representations are tested.

Figure 2A 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 (acc.) is measured as the proportion of cases when the model output is correct. The performance of BERT-WWM, is on the left (blue), and of RoBERTa-L on the right (green). The results in Figure 2A and Table 2 show that RoBERTa-L compares numbers correctly (98% acc.), BERT-WWM achieves higher than random acc. (70% acc.), while BERT-L is random (50% acc.). The performance of MLM-Baseline is also random, as the MLP_{MLM} weights are randomly initialized.

We note that picking the statement for each task was done through manual experimentation. We tried multiple phrasings (Jiang et al., 2019) and chose the one that achieves highest average zero-shot accuracy across all tested LMs.

A case in point ...

Thus, if a model performs well, one can infer that it has the tested reasoning skill. However, failure does not entail that the reasoning skill is missing, as it is possible that there is a problem with the lexical-syntactic construction we picked.

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 (Jiang et al., 2019).

To tackle this, we fine-tune models with a small number of examples. We assume that if the LM representations are useful for a task, it will require few examples to overcome the language

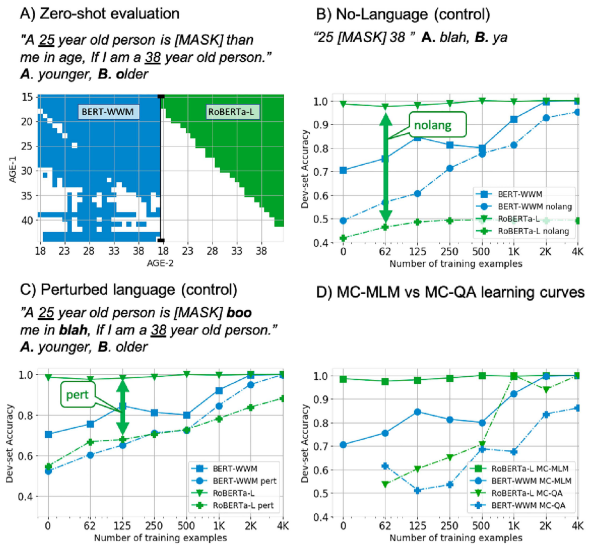

Figure 2: An illustration of our evaluation protocol. We compare RoBERTa-L (green) and BERT-WWM (blue), controls are in dashed lines and markers are described in the legends. Zero-shot evaluation on the top left, AGE−1 is *''younger''* (in color) vs. *''older''* (in white) than AGE−2.

mismatch and achieve high performance. In most cases, we train with $N \in \{62, 125, 250, 500, 1K, 2K, 4K\}$ examples. To account for optimization instabilities, we fine-tune several times with different seeds, and report average accuracy across seeds. The representations $\boldsymbol{h}$ are fixed during fine-tuning, and we only fine-tune the parameters of MLP_{MLM}.

Evaluation and Learning-curve Metrics Learning curves are informative, but inspecting many learning curves can be difficult. Thus, we summarize them using two aggregate statistics. We report: (a) MAX, that is, the maximal accuracy on the learning curve, used to estimate how well the model can handle the task given the limited amount of examples. (b) The metric WS, which is a weighted average of accuracies across the learning curve, where higher weights are given to points where N is small.[3] WS is related to the area under the accuracy curve, and to the online code metric, proposed by Yogatama et al. (2019) and Blier and Ollivier (2018). The linearly decreasing weights emphasizes our focus on performance given little training data, as it highlights what was encoded by the model *before* fine-tuning.

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

Model	Zero	MLP_{MLM}		LINEAR		LANGSENSE	
	shot	WS	MAX	WS	MAX	pert	nolang
RoBERTa-L	98	98	100	97	100	31	51
BERT-WWM	70	82	100	69	85	13	15
BERT-L	50	52	57	50	51	1	0
RoBERTa-B	68	75	91	69	84	24	25
BERT-B	49	49	50	50	50	0	0
Baseline	49	58	79	-	-	0	0

Table 2: AGE-COMPARE results. Accuracy over two answer candidates (random is 50%). LANGSENSE are the Language Sensitivity controls, pert is PERTURBED LANG. and nolang is NO LANG. The baseline row is MLM-BASELINE.

For AGE-COMPARE, the solid lines in Figure 2B illustrate the learning curves of RoBERTA-L and BERT-WWM, 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.

3.3 Controls

Comparing learning curves tells us which model learns from fewer examples. However, because highly parameterized MLPs, as used in LMs, can approximate a wide range of functions, it is difficult to determine whether 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 reason over language examples. Thus, a natural control is to present the reasoning task *without* language and inspect performance. 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 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 behavior. We propose two control tasks:

No Language *control* We remove all input tokens, except for `[MASK]` and the *arguments* of the task, namely, the tokens that are necessary for computing the output. In Age-Compare, an example is reduced to the phrase *''24 [MASK] 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 2B illustrate the learning curves in No Lang.: RoBERTa-L (green) shows high language sensitivity, while BERT-WWM (blue) has lower language sensitivity. This suggests it handles this task partially 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 No Lang. (rounding negative numbers to zero). For RoBERTa-L the value is 51, because RoBERTa-L gets almost 100% acc. in the presence of language, and is random (50% acc.) without language.

Perturbed Language *control* A more targeted language control, is to replace words that are central for the reasoning task with nonsense words. Specifically, we pick key words in each probe template, and replace these words by randomly sampling from a list of 10 words that carry relatively limited meaning.[4] For example, in Property Conjunction, we can replace the word *''and''* with the word *''blah''* to get the example *''What is located at hand **blah** used for writing?''*. If the learning curve of Perturbed Lang. is similar to the original example, then 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. For example, in Age-Compare, the targeted words are *''age''* and *''than''*, resulting in examples like *''A `AGE-1` year old person is [MASK] **blah** me in **da**, If i am a `AGE-2` year old person.''* Figure 2C shows the learning curves for RoBERTa-L and BERT-WWM, where solid lines corresponds to the original examples and dashed lines are the Perturbed Lang. control. Despite this minor perturbation, the performance of RoBERTa-L substantially decreases, implying that the model needs the input. Conversely, BERT-WWM performance decreases only moderately.

Does a linear transformation suffice? In MC-MLM, the representations $\boldsymbol{h}$ are fixed, and only the pre-trained parameters of MLP_{MLM} are fine-tuned. As a proxy for measuring ''how far" the representations are 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 $\boldsymbol{h}$ is required. Table 2 shows the performance of this setup (Linear), compared with MLP_{MLM}.

Why is MC-MLM preferred over MC-QA? Figure 2D compares the learning curves of MC-MLM and MC-QA in Age-Compare. Because in MC-QA, the network MLP_{QA} cannot be initialized by pre-trained weights, zero-shot evaluation is not meaningful, 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 move to describe the research questions and various probes used to answer these questions. For each task we describe how it was constructed, show results via a table as described in the controls section, and present an analysis.

Our probes are mostly targeted towards symbolic reasoning skills (Table 1). We examine the ability of language models to compare numbers, to understand whether an object has a conjunction of properties, to perform multi-hop composition of facts, among others. However, since we generate examples automatically from existing resources, some probes also require background knowledge, such as sizes of objects. Moreover, as explained in §3.1, we test models on a manually-picked phrasing that might interact with the language abilities of the model. Thus, when a model succeeds this is evidence that it has the necessary skill, but failure could be attributed to issues with background knowledge and linguistic abilities as well. In each probe, we will explicitly mention what knowledge and language abilities are necessary.

[4]The list of substitutions is: *''blah''*, *''ya''*, *''foo''*, *''snap''*, *''woo''*, *''boo''*, *''da''*, *''wee''*, *''foe''* and *''fee''*.

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-COMPARE task, in which 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 acc. of 98% in AGE-COMPARE. But is it robust? We test this using perturbations to the task and present the results in Figure 3. Figure 3A 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 3B shows that it correctly flips *''younger''* to *''older''* (76% acc.), reasoning that a person born in 1980 is older than one born in 2000.

However, when evaluated on the exact same statement, but with values corresponding to typical *ages* instead of years (Figure 3D), ROBERTA-L obtains an acc. of 12%, consistently outputting the opposite prediction. With ages as values and not years, it seems to disregard the language, performing the comparison based on the values only. We will revisit this tendency in §4.4.

Symmetrically, Figure 3C shows results when numeric values of ages are swapped with typical years of birth. ROBERTA-L is unable to handle this, always predicting *''older''*.[5] 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; Pezzelle and Fernández, 2019). Can LMs do the same? ***Probe Construction*** We construct statements of 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 from a list of objects from two domains: animals (e.g. *''camel''*) and general objects (e.g. *''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 in DoQ (Elazar et al., 2019), and then manually fix any errors. This probe requires prior knowledge of object sizes and understanding of a comparative language construction. 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% acc. 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

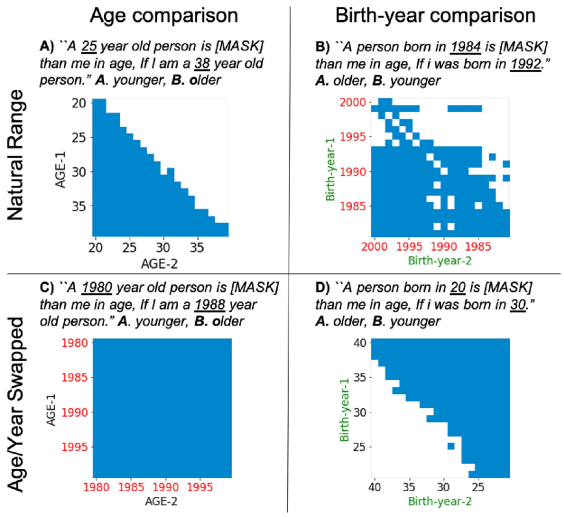

Figure 3: AGE COMPARISON perturbations. Left side graphs are age-comparison, right side graphs are age comparison by birth-year. In the bottom row, the values of ages are swapped with birth-years and vice versa. In blue pixels the model predicts *''older''*, in white *''younger''*. (A) is the correct answer.

[5] We observed that in neutral contexts models have a slight preference for *''older''* over *''younger''*, which could potentially explain this result.

Model	Zero	MLP_{MLM}		Linear		LangSense	
	shot	WS	Max	WS	Max	pert	nolang
RoBERTa-L	84	88	91	86	90	22	26
BERT-WWM	55	65	81	63	77	9	9
BERT-L	52	56	66	53	56	5	4
BERT-B	56	55	72	53	56	2	3
RoBERTa-B	50	61	74	57	66	8	0
Baseline	46	57	74	-	-	2	1

Table 3: Results for the Objects Comparison probe. Accuracy over two answer candidates (random is 50%).

	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: RoBERTa-L Zero-shot Size comp. predictions.

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 when we compare objects to *''sun''*, mostly emitting *''larger''*, ignoring the rest of the statement.

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

Adverbial modifiers such as *''always''*, *''sometimes''*, or *''never''*, tell us about the quantity or frequency of events (Lewis, 1975; Barwise and Cooper, 1981). Anecdotally, when RoBERTa-L predicts a completion for the phrase *''Cats usually drink [MASK].''*, the top completion is *''coffee''*, a frequent drink in the literature it was trained on, rather then *''water''*. However, humans know that *''Cats NEVER drink coffee''*. Prior work explored retrieving the correct quantifier for a statement (Herbelot and Vecchi, 2015; Wang et al., 2017). Here we adapt this task to a masked language model.

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 must know the frequency of an event, and map the appropriate adverbial modifier to that representation. Linguistically, the task tests how well the model predicts frequency quantifiers (or adverbs) modifying predicates in different statements (Lepore and Ludwig, 2007).

Model	Zero	MLP_{MLM}		Linear		LangSense	
	shot	WS	Max	WS	Max	pert	nolang
RoBERTa-L	14	44	55	26	41	3	5
BERT-WWM	10	46	57	32	52	2	3
BERT-L	22	45	55	36	50	3	8
BERT-B	11	44	56	30	52	3	8
RoBERTa-B	15	43	53	25	44	2	6
Baseline	20	46	56	-	-	1	2

Table 5: Results for the Always-Never probe. Accuracy over five answer candidates (random is 20%).

Probe Construction We manually craft templates that contain one slot for a `subject` and another for an `object`, e.g., *''`FOOD-TYPE` is `[MASK]` part of a `ANIMAL`'s diet.''* (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''*.[6] We collected 1,300 examples with 1,000 used for training and 300 for evaluation.

We note that some examples in this probe are similar to Objects Comparison (line 4 in Table 5). However, the model must also determine if sizes can be overlapping, which is the case in 56% of the examples.

Results Table 5 shows the results, where random accuracy is 20%, and majority vote accuracy is 35.5%. In the zero-shot setup, acc. is less than random. In the MLP_{MLM} and Linear setup acc. reaches a maximum of 57% in BERT-L, but

[6]The class distribution over the answers is *''never''*: 24%, *''rarely''*: 10%, *''sometimes''*: 34%, *''often''*: 7%, and *''always''*: 23%.

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

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

MLM-BASELINE obtains similar acc., 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 PERTURBED LANG. control and slightly drops in the NO LANG. control. Figure 1B compares the learning curve of ROBERTA-L with controls. MLM-BASELINE consistently outperforms ROBERTA-L, which display only minor language sensitivity, suggesting that pre-training is not effective for solving this task.

Analysis We generated predictions from the best model, BERT-WWM, and show analysis results in Table 6. For reference, we only selected examples where human majority vote led to the correct answer, and thus the majority vote is near 100% on these examples. 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 role in the inability to correctly determine that *''A rhinoceros NEVER has fur.''* Interestingly, behavioral research conducted on blind humans shows they exhibit a similar bias (Kim et al., 2019).

4.3 Do LMs Capture Negation?

Ideally, the presence of the word *''not''* should affect the prediction of a masked token. However, Several recent works have shown that LMs do not take into account the presence of negation in sentences (Ettinger, 2019; Nie et al., 2020; Kassner and Schütze, 2020). Here, we add to this literature, by probing whether LMs can properly use negation in the context of *synonyms* vs. *antonyms*.

Model	Zero	MLP_{MLM}		LINEAR		LANGSENSE	
	shot	WS	MAX	WS	MAX	pert	nolang
RoBERTa-L	75	85	91	77	84	14	21
BERT-WWM	57	70	81	61	73	5	6
BERT-L	51	70	82	58	74	5	9
BERT-B	52	68	81	59	74	2	9
RoBERTa-B	57	74	87	63	78	10	16
Baseline	47	67	80	-	-	0	0

Table 7: Results for the ANTONYM NEGATION probe. Accuracy over two answer candidates (random is 50%).

Do LMs Capture the Semantics of 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 very* ***rapid****''*, the LM should choose a word like *''very''* in the presence of the synonyms *''fast''* and *''rapid''*. An LM that correctly distinguishes between *''not''* and *''very''*, demonstrates knowledge of the taxonomic relations as well as the ability to reason about the usage of negation 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. Linguistically, we test whether the model appropriately predicts a negation vs. intensification adverb based on synonymy/antonymy relations between nouns, adjectives and verbs.

Results ROBERTA-L shows higher than chance acc. of 75% in the zero-shot setting, as well as high Language Sensitivity (Table 7). MLM-BASELINE, equipped with GloVe word embeddings, is able to reach a comparable WS of 67 and MAX of 80%, suggesting they do not have a large advantage on this task.

4.4 Can 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,`

Model	LearnCurve		LangSense	
	WS	Max	pert	nolang
RoBERTa-L	49	87	2	4
BERT-WWM	46	80	0	1
BERT-L	48	75	2	5
BERT-B	47	71	2	1
RoBERTa-B	40	57	0	0
Baseline	39	49	0	0

Table 8: Results for the Property Conjunction probe. Accuracy over three answer candidates (random is 33%).

predicate, object) triples. We use ConceptNet 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 To construct an example, we first choose a concept that has two properties in ConceptNet, where a property is a (predicate, object) pair. For example, stop 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 ConceptNet predicates to natural language questions. We split examples such that concepts in training and evaluation are disjoint. This linguistic structure tests whether the LM can answer questions with conjoined predicates, requiring world knowledge of object and relations.

Results In MC-QA, we fine-tune the entire network and do not freeze any representations. Zero-shot cannot be applied because the weights of MLP_{QA} are untrained. All LMs consistently improve as the number of examples increases, reaching a Max of 57% to 87% (Table 8). The high Max results suggest that the LMs generally have the required pre-existing knowledge. The WS of most models is slightly higher than the baselines (49% Max and 39 WS). Language Sensitivity is slightly higher than zero in some models. Overall, results suggest the LMs do have some capability in this task, but proximity to baseline results, and low language selectivity make it hard to clearly determine whether it existed before fine-tuning.

To further validate our findings, we construct a parallel version of our data, where we replace the word *''and''* by the phrase *''but not''*. In this version, the correct answer is the first distractor in the original experiment, where one property holds and the other does not. Overall, we observe a similar trend (with an increase in performance across all models): Max results are high (79-96%), pointing that the LMs hold the relevant information, but improvement over ESIM-Baseline and language sensitivity are low. For brevity, we omit the detailed numerical results.

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, for example, *''A beer and a ricotta are both a type of [MASK].''*, where the answer is *''food''* and the distractors are *''cheese''* and *''alcohol''*. This phrasing requires the model to handle conjoined co-hyponyms in the subject position, based on lexical relations of hyponymy / hypernymy between nouns. 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 acc. is substantially higher than random (33%), but overall even after fine-tuning acc. is at most 59%. However, the No Lang. control shows some language sensitivity, suggesting that some models have pre-existing capabilities.

Model	Zero	MLP_{MLM}		Linear		LangSense	
	shot	WS	Max	WS	Max	pert	nolang
RoBERTa-L	45	50	56	45	46	0	3
BERT-WWM	46	48	52	46	46	0	7
BERT-L	53	54	57	53	54	0	15
BERT-B	47	48	50	47	47	0	12
RoBERTa-B	46	50	59	47	49	0	18
Baseline	33	33	47	-	-	1	2

Table 9: Results for the Taxonomy Conjunction probe. Accuracy over three answer candidates (random is 33%).

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
ESIM-Baseline	49	54	3	0

Table 10: Results for Encyclopedic composition. Accuracy over three answer candidates (random is 33%).

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 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 LMs do 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., 2018; 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?''*. Here answers require multiple tokens, thus 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 Krötzsch, 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. This linguistic structure introduces a (restrictive) relative clauses that requires a) Correctly resolving the reference of the noun modified by the relative clause, b) Answering the full question subsequently.

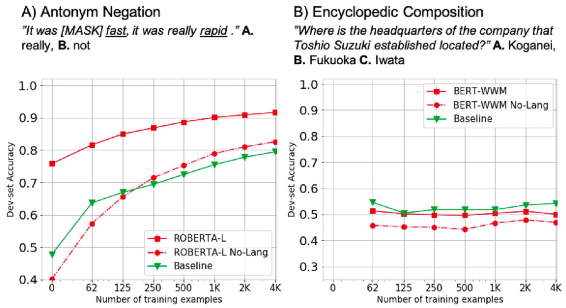

Figure 4: Learning curves in two tasks. For each task, the best performing LM is shown alongside the No Lang. control and baseline model. (A) is the correct answer.

To solve the question, the model must have knowledge of all single-hop encyclopedic facts required for answering it. Thus, we first fine-tune the model on all such facts (e.g., *''What company did Bill Gates establish? Microsoft''*) from the training and evaluation set, and then fine-tune on multi-hop composition.

Results Results are summarized in Table 10. All models achieve low acc. in this task, and the baseline performs best with a Max of 54%. Language sensitivity of all models is small, and MLM-Baseline performs slightly better (Figure 4B), suggesting that the LMs are unable to resolve compositional questions, but also struggle to learn it with some supervision.

Multi-hop Comparison Multi-hop reasoning can be found in many common structures in natural language. In the phrase *''When comparing*

Model	Zero	MLP_{MLM}		Linear		LangSense	
	shot	WS	Max	WS	Max	pert	nolang
RoBERTa-L	29	36	49	31	41	2	2
BERT-WWM	33	41	65	32	36	6	4
BERT-L	33	32	35	31	34	0	3
BERT-B	32	33	35	33	35	0	2
RoBERTa-B	33	32	40	29	33	0	0
Baseline	34	35	48	-	-	1	0

Table 11: Results for Compositional Comparison. Accuracy over three answer candidates (random is 33%).

	RoBERTa Large	BERT WWM	BERT Large	RoBERTa Base	BERT Base
Always-Never					
Age Comparison	✓	✓		✓̸	
Objects Compar.	✓	✓̸			
Antonym Neg.	✓		✓̸	✓̸	
Property Conj.	✓̸	✓̸			
Taxonomy Conj.	✓̸	✓̸		✓̸	
Encyc. Comp.					
Multi-hop Comp.					

Table 12: The oLMpic games medals, summarizing per-task success. ✓ indicate the LM has achieved high accuracy considering controls and baselines, ✓̸ indicates partial success.

a 83 year old, a 63 year old and a 56 year old, the [MASK] is oldest'' one must find the oldest person, then refer to its ordering: first, second, or third.

Probe Construction We use the template above, treating the ages as arguments, and *''first''*, *''second''*, and *''third''* as answers. Age arguments are in the same ranges as in Age-Compare. Linguistically, the task requires predicting the subject of sentences whose predicate is in a superlative form, where the relevant information is contained in a ''when''-clause. The sentence also contains nominal ellipsis, also known as fused-heads (Elazar and Goldberg, 2019).

Results All three possible answers appear in RoBERTa-L's top-10 zero-shot predictions, indicating that the model sees the answers as viable choices. Although successful in Age-Compare, the performance of RoBERTa-L is poor in this probe (Table 11), With zero-shot acc. that is almost random, WS slightly above random, Max lower than MLM-Baseline (48%), 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 65% 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 12. Generally, the LMs did not demonstrate strong pre-training capabilities in these symbolic reasoning tasks. BERT-WWM showed partial success in a few tasks, whereas RoBERTa-L showed high performance in Always-Never, Objects Comparison and Antonym Negation, and emerges as the most promising LM. However, when perturbed, RoBERTa-L has failed to demonstrates consistent generalization and abstraction.

Analysis of correlation with pre-training data A possible hypothesis for why a particular model is successful in a particular task might be that the language of a probe is more common in the corpus it was pre-trained on. To check that, we compute the unigram distribution over the training corpus of both BERT and RoBERTa. We then compute the average log probability of the development set under these two unigram distributions for each task (taking into account only content words). Finally, we compute the correlation between which model performs better on a probe (RoBERTa-L vs. BERT-WWM) and which training corpus induces higher average log probability on that probe. We find that the Spearman correlation is 0.22, hinting that the unigram distributions do not fully explain the difference in performance.

6 Discussion

We presented eight different tasks for evaluating the reasoning abilities of models, alongside an evaluation protocol for disentangling pre-training from fine-tuning. 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 appropriate data and optimization, models can acquire from an LM objective skills that might be surprising intuitively.

However, when current LMs succeed in a reasoning task, they do not do so through abstraction and composition as humans perceive it. The abilities are context-dependent, if ages are compared–then the numbers should be typical ages. Discrepancies from the training distribution

lead to large drops in performance. Last, the performance of LM in many reasoning tasks is poor.

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

Acknowledgments

This work was completed in partial fulfillment for the PhD degree of the first author. We thank our colleagues at The Allen Institute of AI, especially Kyle Richardson, Asaf Amrami, Mor Pipek, Myle Ott, Hillel Taub-Tabib, and Reut Tsarfaty. This research was partially supported by The Israel Science Foundation grant 942/16, The Blavatnik Computer Science Research Fund and The Yandex Initiative for Machine Learning, and the European Union's Seventh Framework Programme (FP7) under grant agreements no. 802774-ERC-iEXTRACT and no. 802800-DELPHI.

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*.

Jon Barwise and Robin Cooper. 1981. Generalized quantifiers and natural language, *Philosophy, language, and artificial intelligence*, Springer, pages 241–301. **DOI:** https://doi.org/10.1007/978-94-009-2727-8_10

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72. **DOI:** https://doi.org/10.1162/tacl_a_00254

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. **DOI:** https://doi.org/10.18653/v1/P17-1152

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*.

Andrew M. Dai and Quoc V. Le. 2015, Semi-supervised sequence learning, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 3079–3087. Curran Associates, Inc.,

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)*.

Yanai Elazar and Yoav Goldberg. 2019. Wheres my head? definition, data set, and models for numeric fused-head identification and resolution. *Transactions of the Association for Computational Linguistics*, 7:519–535. **DOI:** https://doi.org/10.1162/tacl_a_00280

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. **DOI:** https://doi.org/10.18653/v1/P19-1388

Allyson Ettinger. 2019. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *arXiv preprint arXiv:1907.13528*. **DOI:** https://doi.org/10.1162/tacl_a_00298

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. **DOI:** https://doi.org/10.18653/v1/W16-2524

C. Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press. **DOI:** https://doi.org/10.7551/mitpress/7287.001.0001

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. **DOI:** https://doi.org/10.18653/v1/P17-1025

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.

Aurélie Herbelot and Eva Maria Vecchi. 2015. Building a shared world: Mapping distributional to model-theoretic semantic spaces. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 22–32. **DOI:** https://doi.org/10.18653/v1/D15-1003

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. **DOI:** https://doi.org/10.18653/v1/D19-1275

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.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2019. How can we know what language models know? *arXiv preprint arXiv:1911.12543*. **DOI:** https://doi.org/10.1162/tacl_a_00324

Nora Kassner and Hinrich Schütze. 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7811–7818, Online. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/2020.acl-main.698

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. **DOI:** https://doi.org/10.1073/pnas.1900952116, **PMID:** 31113884, **PMCID:** PMC6561279

Ernest Lepore and Kirk Ludwig. 2007. *Donald Davidson's truth-theoretic semantics*. Oxford University Press. **DOI:** https://doi.org/10.1093/acprof:oso/9780199290932.001.0001

David Lewis. 1975. Adverbs of quantification. *Formal semantics-the essential readings,* 178:188. **DOI:** https://doi.org/10.1002/9780470758335.ch7

Yongjie Lin, Yi Chern Tan, and Robert Frank. 2019. Open sesame: Getting inside BERTs linguistic knowledge. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 241–253.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016a. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *TACL*, 4:521–535. **DOI:** https://doi.org/10.1162/tacl_a_00115

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. **DOI:** https://doi.org/10.1162/tacl_a_00115

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. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

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. **DOI:** https://doi.org/10.3115/v1/D14-1162

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)*. **DOI:** https://doi.org/10.18653/v1/N18-1202

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. **DOI:** https://doi.org/10.18653/v1/D18-1179

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 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)*, pages 2463–2473. **DOI:** https://doi.org/10.18653/v1/D19-1250

Sandro Pezzelle and Raquel Fernández. 2019. Is the red square big? malevic: Modeling adjectives leveraging visual contexts. 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 2858–2869. **DOI:** https://doi.org/10.18653/v1/D19-1285

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

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. **DOI:** https://doi.org/10.18653/v1/K19-1019

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725. **DOI:** https://doi.org/10.18653/v1/P16-1162

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)*. **DOI:** https://doi.org/10.1162/tacl_a_00277

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-First AAAI Conference on Artificial Intelligence*.

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, Jul. 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*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

D. Vrandečić and M. Krőtzsch. 2014. Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, 57. **DOI:** https://doi.org/10.1145/2629489

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. **DOI:** https://doi.org/10.18653/v1/D19-1534

Mingzhe Wang, Yihe Tang, Jian Wang, and Jia Deng. 2017. Premise selection for theorem proving by deep graph embedding, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 2786–2796. Curran Associates, Inc.

Alex Warstadt, Yu Cao, Ioana Grosu, Wei Peng, Hagen Blix, Yining Nie, Anna Alsop, Shikha Bordia, Haokun Liu, Alicia Parrish, Sheng-Fu Wang, Jason Phang, Anhad Mohananey, Phu Mon Htut, Paloma Jeretič, and Samuel R. Bowman. 2019. Investigating BERTs 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. **DOI:** https://doi.org/10.18653/v1/D19-1286

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association for Computational Linguistics*, 6287–302. **DOI:** https://doi.org/10.1162/tacl_a_00021

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. **DOI:** https://doi.org/10.18653/v1/P18-2102

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)*. **DOI:** https://doi.org/10.18653/v1/D18-1259, **PMCID:** PMC6156886

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R. Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5753–5763.

D. Yogatama, C. de M. d'Autume, J. Connor, T. Kocisky, M. Chrzanowski, L. Kong, A. Lazaridou, W. Ling, L. Yu, C. Dyer, and Phil Blunson. 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)*. **DOI:** https://doi.org/10.18653/v1/D18-1009