On “Scientific Debt” in NLP: A Case for More Rigour in Language Model Pre-Training Research

Made Nindyatama Nityasya[1], **Haryo Akbarianto Wibowo**[1], **Alham Fikri Aji**[2],
Genta Indra Winata[3], **Radityo Eko Prasojo**[4], **Phil Blunsom**[5,6], **Adhiguna Kuncoro**[7]
[1]Independent Researcher [2]MBZUAI [3]Bloomberg [4]Universitas Indonesia
[5]Cohere.AI [6]University of Oxford [7]DeepMind
{made.nindyatama,haryo.akbarianto}@gmail.com, alham.fikri@mbzuai.ac.ae,
gwinata@bloomberg.net, radityo.ep@ui.ac.id, phil.blunsom@cs.ox.ac.uk,
akuncoro@deepmind.com

Abstract

This evidence-based position paper critiques current research practices within the language model pre-training literature. Despite rapid recent progress afforded by increasingly better pre-trained language models (PLMs), current PLM research practices often conflate different possible sources of model improvement, without conducting proper ablation studies and principled comparisons between different models under comparable conditions. These practices (i) leave us ill-equipped to understand which pre-training approaches should be used under what circumstances; (ii) impede reproducibility and credit assignment; and (iii) render it difficult to understand: “*How exactly does each factor contribute to the progress that we have today?*” We provide a case in point by revisiting the success of BERT over its baselines, ELMo and GPT-1, and demonstrate how — under comparable conditions where the baselines are tuned to a similar extent — these baselines (and even-simpler variants thereof) can, in fact, achieve competitive or better performance than BERT. These findings demonstrate how disentangling different factors of model improvements can lead to valuable new insights. We conclude with recommendations for how to encourage and incentivize this line of work, and accelerate progress towards a better and more systematic understanding of what factors drive the progress of our foundation models today.

1 Introduction

In recent years, language models that are pre-trained on large amounts of data have become the foundation models (Bommasani et al., 2021) for achieving state-of-the-art results on many NLP tasks (Peters et al., 2018; Devlin et al., 2019; Raffel et al., 2020; Liu et al., 2019, *inter alia*), and performed well at novel tasks through demonstrations (Brown et al., 2020). Hence, given the vast potential of pre-trained language models (PLMs), substantial effort has since been dedicated into developing the next state-of-the-art PLM, whether through new objective functions (Devlin et al., 2019; Yang et al., 2019; Clark et al., 2020; Raffel et al., 2020; Tay et al., 2022, *inter alia*), increasing model size (Kaplan et al., 2020; Brown et al., 2020; Chowdhery et al., 2022), better data filtering (Lee et al., 2022), or making a better use of the compute budget by training longer (Hoffmann et al., 2022).

To a large extent, this rapid progress is made possible by a strong emphasis on designing better foundation models and achieving new state-of-the-art results. In pursuit of this goal, each new PLM often leverages all possible means for improving performance: Indeed, it is often the case that a new state-of-the-art PLM paper not only proposed a new pre-training loss function as its key novelty, but was also trained on more data, used a larger model size, and benefited from the latest proven hyper-parameter settings and tricks of the trade compared to earlier PLMs. We argue, however, that progress through these practices also comes at a cost: As each new PLM differs from earlier one in *multiple* dimensions at once, it has become increasingly harder to (i) disentangle how these different components contribute to the model performance and progress that we observe today; (ii) understand which approaches work well under what circumstances; (iii) distill generalizable patterns and understand how well each result would transfer to new tasks, datasets, and settings (*e.g.,* in low-resource data and compute scenarios); and (iv) replicate prior results and conduct the appropriate credit assignment for the techniques and prior work that are most responsible for our progress today.

Much like how *technical debt* often arises when developing new software at breakneck speed,[1] we propose the term “**scientific debt**” to refer to the

[1] Technical debt refers to the results when technical teams develop new software at great speed, which later need to be reworked due to choosing limited solutions at the expense of more generalizable and principled approaches that take longer.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 8554–8572
July 9-14, 2023 ©2023 Association for Computational Linguistics

issues above that arise due to these PLM research practices. In this evidence-based position paper, we argue that scientific progress in NLP should strike a delicate balance between achieving the best performance on various benchmarks and leaderboards — an area where great progress has been made in recent years — and also on understanding: *How exactly does each different component affect the PLM performance that we observe today?* While this question is difficult to answer in light of the current PLM research practices that conflate different sources of model performance, we encourage the community to dedicate more effort into disentangling the performance gains from these interacting factors. Doing so would pave the way for achieving more progress in the future, in a way that is more scientifically rigorous, generalizable, reproducible, and *well-grounded* in a better understanding of how well each approach works under different settings.

We begin by motivating the importance of disentangling multiple possible factors of model improvement through an analogy with medicine, and discuss their parallels for PLM research (§2). We then provide empirical evidence by revisiting the success of BERT (Devlin et al., 2019), and demonstrate that prior PLMs like ELMo (Peters et al., 2018) and GPT-1 (Radford et al., 2018) can, in fact, achieve nearly the same performance ($\sim 1\%$ difference in aggregate GLUE) as BERT under *comparable* experimental conditions (§3). These findings serve to (i) further our understanding of the effectiveness of the masked language modelling loss compared to prior approaches, under comparable experimental conditions; and (ii) provide an example for how such work can yield valuable new insights regarding which approaches should be used under what conditions. We then conclude with several key recommendations, lessons learnt, and calls for change that would encourage and facilitate this line of work in the future, and accelerate our progress towards resolving the scientific debt that arises due to current PLM research practices (§4).

2 Analogy with Medicine and Parallels with PLM Research

Consider the following analogy with clinical trials in medicine. A drug trial showing that drug A (taken 10 times a day) works better than drug B (taken only twice a day) would raise a few critical questions: Would drug A still work just as well if it is taken at a lower dose? Can we get the same results by increasing the dose of drug B? How well would each drug work under comparable conditions, and are there any particular trade-offs (*e.g.,* at a lower dose, drug A works better than drug B; at a higher dose, drug B works better)? Answering these questions — which requires disentangling the effects of the drug dose regimen — would yield valuable insights, and facilitate more informed decisions over which drugs to produce at scale, and which drugs should be offered to which patients.

Parallels with PLM research. It is straightforward to see a parallel between this (flawed) drug trial setup with the way PLM research in NLP is done today. As each new PLM differs from earlier ones in multiple dimensions, it is increasingly harder to disentangle how much each component (*e.g.,* objective function, model size, pre-training data amount) contributes to performance. This leaves us ill-equipped to answer questions like:

- How well would earlier PLMs in the literature work if we augment them with the latest techniques, such as using the latest hyper-parameter settings or training them on more data? Would they match the performance of newer PLMs?
- To what extent should we attribute each PLM's performance to its key novelty (*e.g.,* the *bidirectional* masked language modelling loss for BERT), as opposed to other factors like the size of the model or the size of its training data?
- Which pre-training approaches should we use under considerations where efficiency considerations are paramount, such as for low-resource languages with limited amounts of monolingual data or under compute resource constraints?[2]

Reasons for the scientific debt. While answering the questions above would drive better-informed progress in the field, in practice there are multiple barriers to doing so; to some extent, these barriers account for why this scientific debt arises in the first place. These include (i) variations in the choice of hyper-parameters and experimental settings, which can result in a large variance in model performance (Bouthillier et al., 2019; Dodge et al., 2020); (ii) the proprietary and opaque nature of many PLMs and their training datasets — especially large-scale ones — rendering standardization

[2] Data efficiency considerations are also important for understanding and "reverse-engineering" human language learners, who are able to acquire language proficiency with much less amounts of data than current PLMs need (Hart and Risley, 1995; Dupoux, 2018; Cristia et al., 2019; Linzen, 2020).

difficult; (iii) the increasing computational costs of large-scale PLMs, which increases the costs of running multiple pre-training experiments (*e.g.,* with different model sizes or training data); and (iv) a strong emphasis in the field for achieving state-of-the-art results — indirectly creating an incentive to spend all of one's compute, time, and effort to achieve the best results, albeit sometimes at the expense of scientific rigour, performing rigorous experiments under comparable conditions, tuning the baselines, and disentangling different possible sources of model improvements. We revisit these barriers, and outline our recommendations in §4.

The costs and benefits of scientific debt, and *why* we should address it. In practice, the community goes into scientific debt because there are certain benefits of doing so. Indeed, by rapidly sharing and publishing new progress and state-of-the-art models — even when we do *not yet* fully understand how each factor contributes to the final performance of the model — the community is able to share, use, and build on exciting findings (*e.g.,* more accurate and faster models, etc.) much more quickly at a time of rapid progress. Yet on the other hand, accumulating too much scientific debt — without a good plan to address it and eventually pay it off — also carries an important risk: The lack of fair comparisons and proper ablations can lead the community down the wrong path, make sub-optimal choices, and waste precious community time, effort, and computational resources in the wrong direction. Paying off this scientific debt would crucially (i) enable the community to direct our collective efforts and resources into the research directions that matter the most for improving model performance, (ii) understand what factors enable PLMs' remarkable success today, and (iii) better comprehend the trade-offs between different approaches under various types of settings.

Large variability in current PLMs. To illustrate the extent of this issue, we summarize several key design choices behind some well-known PLMs in Table 3 (Appendix A), revealing a large *variability* in the key design choices (*e.g.,* model size, training data corpus and size, subword pre-processing algorithm, pre-training task, etc.) behind each PLM. Some common patterns include scaling the model while also using different, often larger pre-training data, as well as using different training regimes altogether. As each design choice impacts model performance in different ways (Sennrich and Zhang, 2019; Jiao et al., 2019) — combined with the fact that not all prior work conducted thorough ablations to understand how each component affects overall performance — it has become increasingly difficult to understand *why* a PLM outperforms the baselines, *which* design choices should be used under what settings, and *how much* of the improvement can be attributed to each work's novelty.

3 Empirical Evidence: The Case of BERT

Having motivated the importance of better disentangling the impact of different design choices in PLM research, we conduct experiments in pursuit of this goal. These experiments serve to (i) further our understanding of the effectiveness of the masked language modelling *objective* (Devlin et al., 2019), compared to alternative approaches under comparable conditions; (ii) demonstrate how these experiments can yield new insights; and (iii) form the basis for our recommendations and lessons learnt for accelerating progress in this line of work.

At the time of its release, BERT (Devlin et al., 2019) attracted a lot of attention by virtue of its strong performance on many tasks, outperforming earlier PLMs like ELMo (Peters et al., 2018) and GPT-1 (Radford et al., 2018) by substantial margins. At its core, BERT combines the following:

- Language model (LM) pre-training on large amounts of unlabelled data (Dai and Le, 2015; Peters et al., 2018; Radford et al., 2018; Howard and Ruder, 2018).
- Conducting *whole-model fine-tuning* (Radford et al., 2018) on each downstream task, as opposed to only using the resulting contextual word representations (*i.e.,* the neural model's frozen hidden state vectors) as features for a downstream task, as was done in the case of ELMo.
- Like the GPT-1 model, using a Transformers (Vaswani et al., 2017) architectural backbone, as opposed to (bidirectional) LSTMs (Hochreiter and Schmidhuber, 1997) in the case of ELMo.
- Using a novel *bidirectional* masked language modelling loss, which predicts the identity of each masked token by attending to *both* its left and right context,[3] rather than only the left/right context as in GPT-1 and ELMo pre-training.

[3] The bidirectional attention used within BERT is enabled by the use of Transformer architectures, which — unlike LSTM models — have no inherent directionality constraints.

Based on the different components of BERT above, it combines previous known techniques with a key novelty: the masked language modelling objective, which — unlike prior approaches like ELMo and GPT-1 — enables it to leverage and fuse bidirectional context at pre-training. Nevertheless, BERT differs from its prior approaches on many *other* factors (*e.g.,* the pre-training data corpus and size, model size, using Transformers vs LSTMs, length of the training cycle, tokenizer, etc.). This renders a principled comparison difficult, and makes it hard to *isolate* the importance of the masked language modelling objective from other factors that also affect model performance. We therefore here ask:

- To what extent can we attribute BERT's superior performance over its GPT-1 and ELMo baselines to its key novelty (*i.e.,* the bidirectional masked LM loss), as opposed to other design choices?
- Can the baseline models achieve similar performance with BERT, if we augment them with a similar set of design choices that the BERT model used (*e.g.,* whole model fine-tuning, using Transformers as opposed to LSTMs, etc.)?
- Can we come up with simpler variants of the baseline models, which can approximate the performance of more sophisticated approaches?
- How exactly would the findings change in the case where pre-training efficiency considerations are paramount (*e.g.,* where there is a more limited amount of pre-training compute available)?

3.1 Experimental Setup

We aim to isolate the importance of BERT's bidirectional masked LM objective, in comparison to two prior baselines: ELMo and GPT-1, under *comparable* experimental conditions. We compare our experimental setup with each model's original pre-training configuration in Table 4 (Appendix B).

Training data. For all three models, we use the original BERT pre-training data (Devlin et al., 2019), containing a combination of Wikipedia[4] and BookCorpus (Zhu et al., 2015). This dataset — which is larger than either of the ELMo or GPT-1 pre-training dataset[5] — consists of $\sim$ 3.3B words.

Model. We use a Transformer backbone for all three models, which has been shown to outperform LSTMs, and is more amenable to scaling to larger training datasets. Concretely, we use a BERT-Base architectural backbone, as implemented on HuggingFace[4] (Wolf et al., 2020), with $\sim$ 110 million parameters. Whereas the underlying model architectures are identical across all models, the pre-training objective function is naturally tailored to each approach, *e.g.,* masked LM for BERT, causal / left-to-right LM for GPT-1, and two independent causal LMs for ELMo: One operating in a left-to-right fashion, and another operating right-to-left.

The implementation of the different pre-training objectives requires two changes. First, when applicable, we update the "input mask" function on each attention layer (*e.g.,* using a standard causal attention mask in the case of GPT-1 to enforce a left-to-right directionality constraint). Second, we change the loss function to reflect each pre-training objective. For instance, we predict each next word conditional on its *left* context for GPT-1; for BERT, we predict the $\sim$ 15% masked words conditional on the (slightly corrupted) *bidirectional* context. One key difference is that our BERT implementation excludes the next-sentence prediction (NSP) pre-training loss, in accordance with the findings of Liu et al. (2019).[6] All other variables (*e.g.,* dataset, hyper-parameter choices, etc.) are kept identical across all models to facilitate a fair comparison.

Pre-processing. We use the same WordPiece tokenization as the original BERT model. We follow the procedure of Liu et al. (2019) for sampling sequences for each batch, where the input is constructed by repeatedly sampling multiple sentences until we reach the maximum sequence length of 512, while respecting document boundaries.

Fine-tuning. We use whole-model fine-tuning (Radford et al., 2018) for all models, which works better than the feature-based contextual word embedding approach of the original ELMo. As is standard practice, we take the top-layer contextual embedding of the [CLS] token to represent the whole sequence when fine-tuning BERT; for the left-to-right GPT-1 rerun, we take the top-layer contextual embedding of the *last token*, where the model has

[4] All model and data license information is in Appendix E.

[5] As pre-training data size and quality have been shown to be an important factor of LM success (Liu et al., 2019; Hoffmann et al., 2022), we hypothesize that BERT's larger pre-training data is an important factor behind its success compared to prior approaches — independently of the masked LM objective.

[6] By using the same codebase for all three models, we eliminate confounds arising from minor technical differences, such as whether or not to use segment embeddings (as BERT does), what kind of positional encoding schemes are used, what vocabulary size and subword preprocessing algorithms are used, and how each batch of sequences is sampled.

observed the entire sequence.[7] For each GLUE task, we run a grid search over 7 fine-tuning learning rates, 2 batch sizes, and 3 random seeds (Appendix B). We submit the best-performing model on the validation set to the GLUE leaderboard.

ELMo rerun. Following Peters et al. (2018), we pre-train ELMo by independently pre-training two separate, causal / unidirectional models: a left-to-right one and a right-to-left one.[8] At fine-tuning, Peters et al. (2018) combined the output layer of the left-to-right and right-to-left models, and used that combination as the representation of the whole sequence, based on which the fine-tuning cross-entropy loss is then calculated. In contrast, we employ a slight modification of ELMo where we first calculate the probability of each downstream task label under the left-to-right and right-to-left models, denoted as $p_{\boldsymbol{\theta}}^{\text{L2R}}(y \mid \mathbf{x})$, and $p_{\boldsymbol{\psi}}^{\text{R2L}}(y \mid \mathbf{x})$, respectively; y denotes the downstream task label for a sequence $\mathbf{x}$, while $\boldsymbol{\theta}$ and $\boldsymbol{\psi}$ denote the parameters of the left-to-right and right-to-left models, respectively. The aim of ELMo fine-tuning is to find fine-tuned (FT) ELMo parameters $\{\boldsymbol{\theta}_{\text{FT}}^{\star}, \boldsymbol{\psi}_{\text{FT}}^{\star}\}$:

$$\boldsymbol{\theta}_{\text{FT}}^{\star}, \boldsymbol{\psi}_{\text{FT}}^{\star} \overset{\text{def}}{=} \arg\min_{\boldsymbol{\theta}, \boldsymbol{\psi}} \sum_{(\mathbf{x}, y) \in D} - \log \left(\lambda \, p_{\boldsymbol{\theta}}^{\text{L2R}}(y \mid \mathbf{x}) + (1 - \lambda) \, p_{\boldsymbol{\psi}}^{\text{R2L}}(y \mid \mathbf{x}) \right) .$$

We simply set the interpolation coefficient $\lambda = 0.5$ for all tasks. We find this variant of ELMo to perform better than the original ELMo formulation by a small margin ($\sim 0.5\%$ in aggregate GLUE validation performance) in our preliminary experiments.

Hyper-parameters. We summarize the pre-training and fine-tuning hyper-parameters for each model in Appendix B. We spend a similar amount of compute in tuning the hyper-parameters of each model, facilitating a fair comparison across models.

Evaluation Tasks. We evaluate each model on the GLUE classification tasks (Wang et al., 2018), as done in the original BERT paper. We leave other evaluation benchmarks like SuperGLUE (Wang et al., 2019) to future work, although due to BERT's bidirectional nature, conducting generative evaluation that requires sampling text from BERT is non-trivial (Wang and Cho, 2019; Goyal et al., 2022).

Compute. Pre-training each model took 5 days with 8 V100 GPUs, while it took roughly a day to run GLUE fine-tuning with 8 GPUs (42 fine-tuning hyper-parameters for each {model, task}, §3.1). All in all, we used 8700 GPU hours for pre-training and 1100 GPU hours for fine-tuning all models.

3.2 Empirical Findings

We summarize the GLUE test set results in Table 1, based on which we remark on four observations.

- Under comparable conditions, the test GLUE performance of our variant of ELMo is at 76.8%, representing a relatively small 1.2% gap with the standard BERT rerun (78%). This improved performance represents a vast, $> 6\%$ improvement from the original ELMo's reported result of 70.3%; we attribute this gap to the use of a larger, BERT-equivalent pre-training data, a Transformer backbone, and whole model fine-tuning. We also see a small gain of our GPT-1 Rerun — an improvement we attribute to a larger and better pre-training dataset than the original model. Hence, we conclude that most of the substantial $> 8\%$ gap between the original BERT and ELMo results can, in fact, be attributed to using Transformers rather than LSTMs, conducting whole model fine-tuning, and using a larger pre-training data, rather than BERT's bidirectional masked LM loss in and of itself. Altogether, this result reaffirms how augmenting the baselines with a similar set of techniques and advances as later generation models can substantially improve their performance, and yield results that are close to those of more recent models (Melis et al., 2018; Merity, 2019; Lei, 2021, *inter alia*).
- However, there remains a larger gap between the causal / unidirectional PLMs and BERT; this holds for both the left-to-right (2.6% worse than BERT) and the right-to-left (3.4% worse) models. Note, however, that this 2.6% gap under comparable conditions is smaller than the original reported GPT-1 result, which had a $> 4\%$ gap with the original BERT — a result we attribute to the smaller dataset used to pre-train the original GPT-1. These findings further emphasize the importance of comparing the baselines and more recent PLMs under comparable conditions.
- Although the left-to-right and right-to-left models still lag behind BERT, a simple *ensemble* of two independently pre-trained and fine-tuned left-to-right and right-to-left models can never-

[7] By the same logic, we take the top-layer contextual embedding of the *first* token to represent the whole sequence instead when fine-tuning the right-to-left GPT-1.

[8] Note that there is no need for cross-model communication when pre-training the left-to-right and right-to-left ELMo models; hence this ELMo pre-training stage can be done completely in parallel, *e.g.,* on two completely different machines.

theless approach BERT's and ELMo's performance (76.3% for the ensemble, 78% and 76.8% for BERT and ELMo reruns, respectively). Remarkably, we do not observe the same gains when ensembling two left-to-right models from different random seeds, suggesting that ensembling unidirectional models with different directionalities is crucial for performance. All in all, these results highlight the need to explore *simpler baselines*, which can approximate the performance of more sophisticated approaches.[9]

- As shown in Table 2, when efficiency considerations are paramount (*i.e.,* where each model is only pre-trained for 200k steps, and not the full 1M), the performance gap between the left-to-right GPT-1 rerun and BERT nearly vanishes (0.5% gap, as opposed to a 2.6% gap in the 1M-pre-training-steps setup). We attribute this to the fact that BERT only uses $\sim$ 15% of the tokens in a batch as the masked LM target, whereas the left-to-right LM can leverage all 100% of the tokens as pre-training supervision in a similar fashion as Electra (Clark et al., 2020), hence resulting in more efficient learning. Remarkably, despite its simplicity, an ensemble of independently-pre-trained-and-fine-tuned left-to-right and right-to-left models **outperforms** the BERT model by 1.1% in this efficient learning scenario. This finding (i) suggests that approaches that work best in the high-data/compute scenario may not necessarily transfer to cases where efficiency considerations are paramount; and (ii) highlights the need to train, evaluate, and ultimately *benchmark* models in efficient learning scenarios, such as in languages where monolingual data are not abundant or in cases where there is only a limited amount of compute available for pre-training.

Validation set results with error bars. To preserve test set integrity, we only submitted the single-best validation model to the test set (Table 1). In Appendix D, we report the validation set performance that includes error bars over three different random seeds, which broadly show the same trend.

4 Paying Off the Scientific Debt: Recommendations and Lessons Learnt

We proceed to outline several key recommendations and lessons learnt for encouraging, incentivizing, and accelerating progress in this line of work.

Establish standard, publicly available pre-training corpora at multiple data scales. As seen in §3, the size and quality of the pre-training data is an important driver behind model performance (Liu et al., 2019; Hoffmann et al., 2022), which makes a rigorous comparison between different PLMs difficult. Hence, our first recommendation is to establish standard pre-training corpora that are publicly available.[10] We further recommend releasing the pre-training corpora under multiple data scales, as approaches that work best under strict compute or data resource requirements may be different from the case where there is a large amount of compute and data available (§3, Clark et al., 2020; Treviso et al., 2022). Note that this does *not* mean that we are discouraging the use of non-standard or even-larger corpora than those that are publicly available. On the contrary, researchers *should* continue to push the boundaries of what is possible by training on more, better quality, and more recent data. In such cases, we recommend researchers to *also* release versions of their models that are trained on the standard pre-training corpora — above and beyond the version trained on proprietary & large-scale data that would presumably be necessary to achieve a new state-of-the-art — in order to facilitate a fair and principled comparison with prior work. We encourage the community to *continually* release new standardized pre-training datasets as time passes to avoid the effect of pre-training data staleness (Lazaridou et al., 2021).

Explicitly delineate the different types of contributions behind each work, including both the key novelty and engineering contributions. We recommend that PLM research explicitly state the key novelty behind each work (*e.g.,* the bidirectional masked LM loss for BERT), delineate and explicitly state other contributions (including engineering ones) and design choices that can impact performance, and outline how these differ from prior work (*e.g.,* better model partitioning for train-

[9] Ensembling independently pre-trained and fine-tuned left-to-right and right-to-left models is a *late-fusion* approach, without the need for ELMo's *joint* fine-tuning stage.

[10] The establishment of standardized corpora (along with their corresponding training / validation / test splits) have been a standard feature of statistical NLP for the last decades. To that end, researchers would compare models that are trained and evaluated on the same datasets to enable a fair comparison, such as the Penn Treebank (Marcus et al., 1993) for parsing or SQuAD for question answering (Rajpurkar et al., 2016). But pre-trained foundation models (Bommasani et al., 2021) introduce an additional confound: two PLMs that are fine-tuned on the exact same task and dataset can have different performance simply because one of them is *pre-trained* on more data, irrespective of the novelty of each PLM.

Model	CoLA	MNLI(-m)	MRPC	QNLI	QQP	RTE	SST-2	STS-B	Avg
Original published results									
BERT	52.1	84.6	88.9	90.5	71.2	66.4	93.5	85.8	79.1
BERT Large	60.5	86.7	89.3	92.7	72.1	70.1	94.9	86.5	81.6
GPT-1	45.4	82.1	82.3	87.4	70.3	56.0	91.3	80.0	74.4
BiLSTM + ELMo + Attn	36.0	76.4	84.9	79.8	64.8	56.8	90.4	73.3	70.3
Our replication with proper controls & comparable experimental conditions									
BERT Rerun	50.8	84.5	89.0	90.5	71.0	61.0	93.1	84.4	78.0
Comparable GPT-1 Rerun - L2R	41.6	87.4	84.7	86.6	68.8	62.9	91.8	79.3	75.4
Comparable GPT-1 Rerun - R2L	42.5	82.0	85.5	88.3	69.1	57.6	92.8	79.1	74.6
Comparable ELMo-variant Rerun	46.8	83.6	85.8	89.9	70.8	61.9	93.1	82.1	76.8
Ensemble of Comparable GPT-1: L2R + R2L	45.1	83.7	85.8	88.9	70.8	62.4	92.9	81.0	76.3
Ensemble of Comparable GPT-1: L2R + L2R	42.4	83.5	85.1	87.8	70.0	63.1	93.1	79.9	75.6

Table 1: GLUE **test** results. We use F1 scores for MRPC and QQP, Matthew's Correlation for CoLA, SpearmanR for STS-B, and accuracy for the rest; all models are pre-trained with the same batch size & compute (1M steps).

Model	CoLA	MNLI(-m)	MRPC	QNLI	QQP	RTE	SST-2	STS-B	Avg
BERT Rerun	43.8	80.9	86.4	87.9	69.3	59.3	90.0	80.4	74.8
Comparable GPT-1 Rerun: L2R	43.5	80.3	84.1	86.3	68.4	63.0	91.0	77.8	74.3
Comparable GPT-1 Rerun: R2L	36.2	80.6	82.4	88.2	68.7	53.7	93.0	77.8	72.6
Ensemble of GPT-1 Rerun: L2R + R2L	45.1	82.6	84.4	88.3	70.8	62.9	93.5	79.9	75.9

Table 2: GLUE test set results using the pre-trained model, after training for 200,000 steps followed by fine-tuning.

ing larger models on multiple devices, better filtering of the training data, more extensive hyper-parameter tuning, etc.). Combined with strong baselines and extensive ablations (see below), this will enable us to better understand *how much* of the performance gains can be attributed to each factor, including the key novelty behind the approach.

Invest comparable effort into tuning both the baselines *and* the newly-proposed models. In practice, many of the contributions (*e.g.,* better hyper-parameters, better data, etc.) would also be applicable to the baselines. We recommend each PLM work to discern which of their design choices can *also* be applied to the baselines, and apply those techniques in order to create stronger baselines that may nevertheless rival the performance of more recent models (§3, Melis et al., 2018; Lei, 2021).

More extensive ablation studies. When proposing multiple contributions at once (as many PLM papers do), we recommend conducting as many ablation studies as is feasible to isolate the impact of each component under comparable conditions. In light of recent trends where models — including open-sourced ones — are publicly released without technical reports or papers that outline technical details regarding model evaluation and benchmarking (Taori et al., 2023; Chiang et al., 2023), we argue that our recommendation for conducting more thorough evaluations is even more critical.

Better credit assignment is needed. As shown in Table 1, the vast gap between BERT and ELMo can nearly be bridged by using (i) the same (larger) pre-training data, (ii) Transformer architectures, and (iii) whole model fine-tuning; all of which were already used and proposed by the GPT-1 model. As these techniques account for a more significant chunk of the performance difference than the bidirectional masked LM loss, disentangling each factor's contribution thus provides an opportunity to conduct better credit assignment in the field.

Strike a balance between pushing the state-of-the-art and advancing our scientific understanding. In some sense, recent rapid progress is made possible by a strong emphasis on building the next state-of-the-art PLMs and foundation models, although it comes at a cost of understanding — from the scientific point of view — where the performance improvements are coming from, and which techniques work best under what circumstances. We argue that *both* lines of work — one that pushes the state-of-the-art at breakneck speed and through all available means, and another that aims to resolve the scientific and technical debt by disentangling the impact of multiple factors of model improvements (which we argue is still currently underrepresented in the field) — should be conducted, encouraged, and rewarded within the field. We outline two concrete recommendations for striking a better balance between the two lines of work. First, public release of PLMs or their downstream applications should be *promptly* accompanied by a technical description of the model, ideally in the form of a technical report or a scientific paper.[11] This would enable the community to better understand the key component behind these models' success, allow future work to replicate the results, and promptly disentangle the different components behind model improvements. Second, we as a community should not necessarily expect *both* types of contributions under the same paper. Just like how the cleaning up of technical debt happens *after* the initial code has been written, it is often the case that prior work that resolves the scientific debt through principled comparisons was only conducted after substantial progress in advancing the state-of-the-art (often through all available means for improving model performance) had been made. We should, however, encourage the community to conduct such understanding line of work promptly after major milestones or exciting results.

Reward and encourage a line of work that focuses on understanding (not just those that chase a new state-of-the-art), even when they are imperfect. The current, rapid pace of the field provides an incentive to spend one's (finite) computational resources and effort for building the next state-of-the-art, albeit at the expense of scientific rigour and principled comparisons. Given a finite amount of compute, there is arguably more incentive in tuning one's proposed approach through all possible means (*e.g.,* using larger datasets and larger models, training for longer, etc.), topping the leaderboards, and publishing the paper, even if this leaves no computational resources to tune the baselines and conduct rigorous ablations. Furthermore, the rapidly increasing cost of training ever-larger PLMs means that any principled comparisons are most likely imperfect (§7) — *e.g.,* how do our findings in §3 change with models that are trained for longer, like RoBERTa? Or with encoder-decoder models like T5? Or in other languages? Indeed, our experiments in §3 are fairly narrow in scope, involving only three non-recent models (BERT, ELMo, GPT-1) and a training dataset that is small by today's standards. Yet due to the rigorous hyper-parameter tuning of all three models, conducting these principled comparisons required an enormous amount of compute resources — equivalent to training 10 BERTs from scratch. This cost would have been even higher with the inclusion more models, languages, and larger datasets. On this point, we remark that doing such principled comparisons — even when they are limited in scope and done on smaller models — *still* contributes towards paying off the scientific debt, better understanding where our current progress is coming from, and deriving valuable insights that can contribute to the development of next generation PLMs. We additionally call on those in our community who serve as reviewers to recognize and reward these types of research contributions, which are *complementary* (if perhaps equally important) to a parallel line of work that pushes the state-of-the-art in PLM research through all possible means.

We need more comprehensive PLM scaling laws. Our experiments and recommendations still leave a major open question: How can we scale these kinds of investigations to much larger PLMs, which are much more computationally expensive? To that end, **scaling laws** (Kaplan et al., 2020; Hoffmann et al., 2022) provide an account of how PLM performance changes with respect to different factors, allowing us to accurately extrapolate that a PLM with X parameters and Y training steps should achieve a perplexity of Z. However, we argue that current scaling laws are still overly narrow in scope: Concretely, existing scaling laws often only apply to decoder-only / unidirectional PLMs, and only provide an account of how their performance changes with respect to (i) model size and (ii) the number of training tokens. We call on the community to develop more comprehensive scaling laws that take into account and characterize how other factors impact LM performance and downstream behavior, including how model performance and behavior change with respect to the choice of the objective function and model hyper-parameters, and the quality of the pre-training data. The existence of such scaling laws — which can happen by pooling community data on various PLM pre-training runs and their corresponding perplexity and downstream performance — would allow other researchers to accurately *extrapolate* how their findings would generalize to other PLM model sizes, objective functions, etc. Most importantly, comprehensive scaling laws

[11] In some cases, the relevant technical descriptions were not promptly released following model release, such as the community project BLOOM (Scao et al., 2022) or ChatGPT (whose blog post does not cover much technical detail.).

can disentangle and *quantify* how these different factors contribute to determine the final model performance under various experimental conditions.

How conducting rigorous experiments and ablation studies can lead to new state-of-the-art results. Lastly, we argue that conducting rigorous experiments and ablation studies for paying off the scientific debt should *not* necessarily come at the expense of achieving a new state-of-the-art. In contrast, doing so can be a key ingredient for building the next state-of-the-art PLMs. In 2020, Kaplan et al. (2020) proposed a seminal scaling law that showed how larger PLMs are more sample-efficient, and that one should always increase model size when increasing the pre-training compute budget, leading the community to develop ever-larger PLMs in response (Rae et al., 2021; Smith et al., 2022, *inter alia*). Nevertheless, subsequent rigorous experiments from Hoffmann et al. (2022) demonstrated that the optimal pre-training compute allocation should, in fact, *also* be scaled in another dimension: The amount of pre-training data that the model is trained on. This insight was then used to build smaller, more efficient, and cheaper-to-run PLMs that, at the time of its release, achieved new state-of-the-art results that outperformed much larger PLMs that were under-trained in comparison. Going forward, we conjecture that rigorous experiments and ablation studies that look at factors *above and beyond* model size and data quantity, such as the *quality* of the pre-training data, the exact hyper-parameters, the pre-training objective, etc., will not only be useful to understand how these factors improve performance and thus pay off the scientific debt, but also form a key ingredient for building the next generation of better PLMs.

5 Related Work

A number of prior work has made progress in disentangling the impact of different language modelling pre-training objectives by conducting principled ablation studies under comparable experimental conditions (Dong et al., 2019; Raffel et al., 2020; Tay et al., 2022; Artetxe et al., 2022, *inter alia*). However, some of the recently released models do not provide any technical details on how they are trained, such as ChatGPT[12] and GPT-4 (Bubeck et al., 2023). We discuss these in an extended related work section (Appendix C), but briefly remark on how our findings complement theirs. First, we revisit and augment ELMo — which incorporates a degree of bidirectionality at fine-tuning (albeit not at pre-training) — with Transformers and whole model fine-tuning, facilitating a fair comparison with BERT. We show that the resulting ELMo achieves competitive performance with BERT on GLUE; to our knowledge, no such ELMo baseline — or an even simpler ensemble of a left-to-right and right-to-left PLM — was explored in prior work.

While our work shares several similarities with (Melis et al., 2018), our work differs by virtue of being a position paper that focuses on an important issue in the field (*i.e.,* the lack of fair comparisons between past PLMs), and chart the way forward for mitigating this issue. Our experiments in this work mostly aim to provide an example of this issue in action, and form the basis for some of the lessons learnt and recommendations that we outline in §4. Unlike Melis et al. (2018), our experiments are not designed to achieve new state-of-the-art results. Moreover, above and beyond our empirical contributions, we outline key recommendations that would encourage and incentivize this line of work; we hope that these recommendations would be adopted by the broader community, with the aim of accelerating progress towards resolving the scientific debt in foundation model research.

6 Conclusion

Recent rapid progress within the PLM literature has led to tremendous advances within NLP. Despite this progress, current PLM research practices that change multiple different things at once — often without proper ablation studies and conducting principled comparisons that disentangle the impact of different components — have introduced certain issues that we call "scientific debt". Through experiments that disentangle the contribution of BERT's bidirectional masked LM objective through principled comparison with prior work, we demonstrate how asking "*which factors contribute the most to the model performance that we observe today*?" can lead to valuable new insights, including the existence of simple yet stronger-than-expected and more efficient baselines. We outlined several recommendations that would encourage and incentivize this line of work that aims to better understand how each factor contributes to the rapid progress of our PLMs today, and better address the ongoing issue of accumulating scientific debt within our current PLM research literature.

[12] The model inference is accessible via OpenAI API.

7 Limitations

Our work has the following limitations.

Comparisons with more recent models. In §3, we conducted a principled comparison between BERT, ELMo, and GPT-1 under comparable experimental conditions. This comparison notably excludes more recent models that benefit from more parameters, larger training data, or different loss functions, such as RoBERTa, Electra, and T5. Due to the even-higher cost of pre-training these more recent models, we leave a principled comparison that includes these models to future work, although we identified the development of more comprehensive PLM scaling laws as a promising future research direction that would allow us to extrapolate how our findings would generalize to different pre-training data sizes, objective functions, etc. (§4).

Interaction between different factors. In §3, we have conducted a principled comparison by varying only the pre-training objective function and the length of model training, whilst keeping all the other variables constant. In practice, however, the exact choice of these different control variables (*e.g.,* what positional encodings to use, how we pre-process the data, etc.) can *interact* and affect the findings in a material way. It is conceivable — and rather likely — that our findings on the performance gap between BERT, ELMo, and GPT-1 may change under different experimental settings.

Simulated efficient learning scenario. Our efficient learning scenario in §3 constitutes a simulated one, where we artificially limit the number of updates to 200,000 steps (as opposed to 1M steps in the full setting). We leave the extension to more realistic efficient learning scenarios, such as in languages where there is only a limited number of monolingual data, or where there is a hard limit on what pre-training computational resources we can use (*e.g.,* 1 GPU for 3 days), to future work.

Extension to multi-lingual settings. Our experiments are thus far conducted only in English. We leave the extension to other languages — including low-resource languages with only a limited amount of monolingual data as a realistic and necessary benchmark of efficient learning — to future work.

The increasing prevalence of closed-source / proprietary PLMs. Despite our recommendations and calls for change, we acknowledge the fact that recent PLM trends have shifted more towards proprietary and closed-source models — a development we attribute to the rapidly increasing commercialization potential of this technology. Under this trend, very little is known about how each PLM is developed, as the vast majority of the technical details (*e.g.,* the amount and source of the pre-training data, the data filtering strategy, the size and hyper-parameters of the model, how the model is implemented, etc.) are kept proprietary. While these trends may mean that our recommendations are more unlikely to be adopted by proprietary PLMs, we argue that our position paper and recommendations are still important (if not even more so) for two reasons. First, open-sourced community models, such as BLOOM (Scao et al., 2022), OPT (Zhang et al., 2022), and Alpaca (Taori et al., 2023), are gaining traction, and have rapidly narrowed the gap with proprietary models. This progress reflects the community's strong desire to have open-sourced models that can rival proprietary ones in terms of model quality. The rise of these open-sourced models thus gives rise to the question: How can these community-driven models help the community pay off our scientific debt? To that end, our recommendations provide concrete and actionable steps in this direction. For instance, our recommendations call for standardizing the pre-training dataset, which has not yet been done thus far, even though there are plausible, open-sourced datasets that can be used for doing so. Furthermore, we also encourage the community to release the full evaluation results of their models, alongside the relevant hyper-parameter information, etc., such that we can *collectively* build a more comprehensive scaling law through crowd-sourcing (§4). Second, prior work that conducts extensive ablation studies and rigorous experiments (Raffel et al., 2020; Sun and Iyyer, 2021, *inter alia*) remains the exception, rather than the rule. Our position paper includes a call for change that will make it *easier* to pay off this scientific debt going forward, which is ever-more important in light of impressive progress from both proprietary and open-sourced PLMs.

Ethical Considerations

Our experiments replicate prior work under comparable experimental conditions. For this reason, we do not expect our work to introduce any novel ethical issues, although our experiments may inherit a similar set of issues concerning PLM (especially large-scale ones), as outlined by various prior work (Gehman et al., 2020; Bender et al., 2021; Rae et al., 2021; Dinan et al., 2021; Bommasani et al., 2021; Kenton et al., 2021; Weidinger et al., 2021, *inter alia*). We remark, however, that conducting these principled comparisons across different models — which requires a degree of hyper-parameter tuning for each model (both at pre-training and fine-tuning stages) in order to enable a fair comparison — requires a large number of computational resources, which may contribute to increased carbon emissions (Strubell et al., 2019; Patterson et al., 2021).

Acknowledgement

We thank Chris Dyer, John T. Hale, and Laura Rimell at DeepMind, Noah A. Smith at the University of Washington & Allen Institute for Artificial Intelligence, and Qi Huang at Bloomberg for their valuable insights, feedback, and suggestions.

References

Mikel Artetxe, Jingfei Du, Naman Goyal, Luke Zettlemoyer, and Ves Stoyanov. 2022. On the role of bidirectionality in language model pre-training. In *Findings of the Association for Computational Linguistics: EMNLP 2022*.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proc. of FAccT '21*. Association for Computing Machinery.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri S. Chatterji, Annie S. Chen, Kathleen Creel, Jared Quincy Davis, Dorottya Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah D. Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark S. Krass, Ranjay Krishna, Rohith Kuditipudi, and et al. 2021. On the opportunities and risks of foundation models. *CoRR*, abs/2108.07258.

Xavier Bouthillier, César Laurent, and Pascal Vincent. 2019. Unreproducible research is reproducible. In *International Conference on Machine Learning*, pages 725–734. PMLR.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

Alejandrina Cristia, Emmanuel Dupoux, Michael Gurven, and Jonathan Stieglitz. 2019. Child-directed speech is infrequent in a forager-farmer population: A time allocation study. *Child development*, 90(3).

Andrew M Dai and Quoc V Le. 2015. Semi-supervised sequence learning. In *Advances in Neural Information Processing Systems*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Emily Dinan, Gavin Abercrombie, A. Stevie Bergman, Shannon L. Spruit, Dirk Hovy, Y-Lan Boureau, and Verena Rieser. 2021. Anticipating safety issues in E2E conversational AI: framework and tooling. *CoRR*, abs/2107.03451.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305*.

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. In *Proc. of NeurIPS*.

Emmanuel Dupoux. 2018. Cognitive science in the era of artificial intelligence: A roadmap for reverse-engineering the infant language-learner. *Cognition*.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. 2020. RealToxicityPrompts: Evaluating neural toxic degeneration in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*.

Kartik Goyal, Chris Dyer, and Taylor Berg-Kirkpatrick. 2022. Exposing the implicit energy networks behind masked language models via metropolis–hastings. In *Proc. of ICLR*.

Betty Hart and Todd R. Risley. 1995. *Meaningful Differences in the Everyday Experience of Young American Children*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

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

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2019. Tinybert: Distilling bert for natural language understanding. *arXiv preprint arXiv:1909.10351*.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8.

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. *arXiv preprint arXiv:2001.08361*.

Zachary Kenton, Tom Everitt, Laura Weidinger, Iason Gabriel, Vladimir Mikulik, and Geoffrey Irving. 2021. Alignment of language agents. *CoRR*, abs/2103.14659.

Angeliki Lazaridou, Adhiguna Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Sebastian Ruder, Dani Yogatama, Kris Cao, Tomás Kociský, Susannah Young, and Phil Blunsom. 2021. Mind the gap: Assessing temporal generalization in neural language models. In *Proc. of NeurIPS*.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022. Deduplicating training data makes language models better. In *Proc. of ACL*.

Tao Lei. 2021. When attention meets fast recurrence: Training language models with reduced compute. In *Proc. of EMNLP*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Tal Linzen. 2020. How can we accelerate progress towards human-like linguistic generalization? In *Proc. of ACL*.

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*.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2).

Gábor Melis, Chris Dyer, and Phil Blunsom. 2018. On the state of the art of evaluation in neural language models. In *Proc. of ICLR*.

Stephen Merity. 2019. Single headed attention RNN: stop thinking with your head. *CoRR*, abs/1911.11423.

David A. Patterson, Joseph Gonzalez, Quoc V. Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David R. So, Maud Texier, and Jeff Dean. 2021. Carbon emissions and large neural network training. *CoRR*, abs/2104.10350.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

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

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. 2021. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proc. of EMNLP*.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Rico Sennrich and Biao Zhang. 2019. Revisiting low-resource neural machine translation: A case study. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 211–221.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*.

Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, Elton Zheng, Rewon Child, Reza Yazdani Aminabadi, Julie Bernauer, Xia Song, Mohammad Shoeybi, Yuxiong He, Michael Houston, Saurabh Tiwary, and Bryan Catanzaro. 2022. Using deepspeed and megatron to train megatron-turing NLG 530b, A large-scale generative language model. *CoRR*, abs/2201.11990.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proc. of ACL*.

Simeng Sun and Mohit Iyyer. 2021. Revisiting simple neural probabilistic language models. In *Proc. of NAACL-HLT*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

Yi Tay, Mostafa Dehghani, Vinh Q Tran, Xavier Garcia, Dara Bahri, Tal Schuster, Huaixiu Steven Zheng, Neil Houlsby, and Donald Metzler. 2022. Unifying language learning paradigms. *arXiv preprint arXiv:2205.05131*.

Marcos Treviso, Tianchu Ji, Ji-Ung Lee, Betty van Aken, Qingqing Cao, Manuel R. Ciosici, Michael Hassid, Kenneth Heafield, Sara Hooker, Pedro H. Martins, André F. T. Martins, Peter Milder, Colin Raffel, Edwin Simpson, Noam Slonim, Niranjan Balasubramanian, Leon Derczynski, and Roy Schwartz. 2022. Efficient methods for natural language processing: A survey.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Alex Wang and Kyunghyun Cho. 2019. BERT has a mouth, and it must speak: BERT as a Markov random field language model. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, Zac Kenton, Sasha Brown, Will Hawkins, Tom Stepleton, Courtney Biles, Abeba Birhane, Julia Haas, Laura Rimell, Lisa Anne Hendricks, William S. Isaac, Sean Legassick, Geoffrey Irving, and Iason Gabriel. 2021. Ethical and social risks of harm from language models. *CoRR*, abs/2112.04359.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Proc. of NeurIPS*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

A Model Comparison

In Table 3, we outline a summary of various aspects of some commonly-used PLMs that have been proposed to date, summarizing the size of the model, the training data and its size, the text pre-processing scheme, the pre-training objective, and some hyper-parameter details. This table reveals a large variation in the design choices of these PLMs, hence rendering it difficult to conduct apple-to-apple comparisons between different approaches. Common patterns include scaling the model while also using different (often larger) pre-training data, as well as using different training regimes altogether. Each design choice impacts model performance in different ways (Sennrich and Zhang, 2019; Jiao et al., 2019), emphasizing the importance of conducting thorough ablations and apple-to-apple comparisons.

B Detailed Hyper-Parameters

When conducting experiments, we follow BERT's architecture, training data, and overall hyper-parameter choices (Devlin et al., 2019). One noticeable difference, however, is that we follow RoBERTa to train the model without using a next-sentence prediction loss (Liu et al., 2019), which has been shown to have a minimal impact on model performance. Each of the three models in our rerun not only has the exact same design choices in terms of training data, text processing, model architecture, etc., but is also implemented on the exact same codebase. Concretely, we use the BERT codebase as implemented on Huggingface, and conduct some slight modifications in terms of removing the next-sentence prediction loss as stated above. When implementing the other models, we take the BERT implementation, and simply change the masking function and pre-training objective in order to replicate the results of GPT-1 and ELMo under comparable conditions as our BERT model. This means that our reruns of the GPT-1 and ELMo models benefit from the exact same technical implementation details as our BERT model by virtue of using identical positional encoding, segment embeddings, etc. We tune the pre-training and fine-tuning learning rate of each model independently (hence the final learning rate for each model may be different), although we strive to dedicate the same amount of compute resources in tuning the hyper-parameters of each model, in order to avoid favoring one model over the others. We summarize some key design choices in Table 4.

Model Size	Training Data (Size)	Text Pre-processing	Pre-training Objective	Setup
BERT (Devlin et al., 2019) ♢				
Smallest: 110M Largest: 340M	English Wikipedia, Book Corpus (16 GB)	Wordpiece (30k tokens)	MLM and NSP simultaneously	**Optimizers**: 1e-4, Adam, Linear Decay **Batch Size, Max Sequence Length**: 256, 512 **Steps/Epochs**: 1,000,000
RoBERTa (Liu et al., 2019) ♢				
Smallest: 125M Largest: 355M	English Wikipedia, Book Corpus CCNews, OpenWebText, Stories	BPE (50k tokens)	MLM	**Learning Rate**: (7e-4, 1e-4, 1e-3), Adam, Linear decay **Batch Size, Max Sequence Length**: 256, 512 **Steps/Epochs**: 1,000,000
Megatron-BERT (Shoeybi et al., 2019) ♢				
Smallest: 336M Largest: 3.9B	Wikipedia, RealStory Book Corpus, CC-Stories OpenWebText (174 GB)	Wordpiece (30k tokens)	MLM and NSP simultaneously	**Optimizers**: 1e-4, Adam, Linear Decay **Batch Size**: 1024 **Steps/Epochs**: 2,000,000
GPT-1 (Radford et al., 2018) ♠				
117M	1B Word Benchmark Book Corpus (∼ 9GB)	BPE (40k tokens)	CLM	**Optimizer**:2.5e-4, Adam, cosine annealing scheduler **Batch Size, Context Size**: 64, 512 **Steps/Epochs**: 100 epochs
GPT-2 (Radford et al., 2019) ♠				
Smallest: 110M Largest: 1542M	WebText (40 GB)	BPE (52k tokens)	CLM	**Optimizer**: tuned, unknown hyperparameter **Batch Size, Context Size**: 512, 1024
GPT-3 (Brown et al., 2020) ♠				
Smallest: 125M Largest: 172B	Expanded WebText Filtered CommonCrawl Internet Book Corpora English Wikipedia (499B tokens)	BPE (52k tokens)	CLM	**Optimizer**: Adam, Linear Warmup **Batch Size, Context Size**: Dynamic, 2048
Megatron-GPT (Shoeybi et al., 2019) ♠				
Smallest: 355M Largest: 8.3B	Wikipedia, RealStory, Book Corpus, CC-Stories OpenWebText (174 GB)	Follow GPT-2	CLM	**Optimizer**: 1.5e-4, Adam, warmup + cosine decay **Batch Size, Context size**: 512, 1024 **Steps/Epochs**: 300k steps
OPT (Zhang et al., 2022) ♠				
Smallest: 125M Largest: 175B	English Wikipedia, CC-Stories The Pile (deduped) PushShift.io Reddit CCNewsV2 (180B Tokens, 800GB)	Follow GPT-2	CLM	**Optimizer**: Adam, Linear scheduling **Batch Size, Context Size**: 0.5M to 4M tokens, 2048
T5 (Raffel et al., 2020) ♢♠				
Smallest: 220M Largest: 11B	C4 filtered (750 GB)	Wordpiece (32k tokens)	MLM on encoder/decoder, continued with prompted tasks	**Optimizer**: 0.01, Adafactor, insqrt scheduler **Batch Size, Max Sequence Length**: 128, 512 **Steps/Epochs**: 524,288
BART (Lewis et al., 2020) ♢♠				
Smallest: 140M Largest: 400M	English Wikipedia, Book Corpus CCNews, OpenWebText, Stories (160 GB)	BPE (50k tokens)	CLM with corrupted encoder input	**Optimizer**: tuned **Batch Size, Context Size**: 8000, tuned **Steps/Epochs**: 500,000
Gopher (Lewis et al., 2020) ♢♠				
Smallest: 44M Largest: 280B	Massive Web Books, C4, News, GitHub, Wikipedia (11 TB)	BPE (32k tokens)	CLM	User RMSNorm and relative positional encoding **Optimizer** Adam (different LR) **Batch Size, Context Size**: 0.25 M to 6M (depends on the model size), 2048 **Steps/Epochs**: unknown
Chinchilla (Lewis et al., 2020) ♢♠				
70B	Same as Gopher	BPE w/o NFKC-norm (32k tokens)	CLM	Similar to Gopher, but use AdamW as its optimizer
BLOOM (Scao et al., 2022) ♢				
Smallest: 560M Largest: 176B	ROOTS Corpus (1.6 TB)	Byte-level BPE (250k tokens)	CLM	Alibi positional embedding float16 precision training **Batch Size, Max Sequence Length**: 2048, 2048

Table 3: Comparison pre-training setup between popular language models. ♢♠: Encoder-Decoder Transformer. ♢: Encoder Transformer ♠: Decoder Transformer. MLM: Masked Language Modelling, NSP: Next Sentence Prediction, CLM: Causal Language Modelling. Note that what we put here is based on the paper for each language model.

B.1 Hyper-Paramaters for Fine-tuning

We refer to the original BERT's hyper-parameters for fine-tuning, but experiment with more fine-tuning learning rates. In addition to that, we also use three different random seeds. The fine-tuning hyper-parameters that we used are as follows, which is partially based on prior work (Joshi et al., 2020):

- **Batch sizes:** {16, 32}
- **Learning rates:** {2e-4, 1e-4, 5e-5, 3e-5, 2e-5, 1e-5, 5e-6}
- **Epoch:** 4
- **Random seeds**: $\{1, 41, 386\}$

C Extended Related Work

A number of prior work has made progress in disentangling the impact of different language modelling pre-training objectives by conducting principled, apple-to-apple comparisons under comparable experimental conditions; here we highlight four such prior work (among others), and remark on how our findings and recommendations complement theirs.

Raffel et al. (2020) introduced a unified, text-to-text format (*i.e.,* encoder-decoder) framework, and conducted a systematic study over the impact of different pre-training objectives, architectures, and training datasets. More recently, Tay et al. (2022) conducted a series of comprehensive ablation experiments that compared the effectiveness of different pre-training objectives under comparable conditions. They found that interpolation of these objectives can be universally effective across different tasks, setups, and model scales.[13] In line with our findings, their findings similarly demonstrate how conducting these systematic comparisons can lead to new insights and approaches that can rival or outperform current ones. Another line of work Artetxe et al. (2022) examined the role of *bidirectionality* in language model pre-training. This is done by drawing a distinction between bidirectional context and bidirectional attention, generalizing how current approaches fall with respect to these spectrums, and characterizing the effects of each component on different downstream tasks.

Our findings differ from — and further complement — this line of prior work in two ways. First, we revisit and augment the baseline ELMo model — which incorporates a degree of bidirectionality at fine-tuning (albeit not at pre-training) — with Transformer architectures and whole model fine-tuning, hence facilitating a fair comparison with the BERT model. We show that the resulting ELMo can achieve competitive performance with BERT in terms of overall GLUE performance; to our best knowledge, no such ELMo baseline was explored in prior work. We additionally demonstrate that a simple ensemble of left-to-right and right-to-left models, which are pre-trained and fine-tuned completely independently, can approximate the performance of BERT on the full data setup, and even *outperform* it on the efficient learning scenario. Second, above and beyond conducting principled comparisons between BERT and its baselines, we outline several recommendations that would encourage and incentivize this line of work; we hope that these recommendations would be adopted by the broader community, with the aim of accelerating progress towards resolving our current scientific debt in language model pre-training research.

D Validation Set Results with Standard Deviation

In Figure 1, we show the performance of our best validation set hyper-parameters for each task across three different random seeds, based on which we derive the error bars. We remark that the majority of the tasks have fairly small error bars, providing evidence for the robustness and generality of our observations across different random seeds.

E Dataset and Artifacts License

The Wikipedia dataset is available under the Creative Common license, cc-by-sa-3.0, which we use solely for research purposes in accordance with its terms. Our entire codebase is based on Huggingface's open-source Transformer implementation, which is released under the Apache-2.0 license.

[13] A similar line of prior work (Dong et al., 2019) also proposed combining multiple pre-training objectives using the same Transformer model through a Cloze-type formulation.

Model	Training Data	Text Processing	Pre-Training Objective	Architecture	Sequence Length	Batch Size	Steps/Epochs
BERT Original	English Wikipedia Book Corpus	Wordpiece	MLM, NSP	Transformers	512	256	1M steps
GPT-1 Original	Word Benchmark Book Corpus	BPE	CLM	Transformers	512	64	100 epochs
ELMO Original	Word Benchmark	Character level + convolution	CLM	LSTM	2048	N.A.	10 epochs
BERT Rerun	English Wikipedia Book Corpus	Wordpiece	MLM	Transformers	512	256	1M steps
GPT-1 Rerun	English Wikipedia Book Corpus	Wordpiece	CLM	Transformers	512	256	1M steps
ELMO-variant Rerun	English Wikipedia Book Corpus	Wordpiece	CLM	Transformers	512	256	1M steps

Table 4: Comparison of the pre-training hyper-parameters across different models. MLM denotes masked language modelling; NSP denotes next-sentence prediction; CLM denotes causal language modelling, respectively.

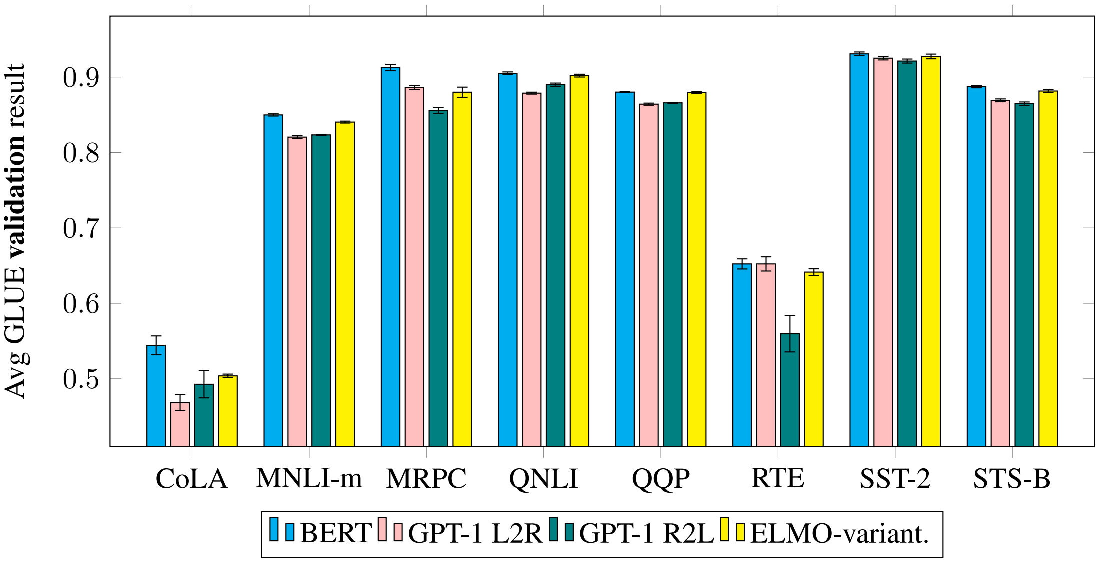

Figure 1: GLUE validation set results. For each task, we show the mean performance — for the best validation set fine-tuning hyper-parameters as outlined in Appendix B.1 — alongside the standard deviation of the results (denoted with the error bars), which is computed based on three different random seeds for the exact same winning hyper-parameter. Note that the reported score for each task follows the exact same metric as the test result in Table 1. The BERT, GPT-1, and ELMO-variant results reported here are based on our reruns with proper controls and comparable conditions between different models, and not from the original reported results of each work.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
7

☑ A2. Did you discuss any potential risks of your work?
Section "Ethical Considerations" (after Section 7)

☑ A3. Do the abstract and introduction summarize the paper's main claims?
"Abstract", 1

☒ A4. Have you used AI writing assistants when working on this paper?
We don't use any AI writing assistants

B ☑ Did you use or create scientific artifacts?

3.1

☑ B1. Did you cite the creators of artifacts you used?
3.1

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
3.1, Appendix E

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Appendix E

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
We use the same exact data as the prior work of BERT (Devlin et al, 2019)

☒ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
We use the same exact data as the prior work of BERT (Devlin et al, 2019)

☒ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
We use the same exact data with the prior work of BERT (Devlin et al, 2019)

C ☑ Did you run computational experiments?

3

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
3.1, Appendix B

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
3.1, Appendix B

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Appendix D

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
3.1

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.