Life after BERT: What do Other Muppets Understand about Language?

Anonymous ACL submission

Abstract

Pre-trained transformers are at the core of natural language processing today. However, the understanding of what model learns during pre-training is still limited. Existing model analysis works usually focus only on one or two model families at a time, overlooking the variety of existing architectures and pre-training objectives. In our work, we utilize the oLMpics benchmark and psycholinguistic probing datasets for a diverse set of 28 models including T5, BART, and ALBERT. Additionally, we adapt the oLMpics zero-shot setup for autoregressive models and evaluate GPT networks of different sizes. Our findings show that none of these models can resolve compositional questions in a zero-shot fashion, suggesting that this skill is not learnable using existing pre-training objectives. Additionally, we find that global model decisions such as architecture, directionality, size of the dataset, and pre-training objective are not predictive of a model's linguistic capabilities.

1 Introduction

After the initial success of pre-training in natural language processing (Howard and Ruder, 2018; Peters et al., 2018), the number of pre-trained models in NLP has increased dramatically (Radford and Narasimhan, 2018; Devlin et al., 2018; Lewis et al., 2019; Liu et al., 2019b; Raffel et al., 2019; Lan et al., 2019; Dong et al., 2019). However, there is a limited understanding of why certain models perform better than others and what linguistic capabilities they acquire through pre-training.

While a lot of work has been done to evaluate these models on general natural language understanding datasets (Wang et al., 2018, 2019; Lai et al., 2017), such datasets do not allow researchers to identify the specific linguistic capabilities of a model. Furthermore, models are fine-tuned on these tasks, with their final performance resulting from a combination of pre-trained knowledge and task-specific information learned through fine-tuning.

Probing tasks (Talmor et al., 2019; Zagoury et al., 2021; McCoy et al., 2019; Goldberg, 2019) give a promising solution to this problem, as they evaluate specific capabilities of pre-trained models and in many designed for zero-shot evaluation, which reveals the knowledge that models have actually learned purely through pre-training.

Currently, most in-depth analysis studies focus on one or two model families. In many cases, analysis papers only probe BERT and similar models (Ettinger, 2020; Kobayashi et al., 2020; Garí Soler and Apidianaki, 2020; Ravichander et al., 2020; Zagoury et al., 2021; Kassner et al., 2020; Mohebbi et al., 2021; Clark et al., 2020; Liu et al., 2021). Fortunately, this trend is changing and now we see more papers that probe models such as ALBERT, T5 or BART (Mosbach et al., 2020; Phang et al., 2021; Jiang et al., 2021). However, only a small number of analysis papers have probed multiple (three or more) model families (Zhou et al., 2021; Ilharco et al., 2021).

In our work, we test 8 families of models on the oLMpics tasks (Talmor et al., 2019) and 6 families of models on psycholinguistic tasks from Ettinger (2020). These models differ in size, architecture, pre-training objective, dataset size, and have other small yet important differences. Such a diverse set of models provides a broader view of what linguistic capabilities are affected by the change of any of these properties. We also include several distilled models in our analysis. We find that different models excel in different symbolic reasoning tasks, suggesting that slight differences related to optimization or masking strategy might be more important than the pre-training approach, dataset size, or architecture. Furthermore, in contrast to Radford et al. (2019), we find that for oLMpics tasks model size rarely correlates with the model performance. In addition, we observe that all mod-

els fail on composition tasks when evaluated in a zero-shot fashion.

2 Related Work

Pre-trained model analysis is a rapidly growing area in NLP today. There are a number of methods for analyzing internal representations of a model, including structured head and FCN pruning (Michel et al., 2019; Voita et al., 2019; Prasanna et al., 2020), residual connection and layer normalization analysis (Kovaleva et al., 2021; Kobayashi et al., 2021), and analyzing attention patterns (Clark et al., 2019; Kovaleva et al., 2019).

Compared to these methods, probing tasks (Conneau et al., 2018; Tenney et al., 2019) provide a more direct way to evaluate what a model can and cannot accomplish. While it is possible to probe embeddings or hidden representations directly (Tenney et al., 2019; Liu et al., 2019a), with the adoption of pre-trained language models has made it possible to evaluate such models by framing probing tasks close to the original model objective (Radford et al., 2019; Talmor et al., 2019; Ettinger, 2020; Goldberg, 2019).

However, when a research area moves this quickly, it can be hard to keep up with many new models. Most of the existing research (Garí Soler and Apidianaki, 2020; Zagoury et al., 2021; Kassner et al., 2020) papers compare only one or two model families. Even some of the most recent works only probe BERT or very similar models (Zagoury et al., 2021; Liu et al., 2021). Only a small number of analysis papers have probed multiple (three or more) model families (Zhou et al., 2021; Ilharco et al., 2021).

In contrast to existing work, we perform a large-scale probing of 28 models across 8 different model families. We apply the existing probing benchmarks, namely, oLMpics (Talmor et al., 2019) and psycholinguistic datasets (Ettinger, 2020), to models that differ in the pre-training objective, datasets, size, architecture, and directionality.

3 Background

3.1 Models

We use 8 different model families in this study. All of them are based on the transformer architecture and pre-trained on general-domain texts, but this is where the similarities end. Their differences are described in Table 1. In this section, we discuss and highlight their differences, from the major ones to the ones that might appear very minor.

BERT (Devlin et al., 2018) is pre-trained on Book Corpus and Wikipedia using a combination of Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). It uses GELU activations (Hendrycks and Gimpel, 2016) for fully-connected layers. For the first 90% of the training iterations, the maximum length is 128, but then it is increased to 512.

RoBERTa (Liu et al., 2019b) is the most similar to BERT in this study; however, it differs from it in many small but important details: the pre-training dataset is considerably larger and includes OpenWebText (Gokaslan and Cohen, 2019), Stories (Trinh and Le, 2018), and CC-News; does not use Next Sentence Prediction; uses dynamic masking; always trains with 512 max tokens; uses a smaller ADAM $\beta = 0.98$; 8 times larger batch size than BERT; and a larger, byte-level BPE vocabulary (50K instead of 31K).

DistilBERT (Sanh et al., 2019) is a distilled version of BERT. It has half the layers of BERT and is trained using soft targets produced by BERT.

ALBERT (Lan et al., 2019) shares parameters across transformer layers and uses an extra projection between the embedding and the first transformer layer. It replaces NSP with the sentence-order prediction. ALBERT uses n-gram masking and the LAMB (You et al., 2019) optimizer. The training setup is similar to BERT, but it trains 90% of the time using the sequence length 512 and randomly reduces it in 10% of iterations. Parameter sharing allows ALBERT to achieve performance similar to BERT with much fewer trainable parameters. The smallest ALBERT model has 12M trainable parameters and the largest has 235M.

ALBERTv2 is a minor modification of ALBERT that was trained without dropout, for twice as many training steps with additional training data [2].

GPT-2 (Radford et al., 2019) is a unidirectional transformer language model trained on the WebText dataset. Unlike other models, it is a Pre-Norm transformer. Similar to RoBERTa, T5 has a 50K vocabulary and a byte-level BPE but treats spaces as a separate symbol. It also comes in multiple sizes from 124M parameters up to 2.8B parameters. There exist several popular reimplementations of this model, such as GPT-Neo (Black et al., 2021),

[1] GPT_{NEO} is trained on a 800Gb dataset.

[2] github.com/google-research/albert

Model	Parameters	Pre-training Data Size	Enc-Dec	Autoregressive	Tokenization	Vocab. Size
BERT	110M - 340M	16 GB	No	No	WordPiece	30,522
RoBERTa	355M	160 GB	No	No	BPE	50,265
DistilBERT	66M	16 GB	No	No	WordPiece	30,522
AlBERT	12M - 235M	16 GB	No	No	SentencePiece	30,000
GPT2	124M - 1.5B	40GB[1]	No	Yes	BPE	50,257
UniLM	340M	16 GB	No	N/A	WordPiece	28,996
BART	406M	160 GB	Yes	Yes	BPE	50,265
T5	223M-2.8B	750 GB	Yes	Yes	SentencePiece	32,128

Table 1: Model families used in this study. Enc-Dec stands for encoder-decoder architecture. Autoregressive means that the model was trained with a causal mask. Note that UniLM is trained using a generalized language modeling objective that includes both unidirectional and bidirectional subtasks and cannot be attributed to either autoregressive or non-autoregressive.

which generally follow the original paper but differ in dataset (Gao et al., 2020), model, and training hyperparameters.

UniLM (Dong et al., 2019) utilizes several attention masks to control the access to context for each word token. It uses a multi-task objective that is modeled by applying different attention masks. The mix of tasks includes masked language modeling, unidirectional language modeling, and sequence-to-sequence language modeling. Additionally, it employs the NSP objective and is initialized using BERT model weights. In optimization, it generally follows BERT but always uses 512 as the maximum sequence length.

BART (Lewis et al., 2019) is an encoder-decoder model that is trained on text infilling and sentence permutation tasks. It is trained on the same dataset as RoBERTa. Compared to BERT, BART does not use an additional projection when predicting word logits. In optimization, it closely follows RoBERTa, but disables dropout for the final 10% of training.

T5 (Raffel et al., 2019) is also an encoder-decoder model. It is trained using a text infilling task on the C4 dataset. However, it only generates the text in place of the [MASK] token and not the full input sentence. Architecturally, it is a Pre-Norm model and Layer Norm inputs are only rescaled with no additive bias applied. Output projection weights are tied with the input embedding matrix. It uses 128 relative positional embeddings that are added at every layer. Unlike most of the models in this study, it uses the ReLU activation. The smallest T5 model used in this study has 233M parameters and the largest has 2.8B.[3]

Unlike the original T5, T5v1.1[4] does not tie logit layer with input embeddings, uses GEGLU activations (Shazeer, 2020) and no dropout. It also slightly changes model shapes.

3.2 oLMpics

The oLMpics benchmark consists of eight tasks that test a model's ability to draw comparisons, associate properties, and perform multi-hop composition of facts, among others. OLMpics tests multiple specific skills, such as the ability to compare ages, understand negation, and perform simple linguistic composition tasks. Some example questions are shown in Table 2.

Zero-Shot vs. Multi-Shot A major advantage of the oLMpics tasks is that zero-shot evaluation can be performed for most tasks due to the task format. Zero-shot evaluation eliminates the ambiguity of whether a model's knowledge is stored in its pre-trained representations or learned during fine-tuning. However, a model may possess the necessary information but fail during zero-shot evaluation due to the wording of the task. Therefore, multi-shot evaluation can also be informative, allowing the model to adapt to the input format and possibly learn task-specific features. OLMpics tasks include training sets specifically for this reason, in order to separate the impact of fine-tuning from pre-training.

MC-MLM vs. MC-QA The oLMpics tasks are framed in one of two ways: MC-MLM (Multiple Choice-Masked Language Modeling) and MC-QA (Multiple Choice-Question Answering). MC-MLM tasks are formulated as a masked language modeling task (Devlin et al., 2018), where the model needs to predict the word replaced by the MASK token. An example of an *Age Comparison* sentence is "A 41 year old is [MASK] a 42 year

[3] We have not performed evaluation on the 11B model.
[4] huggingface.co/google/t5-v1_1-base

Task Name	Example Question	Choices
Age Comparison	A 41 year old person age is [MASK] than a 42 year old person.	younger, older
Object Comparison	The size of a nail is usually [MASK] than the size of a fork.	smaller, larger
Antonym Negation	It was [MASK] a fracture, it was really a break.	not, really
Taxonomy Conjunction	A ferry and a biplane are both a type of [MASK].	airplane, craft, boat
Property Conjunction	What is related to vertical and is related to honest?	straight, trustworthy, steep
Encyclopedic Composition	When did the band where Alan Vega played first form?	1970, 1968, 1969
Hypernym Conjunction	A basset and a tamarin are both a type of [MASK]	primate, dog, mammal
Multi-hop Composition	When comparing a 21 year old, 15 year old, and 19 year old, the [MASK] is oldest.	third, first, second

Table 2: Examples of oLMpics questions, with the correct answer underlined.

old." A model's prediction is determined by the probabilities assigned to the [MASK] token, with "younger" being selected if its probability is higher than "older," and "older" otherwise.

MC-MLM restricts the possible answers to single tokens. Tasks with longer answers require MC-QA. In this method, a new feedforward neural network maps the [CLS] token embedding to a single logit. For prediction, answer choices are individually concatenated to the original question, forming a new sentence for each choice. This set of sentences is input into the model, and the choice corresponding to the sentence with the largest logit is selected. While the MC-QA method allows for longer choices, the added feedforward network must be trained; therefore, zero-shot evaluation is not possible for these tasks.

Extending Beyond MLM The oLMpics MC-MLM method relies on the model giving probabilities of individual words in a bidirectional context. However, models like GPT2 do not have access to the future context, which makes it impossible to resolve examples like "A 41 year old is [MASK] than 42 year old." For these models, we sum the log-probabilities of individual words to find the probability of the whole sentence. We do this for every possible answer, e.g., a sequence with "younger" instead of [MASK] and "older". Then, we select the one with the highest total probability.

Extending BART and T5 is more straightforward because their objectives and architecture are very flexible. For both of these models, we use the original oLMpics input format. T5 has multiple [MASK]-tokens and we always use <X> token in our evaluation. The biggest difference is that BART produces the full sentence and we need to extract the probabilities for the masked words and T5 produces only the tokens in the place of <X>.

3.3 Psycholinguistic Data

Similar to oLMpics, the datasets used by Ettinger (2020) are framed as "fill in the blank" tasks. Unlike oLMpics, the model always needs to predict only the last word, so both bidirectional and unidirectional models can be evaluated on these tasks directly.

The biggest distinction of this dataset is its source. The datasets CPRAG-102 (Federmeier and Kutas, 1999), ROLE-88 (Chow et al., 2016), and NEG-136 (Fischler et al., 1983) come from the psycholinguistics and neuroscience studies and were originally evaluated on humans.

CPRAG-102 targets commonsense and pragmatic inference e.g. *Justin put a second house on Park Place. He and his sister often spent hours playing* __, Target: *monopoly*, other labels: *chess, baseball*. ROLE-88 aims at evaluating event knowledge and semantic roles e.g. *The restaurant owner forgot which customer the waitress had* __ versus *The restaurant owner forgot which waitress the customer had* __, target: *served*.

NEG-136 tests how well models understand the meaning of negation and consists of two subsets: simple (SIMP) and natural (NAT). For example, SIMP: *Salmon is a fish/dog versus Salmon is not a fish/dog.* NAT: *Rockets and missiles are very fast/slow versus Rockets and missiles aren't very fast/slow*. Evaluation of this dataset is performed in two ways: affirmative statements and negative statements. For affirmative ones, the model needs to complete a sentence like *A robin is a* with the expected answer *bird*. For negative, *A robin is not a* should not be completed with a *bird*. (Ettinger, 2020) finds that this type of error is very common in BERT, which suggests that the model cannot handle negation correctly.

Ettinger (2020) tests BERT models in two ways: using a pre-defined set of answers, similar to oLMpics MC-MLM, or computing top-k accuracy

from the whole model vocabulary. We adopt the same approach in this study.

4 Experiments

We evaluate 8 models families on the oLMpics (28 models in total) and 6 families on psycholinguistic data (17 models). This extends the (Talmor et al., 2019) results with 6 new model families and (Ettinger, 2020) with 4.

4.1 Language models are not universal multitask learners

Zero-shot evaluation It's been shown that language models can implicitly learn downstream tasks (Radford et al., 2019; Brown et al., 2020). However, it is still not obvious what tasks are learnable in this manner without explicit supervision. In our study, similar to Talmor et al. (2019), we find that none of the models can solve Multi-Hop Composition or Always-Never tasks substantially better than a majority baseline (see Table 4).

This holds true not only for masked language models but also for unidirectional language models such as GPT2 and text-infilling models such as T5 or BART. Only small and base versions of T5v1.1 outperform the majority baseline on *Multi-Hop Composition* by a small margin.

Multi-shot evaluation Not surprisingly, fine-tuning models on oLMpics improves the scores across the board. This is true even for the tasks on which zero-shot performance is extremely poor. For example, while all models fail on *Multi-hop Composition* during zero-shot evaluation, most models can reach perfect or near-perfect accuracy on this task after fine-tuning. However, *Always-Never* and *Taxonomy Conjunction* remain challenging for all models. For the full multi-shot evaluation, see Table 7 in the Appendix.

4.2 Bigger does not mean better

To check how the size of a model affects the performance, we evaluated different versions of GPT2, T5, and ALBERT models on the oLMpics tasks ranging from 14M (smallest ALBERT) to 2.8B (largest T5) parameters. All of the models perform near-random on 3 out of the 6 tasks, suggesting that Multi-Hop Composition, Antonym Negation, and Always-Never are hard to learn via the language modeling objective. On the rest of the tasks, we observe no clear improvement trend for GPT models based on the model size. In most of the tasks, $\text{GPT}_{\text{large}}$ either performs on par or has higher accuracy than GPT_{xl} while being twice as small.

We also compute Spearman correlation between model accuracy and model size for GPT2, ALBERT, and T5 models.[5] For all GPT2 and ALBERT (v1 and v2) tests, p-value is $\gg 0.05$, suggesting that there is no rank-correlation between model size and task performance. However, in the case of T5 models, there is a strong (1.0) and significant correlation (p-value 10^{-6}) for all tasks except *Always-Never*. We account for multiple hypothesis testing using Bonferroni's method. For the *Taxonomy Conjunction*, the correlation is negative.

4.3 Model properties are not predictive of model performance

On the oLMpics tasks, with rare exceptions (Section 4.1), we do not observe that parameter count, dataset size, model architecture or directionality are predictive of model performance on zero-shot oLMpics (Table 4).

$\text{RoBERTa}_{\text{large}}$ usually performs amongst the best models, while having a very similar architecture and objective to $\text{BERT}_{\text{large}}$. Reasonable explanations would be the dataset size, but this does not align with the $\text{BART}_{\text{large}}$ results. Encoder-decoder architecture does seem not to be indicative of the performance either, as T5_{large} and $\text{BART}_{\text{large}}$ have vastly different results.

Psycholinguistic datasets (Table 5) demonstrate similar behaviour. $\text{RoBERTa}_{\text{large}}$ is generally the stongest model followed by T5_{xl}. We would like to note that these datasets have less than 100 examples and their statistical power (Card et al., 2020) is very small.

Our intuitions about the relative suitability of different model classes are based on their performance on standard benchmarks (Wang et al., 2018, 2019) and existing investigations of scaling laws (Radford et al., 2019; Kaplan et al., 2020). In contrast to this received wisdom, our experiments suggest that this does not in fact lead to better performance on specific linguistic skills.

4.4 RoBERTa is sensitive to negation

Ettinger (2020) observed that BERT is not sensitive to negation in non-natural (SIMP) or less-natural cases.

In our experiments (Table 6), we find that the only model with zero accuracy outside of BERT is

[5] Note that sample size for each test is ≤ 4, so these results should be taken as anecdotal.

Input sequence example	$GPT2_B$	$GPT2_M$	$GPT2_L$	$GPT2_{XL}$
(oLMpics) It was really/not sane, it was really insane	53.3	52.8	**59.0**	60.6
It was really insane. Was it sane ? yes/no	51.6	**58.2**	55.6	**61.4**
It was really insane. Was it really sane ? yes/no	50.2	54	50.2	54.4
It was sane entails it was really insane ? yes/no	49.8	50.2	50	50.6
Sentence 1: It was sane. Sentence 2: It was really insane. Is Sentence 1 synonym of Sentence 2? yes/no	**59.6**	50.2	46.8	48.4

Table 3: Prompts for the *Antonym Negation* task. Random baseline accuracy is 50%. The original oLMpics prompt is the prompt used in Table 4. $GPT2_B$ is the base-sized model, $GPT2_M$ is medium, and $GPT2_L$ is large. Text highlighted in red/green are correct/wrong labels.

	Age Comp.	Always Never	Object Comp.	Antonym Negation	Taxonomy Conj.	Multi-hop Comp.
Majority	50.6	36.1	50.6	50.2	34.0	34.0
$BERT_{base}$	49.4	13.3	55.4	53.8	46.7	33.2
$BERT_{large}$	50.6	22.5	52.4	51.0	**53.9**	33.8
$BERT_{large}$ WWM	76.6	10.7	55.6	57.2	46.2	33.8
$RoBERTa_{large}$	98.6	13.5	87.4	**74.4**	45.4	28.0
$DistilBERT_{base}$	49.4	15.0	50.8	50.8	46.9	33.4
$AlBERT_{base}$	47.0	23.2	50.6	52.6	-	34.0
$AlBERT_{large}$	52.8	30.7	49.2	50.2	-	34.0
$AlBERT_{xlarge}$	39.8	26.1	50.4	44.6	-	32.2
$AlBERT_{xxlarge}$	95.4	22.9	61.0	66.4	-	34.0
$AlBERTv2_{base}$	50.6	21.4	49.4	54.2	-	14.0
$AlBERTv2_{large}$	51.4	31.7	50.6	55.2	-	34.0
$AlBERTv2_{xlarge}$	46.2	**37.9**	50.6	62.4	-	32.4
$AlBERTv2_{xxlarge}$	93.8	23.9	78.8	64.8	-	34.0
$BART_{large}$	49.4	14.3	50.8	53.8	42.6	33.8
$T5_{small}$	49.4	16.1	48.2	47.0	49.3	33.8
$T5_{base}$	49.4	10.7	59.0	53.4	46.6	33.6
$T5_{large}$	94.0	25.7	79.8	59.2	42.2	33.8
$T5_{xl}$	**100.0**	20.4	**90.0**	68.4	41.2	34.4
$T5v1.1_{small}$	49.4	34.3	50.6	51.4	48.2	**37.8**
$T5v1.1_{base}$	50.6	11.8	56.0	45.0	49.9	37.6
$T5v1.1_{large}$	49.6	15.7	50.6	47.0	41.7	33.8
$T5v1.1_{xl}$	49.4	23.9	49.4	54.2	**53.9**	33.8
$UniLM_{base}$	47.9	15.5	47.8	43.5	45.1	34.9
$UniLM_{large}$	47.9	19.2	61.1	50.8	50.2	33.1
$GPT2_{base-0.1B}$	47.6	9.0	50.3	53.3	49.1	32.6
$GPT2_{medium-0.3B}$	50.1	31.3	50.3	52.8	51.9	34.0
$GPT2_{large-0.8B}$	69.6	26.0	50.5	59.0	46.9	34.0
$GPT_{NEO-1.3B}$	58.6	29.0	52.1	65.2	50.6	33.3
$GPT2_{xl-1.5B}$	51.9	26.6	52.6	60.6	45.8	34.0

Table 4: Zero-shot oLMpics evaluation on MC-MLM tasks. "Majority" here is the accuracy when predicting the most frequent class. The first 4 models are our reproduction of the original oLMpics results. The best result on each task is highlighted in bold. We do not evaluate ALBERT on Taxonomy Conjunction because its vocabulary does not contain requires classes as single tokens. A version of this table with confidence intervals can be found in Table 10 in the Appendix.

.

a distilled version of BERT itself. However, multiple models achieve non-zero accuracy on NEG-SIMP (neg), while still being insensitive to the negation. For example, while $ALBERTv1_{xlarge}$ is the best-performing model on NEG-SIMP (neg) with 44.4% accuracy, this accuracy is mainly caused by mistakes in language modeling while still being insensitive to negation (e.g., it predicts *vegetable* for both *An ant is a* and *An ant is not a*). Specifically, $ALBERTv1_{xlarge}$ only changes its predictions in 5.5% cases.

However, unlike other models, $RoBERTa_{large}$ actually change its predicitons in 33% cases, suggesting that that sensitivity to negation is possible to learn via masked language modeling.

4.5 Models make plausible mistakes

One drawback of datasets from (Ettinger, 2020) that we have noticed was the ambiguity of answers. For example, many models predict words like "this", "that", "it" as the next word for *"Checkmate," Rosaline announced with glee. She was getting to be really good at [MASK]* instead of the word "chess". All of these continuations are both grammatically and semantically correct, even though they do not answer the question. Another example would be *I'm an animal like Eeyore!" the child exclaimed. His mother wondered why he was pretending to be a [MASK]*. CPRAG expects the answer "donkey", which assumes that the reader (or model) is familiar with the English names of Winnie-the-Pooh book characters[6].

4.6 Antonym negation: Impact of prompt variation

A number of recent studies have shown that language model performance is sensitive to the task prompt (Schick and Schütze, 2020). We have experimented with four different prompts for the *Antonym Negation* task. Table 3 shows the patterns and the corresponding accuracies of GPT models. All experiments use "yes"/"no" verbalizers.

While some prompts improve the oLMpics prompt results substantially (up to +6%), this improvement is not consistent across models showing that even very similar models are sensitive to prompt variation in different ways.

Additionally, the prompt #4 (Table 3) improves the smallest $\text{GPT2}_{\text{base}}$ model so much that it performs on par with the largest model, demonstrating that parameter count is not a reliable predictor of the model performance.

4.7 Age comparison: Accuracy varies by age group

Similar to Talmor et al. (2019), we find that models do not perform equally well on all age ranges. Figure 1 shows that with the exception of $\text{GPT2}_{\text{base}}$, all GPT2 variants perform well on 10-20 years age group and badly on the 30-40 age group, with a significant drop in performance from 80% to 20%. Generally, GPT2 seems to predict younger ages more accurately. However, the smallest model, $\text{GPT2}_{\text{base}}$, exhibits a different trend than other models as age increases.

[6]Only one of the authors of this paper was able to continue this correctly

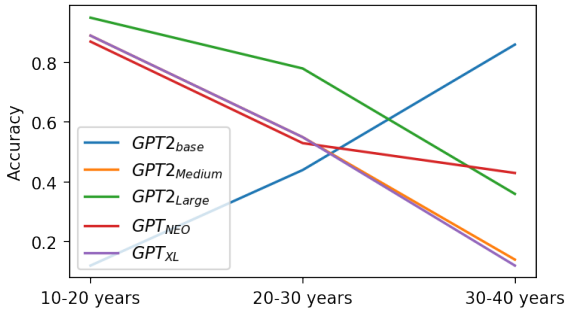

Figure 1: Evaluation of GPT2 variants on Age Comparison task for different age groups.

5 Conclusion

In this work, we apply a large and diverse set of models to oLMpics and psycholinguistic tasks. The variety of models allows us to investigate the performance of different architectures and pre-training methods on a variety of linguistic tasks. Contrary to received wisdom, we find that parameter count within a given model family does not correlate with model performance on these tasks. We find that none of the models, even the 2.8B-sized ones, can resolve *Multi-Hop Composition* and *Always-Never* tasks in zero-shot manner, suggesting that the existing pre-training methods cannot learn such tasks. Finally, we find that different models excel in different symbolic reasoning tasks, suggesting that slight differences related to optimization or masking strategy might be more important than the pre-training approach, dataset size, or architecture.

References

Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. 2021. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow. If you use this software, please cite it using these metadata.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

	CPRAG-126	ROLE-88	NEG-136 SIMP(Aff)	NEG-136 NAT(Aff)
$\text{BERT}_{\text{base}}$	52.9	27.3	**100**	43.8
$\text{BERT}_{\text{large}}$	52.9	37.5	**100**	31.3
$\text{RoBERTa}_{\text{base}}$	50.0	35.0	94.0	56.3
$\text{RoBERTa}_{\text{large}}$	**64.7**	**46.6**	88.0	50.0
$\text{DistilBERT}_{\text{base}}$	35.3	20.1	94.4	37.5
$\text{AlBERTv1}_{\text{base}}$	17.6	0.0	72.2	12.5
$\text{AlBERTv1}_{\text{large}}$	20.6	0.0	72.2	18.8
$\text{AlBERTv1}_{\text{xlarge}}$	32.4	2.3	11.1	6.5
$\text{AlBERTv1}_{\text{xxlarge}}$	47.1	11.3	88.9	43.8
$\text{AlBERTv2}_{\text{base}}$	0.0	0.0	22.2	6.3
$\text{AlBERTv2}_{\text{large}}$	29.4	0.0	50.0	6.3
$\text{AlBERTv2}_{\text{xlarge}}$	61.8	12.5	94.4	37.5
$\text{AlBERTv2}_{\text{xxlarge}}$	61.8	12.5	94.4	37.5
T5_{small}	20.6	9.1	44.4	18.8
T5_{base}	38.2	22.7	88.9	31.3
T5_{large}	50.0	36.4	94.4	43.8
T5_{xl}	55.9	44.3	83.3	**62.5**

Table 5: Zero-shot top-5 word prediction accuracy. Top-5 is selected over the whole model vocabulary. The first 2 models are our reproduction of the original (Ettinger, 2020) results. The best result on each task is highlighted in bold. SIMP stands for simple, NAT stands for natural. Both negation tasks are evaluated in the affirmative form.

	CPRAG	ROLE	NEG SIMP (Aff)	NEG SIMP (Neg)	NEG NAT (Aff)	NEG NAT (Neg)	NEG LNAT (Aff)	NEG LNAT (Neg)
$\text{BERT}_{\text{base}}$	73.5	75.0	**100.0**	0.0	62.5	87.5	75.0	0.0
$\text{BERT}_{\text{large}}$	**79.4**	**86.4**	**100.0**	0.0	**75.0**	**100**	75.0	0.0
$\text{RoBERTa}_{\text{base}}$	26.5	59.0	66.7	33.3	37.5	75.0	75.0	12.5
$\text{RoBERTa}_{\text{large}}$	29.4	68.2	**100.0**	33.3	25.0	75.0	75.0	12.5
$\text{DistilBERT}_{\text{base}}$	67.6	63.6	**100.0**	0.0	62.5	62.5	**87.5**	0.0
$\text{AlBERTv1}_{\text{base}}$	5.8	43.2	72.2	22.2	50.0	25.0	62.5	37.5
$\text{AlBERTv1}_{\text{large}}$	8.8	54.5	72.2	27.8	25	12.5	75.0	37.5
$\text{AlBERTv1}_{\text{xlarge}}$	14.7	52.3	58.3	**44.4**	37.5	25.0	75.0	12.5
$\text{AlBERTv1}_{\text{xxlarge}}$	29.4	45.5	94.4	16.7	24.0	62.5	62.5	25.0
$\text{AlBERTv2}_{\text{base}}$	14.7	36.4	61.1	33.3	50	37.5	50.0	12.5
$\text{AlBERTv2}_{\text{large}}$	18.0	57.0	61.1	38.9	62.5	25.0	50.0	50.0
$\text{AlBERTv2}_{\text{xlarge}}$	26.5	37.5	88.9	11.1	25.0	62.5	37.5	25.0
$\text{AlBERTv2}_{\text{xxlarge}}$	26.5	54.5	88.9	11.1	25.0	62.5	37.5	25.0
T5_{small}	5.9	45.5	55.6	33.3	50.0	25.0	37.5	**62.5**
T5_{base}	14.7	70.5	61.1	27.8	50.0	12.5	37.5	37.5
T5_{large}	17.6	54.5	72.2	16.7	62.5	37.5	37.5	50.0
T5_{xl}	14.7	63.6	66.7	27.8	62.5	50.0	37.5	50.0
$\text{GPT2}_{\text{base}}$	14.7	34.1	66.7	38.9	75.0	25.0	37.5	37.5
$\text{GPT2}_{\text{medium}}$	17.6	36.4	61.1	22.2	50.0	50.0	50.0	**62.5**
$\text{GPT2}_{\text{large}}$	29.4	45.5	77.8	16.7	62.5	50.0	37.5	50.0
GPT_{neo}	20.6	45.5	77.8	33.3	75.0	37.5	62.5	25.0
GPT2_{xl}	17.6	50	61.1	33.3	62.5	75.0	62.5	37.5

Table 6: Zero-shot sensitivity accuracy. The first 2 models are our reproduction of the original (Ettinger, 2020) results. Completion sensitivity is the percentage of instances for which the model assigns a higher probability to the good completion than to the bad completions. The best result on each task is highlighted in bold. SIMP stands for simple, NAT for natural, LNAT for less natural as defined in (Ettinger, 2020).

Dallas Card, Peter Henderson, Urvashi Khandelwal, Robin Jia, Kyle Mahowald, and Dan Jurafsky. 2020. With little power comes great responsibility. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9263–9274, Online. Association for Computational Linguistics.

Wing-Yee Chow, Cybelle Smith, Ellen Lau, and Colin Phillips. 2016. A "bag-of-arguments" mechanism for initial verb predictions. *Language, Cognition and Neuroscience*, 31(5):577–596.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286.

Peter E. Clark, Oyvind Tafjord, and Kyle Richardson. 2020. Transformers as soft reasoners over language. *ArXiv*, abs/2002.05867.

Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *ACL*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. *arXiv preprint arXiv:1905.03197*.

Allyson Ettinger. 2020. What bert is not: Lessons from a new suite of psycholinguistic diagnostics for language models.

Kara D Federmeier and Marta Kutas. 1999. A rose by any other name: Long-term memory structure and sentence processing. *Journal of memory and Language*, 41(4):469–495.

Ira Fischler, Paul A Bloom, Donald G Childers, Salim E Roucos, and Nathan W Perry Jr. 1983. Brain potentials related to stages of sentence verification. *Psychophysiology*, 20(4):400–409.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Aina Garí Soler and Marianna Apidianaki. 2020. Bert knows punta cana is not just beautiful, it's gorgeous: Ranking scalar adjectives with contextualised representations. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Aaron Gokaslan and Vanya Cohen. 2019. Openwebtext corpus. `http://Skylion007.github.io/OpenWebTextCorpus`.

Yoav Goldberg. 2019. Assessing bert's syntactic abilities. *ArXiv*, abs/1901.05287.

Dan Hendrycks and Kevin Gimpel. 2016. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. *ArXiv*, abs/1606.08415.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *ACL*.

Gabriel Ilharco, Rowan Zellers, Ali Farhadi, and Hannaneh Hajishirzi. 2021. Probing contextual language models for common ground with visual representations. In *NAACL*.

Zhengbao Jiang, J. Araki, Haibo Ding, and Graham Neubig. 2021. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models.

Nora Kassner, Benno Krojer, and Hinrich Schütze. 2020. Are pretrained language models symbolic reasoners over knowledge? In *CONLL*.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. 2020. Attention is not only a weight: Analyzing transformers with vector norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7057–7075.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentarou Inui. 2021. Incorporating residual and normalization layers into analysis of masked language models. In *EMNLP*.

Olga Kovaleva, Saurabh Kulshreshtha, Anna Rogers, and Anna Rumshisky. 2021. BERT busters: Outlier dimensions that disrupt transformers. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3392–3405, Online. Association for Computational Linguistics.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the dark secrets of bert. *arXiv preprint arXiv:1908.08593*.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Leo Z. Liu, Yizhong Wang, Jungo Kasai, Hannaneh Hajishirzi, and Noah A. Smith. 2021. Probing across time: What does roberta know and when? *ArXiv*, abs/2104.07885.

Nelson F Liu, Matt Gardner, Yonatan Belinkov, Matthew E Peters, and Noah A Smith. 2019a. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094.

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

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Hosein Mohebbi, Ali Modarressi, and Mohammad Taher Pilehvar. 2021. Exploring the role of bert token representations to explain sentence probing results. In *EMNLP*.

Marius Mosbach, Anna Khokhlova, Michael A. Hedderich, and Dietrich Klakow. 2020. On the interplay between fine-tuning and sentence-level probing for linguistic knowledge in pre-trained transformers. In *FINDINGS*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL*.

Jason Phang, Haokun Liu, and Samuel R. Bowman. 2021. Fine-tuned transformers show clusters of similar representations across layers. *ArXiv*, abs/2109.08406.

Sai Prasanna, Anna Rogers, and Anna Rumshisky. 2020. When bert plays the lottery, all tickets are winning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3208–3229.

Alec Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Abhilasha Ravichander, Eduard Hovy, Kaheer Suleman, Adam Trischler, and Jackie Chi Kit Cheung. 2020. On the systematicity of probing contextualized word representations: The case of hypernymy in BERT. In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, pages 88–102, Barcelona, Spain (Online). Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Timo Schick and Hinrich Schütze. 2020. Few-shot text generation with pattern-exploiting training.

Noam Shazeer. 2020. Glu variants improve transformer.

Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2019. olmpics–on what language model pre-training captures. *arXiv preprint arXiv:1912.13283*.

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

Trieu H. Trinh and Quoc V. Le. 2018. A simple method for commonsense reasoning.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2019. Superglue: A stickier benchmark for general-purpose

language understanding systems. *arXiv preprint arXiv:1905.00537*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. 2019. Large batch optimization for deep learning: Training bert in 76 minutes. *arXiv preprint arXiv:1904.00962*.

Avishai Zagoury, Einat Minkov, Idan Szpektor, and William W. Cohen. 2021. What's the best place for an ai conference, vancouver or _____: Why completing comparative questions is difficult. In *AAAI*.

Pei Zhou, Rahul Khanna, Bill Yuchen Lin, Daniel Ho, Jay Pujara, and Xiang Ren. 2021. Rica: Evaluating robust inference capabilities based on commonsense axioms. In *EMNLP*.

A Additional Tables

The next pages present additional results, including the version of Table 4 with confidence intervals (Table 10), oLMpics MC-QA results (Table 7), T5 zero-shot Encyclopedic Composition and Property Conjunction (Table 8), and T5 evaluated on psycholinguistic datasets when removing stop-words from the model output vocabulary (Table 9).

	Age Comp.	Always Never	Object Comp.	Ant. Neg.	Tax. Conj.	Multi-hop Compos.	Encyc. Conj.	Prop. Conj.
Majority	50.6	20.0	50.0	50.0	34.0	34.0	34.0	34.0
$\text{BERT}_{\text{base}}$	86.8	59.3	86.6	92.0	57.4	86.0	56.1	**62.6**
$\text{BERT}_{\text{large}}$	98.8	58.9	90.4	94.8	**60.8**	99.0	57.1	58.3
$\text{BERT}_{\text{large}}$ WWM	**100.0**	58.9	85.0	95.0	58.8	97.6	56.4	60.1
$\text{RoBERTa}_{\text{large}}$	100	**60.4**	87.2	**96.2**	59.9	**100.0**	55.5	55.5
$\text{DistilBERT}_{\text{base}}$	66.2	60.0	84.2	90.6	55.9	59.4	53.9	56.2
$\text{AlBERT}_{\text{large}}$	91.6	59.3	66.4	90.4	-	80.0	**57.2**	60.2
$\text{BART}_{\text{large}}$	**100.0**	36.1	85.6	95.0	59.8	**100.0**	-	-
T5_{base}	77.6	55.7	91.4	94.4	-	64.8	-	-
T5_{large}	**100.0**	57.9	**93.2**	96.0	-	**100.0**	-	-

Table 7: Multi-shot oLMpics evaluation on MC-MLM and MC-QA tasks. "Majority" here is the accuracy when predicting the most frequent class.

	Encyc. Conj.	Prop. Conj.
T5_{small}	29.0	38.72
T5_{base}	31.4	36.2
T5_{large}	31.6	34.6
T5_{xl}	31.2	38.5
$\text{T5v1.1}_{\text{small}}$	33.4	38.1
$\text{T5v1.1}_{\text{base}}$	31.6	40.0
$\text{T5v1.1}_{\text{large}}$	31.4	**40.1**
$\text{T5v1.1}_{\text{xl}}$	**33.4**	37.1

Table 8: Zero-shot T5 results on MC-QA tasks. As for T5 can generate multiple tokens in place of a single mask, we evaluate in using similar to MC-MLM. In order to get the probability of the answer, we multiply the probabilities for every answer word.

	CPRAG-126	ROLE-88	NEG-136 SIMP(Aff)	NEG-136 NAT(Aff)
T5_{small}	20.6	9.1	44.4	18.8
T5_{base}	38.2	22.7	88.9	31.3
T5_{large}	50.0	36.4	**94.4**	43.8
T5_{xl}	55.9	44.3	83.3	**62.5**
T5_{small} Filtered	20.6	15.9	55.6	25.0
T5_{base} Filtered	42.2	34.1	88.9	37.5
T5_{large} Filtered	52.9	38.6	**94.4**	43.8
T5_{xl} Filtered	**58.8**	**51.1**	88.9	**62.5**

Table 9: Zero-shot top-5 word prediction accuracy. Top-5 is selected over the whole model vocabulary for the first 4 rows (same as Table 5). In the last 4 rows, we remove the 179 most common English stop words, as well as the " " token from the vocabulary.

	Age Comp.	Always Never	Object Comp.	Antonym Negation	Taxonomy Conj.	Multi-hop Comp.
Majority	50.6	36.1	50.6	50.2	34.0	34.0
$\text{BERT}_{\text{base}}$	49.4 ± 0.2	13.2 ± 1.2	55.4 ± 1.0	53.8 ± 1.0	46.8 ± 0.6	33.4 ± 0.6
$\text{BERT}_{\text{large}}$	50.6 ± 0.2	22.5 ± 1.3	52.4 ± 1.6	50.8 ± 0.8	$\mathbf{53.9 \pm 0.9}$	33.8 ± 0.7
$\text{BERT}_{\text{large}}$ WWM	76.4 ± 1.7	10.7 ± 1.5	55.8 ± 1.1	57.2 ± 0.7	46.4 ± 0.8	33.8 ± 0.7
$\text{RoBERTa}_{\text{large}}$	98.6 ± 0.1	13.5 ± 1.6	87.4 ± 0.9	$\mathbf{74.6 \pm 0.8}$	45.4 ± 0.4	28.0 ± 1.0
$\text{DistilBERT}_{\text{base}}$	49.4 ± 0.2	15.0 ± 1.2	51.0 ± 1.3	50.8 ± 0.7	46.8 ± 0.8	34.0 ± 1.0
$\text{DistilRoBERTa}_{\text{base}}$	45.4 ± 1.2	13.9 ± 1.3	50.8 ± 0.7	51.0 ± 1.0	50.6 ± 1.1	34.0 ± 1.0
$\text{AlBERT}_{\text{base}}$	47.0 ± 0.6	23.2 ± 1.2	50.6 ± 0.7	52.6 ± 1.0	-	34.0 ± 1.0
$\text{AlBERT}_{\text{large}}$	52.8 ± 1.2	30.7 ± 1.0	49.2 ± 0.7	50.2 ± 1.0	-	34.0 ± 1.0
$\text{AlBERT}_{\text{xlarge}}$	39.8 ± 0.3	26.1 ± 1.5	50.4 ± 0.8	44.6 ± 1.4	-	32.2 ± 1.2
$\text{AlBERT}_{\text{xxlarge}}$	95.4 ± 0.4	22.9 ± 0.5	61.0 ± 0.7	66.4 ± 0.5	-	34.0 ± 1.0
$\text{AlBERTv2}_{\text{base}}$	50.6 ± 0.2	21.4 ± 0.9	49.4 ± 0.7	54.2 ± 1.7	-	34.0 ± 1.0
$\text{AlBERTv2}_{\text{large}}$	51.4 ± 0.6	31.7 ± 1.5	50.6 ± 0.6	55.2 ± 1.3	-	34.0 ± 1.0
$\text{AlBERTv2}_{\text{xlarge}}$	46.2 ± 0.7	37.9 ± 1.9	50.6 ± 0.7	62.4 ± 0.9	-	32.4 ± 0.8
$\text{AlBERTv2}_{\text{xxlarge}}$	93.8 ± 0.5	23.9 ± 0.7	78.8 ± 0.8	64.8 ± 0.5	-	34.0 ± 1.0
$\text{BART}_{\text{large}}$	49.4 ± 0.2	23.2 ± 1.2	49.4 ± 0.7	49.8 ± 1.0	48.8 ± 0.9	33.8 ± 0.7
T5_{small}	49.4 ± 0.2	16.1 ± 1.6	48.2 ± 0.8	47.0 ± 0.9	49.3 ± 0.4	33.8 ± 0.7
T5_{base}	49.4 ± 0.2	10.7 ± 1.2	59.0 ± 0.7	53.4 ± 0.8	46.6 ± 0.9	33.6 ± 0.7
T5_{large}	94.0 ± 0.4	25.7 ± 0.7	83.2 ± 0.5	64.6 ± 1.4	42.2 ± 1.0	33.8 ± 0.7
T5_{xl}	$\mathbf{100.0 \pm 0.0}$	20.4 ± 1.0	$\mathbf{90.0 \pm 0.5}$	68.4 ± 0.8	41.2 ± 0.8	34.4 ± 0.6
$\text{T5v1.1}_{\text{small}}$	49.4 ± 0.2	34.3 ± 1.8	50.6 ± 0.7	51.4 ± 1.1	48.2 ± 0.7	$\mathbf{37.8 \pm 0.9}$
$\text{T5v1.1}_{\text{base}}$	50.6 ± 0.2	11.8 ± 1.6	56.0 ± 1.5	45.0 ± 0.8	49.9 ± 0.7	37.6 ± 0.9
$\text{T5v1.1}_{\text{large}}$	49.6 ± 0.3	15.7 ± 0.8	50.6 ± 0.8	47.1 ± 1.1	41.7 ± 1.0	33.8 ± 0.7
$\text{T5v1.1}_{\text{xl}}$	49.4 ± 0.2	23.9 ± 1.8	49.4 ± 0.7	54.2 ± 1.2	$\mathbf{53.9 \pm 0.5}$	33.8 ± 0.7
$\text{UniLM}_{\text{base}}$	47.9±1.6	16.1±0.8	48.0±2.7	43.6±1.3	45.1±1.2	34.8±0.9
$\text{UniLM}_{\text{large}}$	47.9±1.6	19.9±1.3	61.4±1.8	51.2±1.4	50.2±2.1	33.6±0.7
$\text{GPT2}_{\text{base}-0.1\text{B}}$	47.6±1.2	**50.1±1.5**	50.1±1	52.8±1.9	48.4±1.0	32.2±2.4
$\text{GPT2}_{\text{medium}-0.3\text{B}}$	50.1±1.3	40.8±2.2	49.6±0.9	54.7±2.4	49.1±1.7	29.6±2.1
$\text{GPT2}_{\text{large}-0.8\text{B}}$	69.6±1.0	20.2±1.7	50.4±1.0	50.1±2.7	46.9±1.5	33.5±1.3
$\text{GPT}_{\text{NEO}-1.3\text{B}}$	58.6±0.7	29.0±1.0	52.1±0.7	65.2±1.1	50.6±1.5	33.3±1.0
$\text{GPT2}_{\text{xl}-1.5\text{B}}$	51.9±1.5	26.6±0.7	52.6±0.7	60.6±1.2	45.8±1.3	34.0±1.0

Table 10: Zero-shot oLMpics evaluation on MC-MLM tasks. "Majority" here is the accuracy when predicting the most frequent class. The first 4 models are our reproduction of the original oLMpics results. The best result on each task is highlighted in bold. Confidence intervals estimated via bootstrapping 20% of the data show errors about 1-2 absolute points.