Benchmarking Large Language Model Capabilities for Conditional Generation

Joshua Maynez
Google DeepMind
joshuahm@google.com

Priyanka Agrawal
Google DeepMind
priyankagr@google.com

Sebastian Gehrmann
Google Research
gehrmann@google.com

Abstract

Pre-trained large language models (PLMs) underlie most new developments in natural language processing. They have shifted the field from application-specific model pipelines to a single model that is adapted to a wide range of tasks. Autoregressive PLMs like GPT-3 or PaLM, alongside techniques like few-shot learning, have additionally shifted the output modality to generation instead of classification or regression. Despite their ubiquitous use, the generation quality of language models is rarely evaluated when these models are introduced. Additionally, it is unclear how existing generation tasks—-while they can be used to compare systems at a high level—-relate to the real world use cases for which people have been adopting them. In this work, we discuss how to adapt existing application-specific generation benchmarks to PLMs and provide an in-depth, empirical study of the limitations and capabilities of PLMs in natural language generation tasks along dimensions such as scale, architecture, input and output language. Our results show that PLMs differ in their applicability to different data regimes and their generalization to multiple languages and inform which PLMs to use for a given generation task setup. We share best practices to be taken into consideration when benchmarking generation capabilities during the development of upcoming PLMs.

1 Introduction

Natural language generation tasks require generating understandable text given textual or non-linguistic information as input, such as documents, tables, or other structured forms. These texts seek to achieve a communicative goal (e.g., summarize a document). The standard approach to tackle these problems over the last years has been to start with a pretrained encoder-decoder model like T5 (Raffel et al., 2020a) or BART (Lewis et al., 2020a) and finetune it on a corpus that captures the downstream task. The recent much larger pretrained language models use a decoder-only architecture and upended this paradigm. These models enabled few-shot or in-context learning approaches in which a model is presented with one or more examples and tasked to continue generating without any finetuning. We refer to both kinds of pretrained models as PLMs.

Due to the lack of grounding in the specific task setup, few-shot learning in generation settings leads to a model approaching the communicative goal from very different angles. These diverse range of outputs make the typical reference-based automatic evaluation strategies largely incompatible. While human evaluation can be used to overcome this shortcoming, it is infeasible to monitor the performance of an actively training model this way or to re-run all evaluations every time a new model is introduced. This leads to the question how one should reliably monitor generation capabilities, a question that is only growing in importance as more tasks are approached by casting them into generation setups.

In this work, we evaluate 8 models in few-shot and finetuning settings on 27 generation tasks covering 14 languages via automatic evaluation, presenting the first large-scale benchmark of PLMs in conditional NLG settings. We discuss design choices and challenges to ensure a fair comparison between the different systems, including suitable methods, tasks, and metrics. Based on our empirical results, we derive recommendations that could be used for future benchmarks during the development of PLMs. To combat the need for repeating computationally expensive explorations, we investigate how many evaluation examples are necessary to identify differences between models and find that, in many cases, fewer than 500 examples are sufficient, which opens the path for future evaluation-only task developments.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 9194–9213

2 Background and Related Work

The shift from specialized pipelines toward pretrained language models has led to significant changes in how models are evaluated. We now focus more on questions such as "*how good are the learned representations?*" instead of user-facing measures of utility. The changes manifested in leaderboards and standard benchmarks that aim to characterize a wide range of model capabilities (Ethayarajh and Jurafsky, 2020).

An additional recent shift is that from finetuning toward few-shot learning. Models like T5 (Raffel et al., 2020a), BART (Lewis et al., 2020a), and mT5 (Xue et al., 2021) were finetuned on supervised datasets covering tasks including translation and summarization, and their outputs are compared to "ground truth" outputs via widely used metrics like ROUGE (Lin, 2004) which provide a noisy indication of the "quality" of the output and which can be used to determine whether a model is better than others.[1] In contrast, large PLMs with autoregressive language modeling pretraining objectives are more capable to produce results without explicit finetuning and are thus typically evaluated via few-shot and in-context approaches, where the model is given the task description and exemplars showing how the task should be completed. GPT-3 (Brown et al., 2020) and models that followed such as GLaM (Du et al., 2022), Gopher (Rae et al., 2021), and LaMDA (Thoppilan et al., 2022), have achieved few-shot state-of-the-art results on a large number of tasks at their time of publication. However, few-shot approaches work best for tasks with a clear answer such as classification or span-based question-answering.[2]

Generation metrics penalize systems when their writing style differs from how the references are written (Mathur et al., 2020; Freitag et al., 2020; Mille et al., 2021). Without finetuning, there is no guarantee that PLMs produce outputs that look like the ground truth, both in style and content. Recent work found that these differences leads to sharp differences in how humans and automatic metrics rate the generation quality (Goyal et al., 2022). Due to this uncertainty, most evaluations of new PLMs are limited to NLU benchmarks such as SuperGLUE (Wang et al., 2019). For example, LaMDA (Thoppilan et al., 2022) did not evaluate on NLG tasks, GLaM (Du et al., 2022) limited its generation evaluation to short span question answering tasks, and GPT-3 (Brown et al., 2020) evaluated only on machine translation. A first autoregressive PLM with broad NLG evaluation, PaLM (Chowdhery et al., 2022), benchmarked summarization and data-to-text tasks in multiple languages.

The recent Holistic Evaluation of Language Models project (HELM, Liang et al., 2022) aims to standardize evaluation of language models. With the explicit goal to broaden the task and metric coverage, HELM has established an impressive few-shot benchmark for many natural language tasks. Corroborating the prior findings, they also conclude that human evaluation is necessary for NLG. This distinction means that the reference-based approach for generated text that the field has used since the advent of deep learning may no longer sufficient and that we need clear evaluation protocols that continue to allow us to answer broad questions about "generation quality" of a model. Complementing this work, we take a deeper look at a wider set of NLG tasks and explore LLMs in finetuning and few-shot setups to identify whether reference-based automatic evaluation can still be used to produce system rankings.

Research Questions We aim to define a methodology that allows us to answer the question "How good are learned representations of a model for generating natural language?" via few-shot and finetuning approaches. To develop and apply this methodology we seek to answer the following three research questions:

R1 *How do different model architectures compare in terms of automatic metrics?*
We aim to identify patterns that emerge in evaluations and to uncover aspects inherent to the tasks, e.g. *have metrics on specific tasks saturated?*, and to the models' architectural choices, e.g., are encoder-decoders better suited for particular task formulations? (Section 4)

R2 *What set of tasks, methods, and metrics is best suited for the monitoring of improvements in language generation capabilities?*
Using the results of R1, we aim to select a subset of tasks, methods, and metrics that robustly produce reliable model rankings. (Section 5)

[1]For an in-depth review of the usefulness of automatic metrics, we refer to Gehrmann et al. (2022b) and point to Section 4 for a discussion of the application of metrics to benchmarks.

[2]We refer to the two task types as NLU and NLG tasks but note that this distinction becomes fuzzy with autoregressive models since technically all answers are "generated".

Dataset	Languages	Length		Size	
		Input	Output	Training	Test
E2E	en	146	135	35k	4.7k
WebNLG	en,ru	169.5	157	14k–35k	1.1k-1.8k
ToTTo	en	357		120k	7.7k
Czech Rest.	cs	70	80	3.5k	842
XSum	en	1845	153	23k	1.2k
WikiLingua	en,es,ru,tr,vi	1k–5k	159–489	5k–3.8M	900-29k
MLSum	es,de	4152	147	220k–250k	10k-13k
XL-Sum	ar,bn,ja,id,sw, ko,ru,te,th,tr, es,vi,hi	1k–10k	137–614	1.3k–300k	500-9k

Table 1: Details of the datasets evaluated in this paper: languages, lengths in number of tokens according to the mT5 tokenizer (Xue et al., 2021), and size of the training and test splits.

R3 *What are the broader implications for how the quality of newly developed models should be monitored?*
Robustly ranking systems is particularly important when monitoring a system during training and when comparing across many tasks. In line with the "reality check" theme track at ACL 2023, we discuss the implications of our findings on how evaluation results should be produced and interpreted. (Section 6)

3 Method

3.1 Data

We select a combination of data-to-text and text-to-text datasets as different input modalities. The selected datasets capture different input and output lengths, domains, languages, and communicative goals. The text-to-text task with most available multilingual datasets is summarization which we pick for this paper.[3] We pick the following tasks:[4]

- **MLSum** (Scialom et al., 2020) – Summarize a news article in multiple sentences.
- **WikiLingua** (Ladhak et al., 2020) – Generate section headers for step-by-step instructions from WikiHow.
- **XSum** (Narayan et al., 2018) – Generate the first sentence of a news article.
- **Clean E2E NLG** (Novikova et al., 2017; Dušek et al., 2019) – Given a set of key-value attribute pairs, describe a restaurant in one or two sentences.
- **Czech Restaurant response generation** (Dusek and Jurvc'ivcek, 2019) – Given a dialog context and a dialog act representation, generate a one sentence long response.
- **WebNLG 2020** (Gardent et al., 2017; Ferreira et al., 2020) – Verbalize subject-predicate-object triples in one or more sentences.
- **ToTTo** (Parikh et al., 2020) – Describe highlighted cells in a table in a single sentence.
- **XL-Sum** (Hasan et al., 2021) – Summarize a news article, in the same language, in a single sentence.

Table 1 provides an overview of these datasets in terms of languages, the lengths of input and output and split sizes. For highly multilingual datasets, we evaluate on a subset of typologically diverse languages following the selection by Clark et al. (2020). To this selection, we add languages that appear bothin WikiLingua and XL-Sum.

3.2 Models

Prior results for the benchmarked tasks primarily come from finetuning T5 (Raffel et al., 2020b), mT5 (Xue et al., 2021), or BART (Lewis et al., 2020b), which are encoder-decoder models pretrained with an infilling objectives. These models are significantly smaller than newer models like GPT-3, with sizes ranging from 130M to 13B parameters. Encoder-decoder models trained for infilling often outperform larger decoder-only LMs in the finetuning setting (Tay et al., 2022), while the latter work better for few-shot setting. There has also been recent work on reducing the computational cost of large models by ∼10x by using a mixture of experts (Zoph et al., 2022). It is important to compare these diverse set of models to understand how scale plays a role with the model's architecture and its pretraining. We benchmark the following models:[5]

- **PaLM** PaLM is a pretrained decoder-only transformer-based model trained with standard left-to-right language modeling objective. It is pretrained on a range of multilingual corpora including Wikipedia, news, and code. In this work, we use two models scales: 8B parameters and 540B parameters.

[3] Since benchmarks for machine translation are well-established (e.g., Akhbardeh et al., 2021) we exclude it from our scope. However, any methodological outcomes of our work can be applied to translation or similar tasks.

[4] All datasets were retrieved via the Generation Evaluation and Metrics benchmark (Gehrmann et al., 2021, 2022a). We use these datasets for research purposes only in line with their intended use.

[5] Model names omitted for anonymity.

- **GPT-3.5** (Ouyang et al., 2022b) GPT-3.5 is a 175B parameter decoder-only transformer-model of the GPT-3 family (Brown et al., 2020) but trained on a blend of text and code from before Q4 2021. This model, named code-davinci-002, was introduced as the base model for InstructGPT-3 (Ouyang et al., 2022b) without the supervision on human-written demonstrations and human-vetted model samples.[6]
- **ST-MoE** (Zoph et al., 2022) ST-MoE is a 269B sparse pretrained variant of a dense encoder-decoder transformer-based model.
- **LaMDA** (Thoppilan et al., 2022) LaMDA (137B parameters) is a decoder-only transformer-based language model specialized for dialog applications. It is pretrained on dialog data as well as web text data followed by rank-based tuning.
- **T5** (Raffel et al., 2020a) T5-XXL (11B parameters) is a pretrained encoder-decoder transformer-based model trained on a span corruption objective with a novel unified text-to-text format. It is pretrained on Common Crawl data, mostly containing English-only documents.
- **mT5** (Xue et al., 2021) mT5-XXL (11B parameters) is a multilingual variant of T5 that was pretrained on a multilingual corpus, mC4, covering 101 languages.
- **LongT5** (Guo et al., 2021) LongT5 (3B parameters) a similar architecture as T5, where the encoder is extended to have global-local attention sparsity patterns to handle long inputs.

3.3 Few-shot evaluation methodology

To evaluate the models for few-shot inference, we concatenate a task-specific prompt[7] to the input and prepend an output prompt to the output. To handle the oftentimes very long inputs or outputs for tasks such as summarization, inputs were truncated to 2048 tokens and inference was done providing only one exemplar at a time, referred to as 1-shot. These simple prompts are analogous to those used in related work (Chowdhery et al., 2022; Scao et al., 2022). We do not tune the prompts or use more complex strategies to keep fair comparisons between multiple systems, as prompt selection can lead to overfitting. The exemplars are separated through double linebreaks, which are also used to truncate output predictions for evaluation. All few-shot exemplars are randomly sampled from the training corpus. From early experimentation, we found this particularly important since it avoids overfitting to exemplars that work well for one model but not another.

3.4 Finetuning methodology

To use the decoder-only architectures during finetuning, inputs and targets are concatenated. The concatenated sequences are truncated to 2048 tokens, the training context used during pretraining, with 512 tokens reserved for the target. Only summarization tasks required input truncation. We finetuned models with standard hyperparameters; refer to Appendix-B for thorough details. The best model checkpoint for each dataset was selected by the best performing geometric mean of ROUGE-1, ROUGE-2 and ROUGE-L scores on the validation set. Decoding was done with beam-search with a beam size of 4 for encoder-decoder models, while inference in decoder-only PLMs (LaMDA, PaLM, ST-MoE) was performed using top-k sampling with k=10, due to issues with scaling beam search at the time of publication.

3.5 Metrics

Following the suggestions by Gehrmann et al. (2022b), we report a combination of lexical and learned metrics, starting with ROUGE-2 and ROUGE-L (Lin, 2004). Since the default ROUGE implementation uses English-specific tokenization, stemming and punctuation normalization, it is incompatible with other languages. Hasan et al. (2021) extended ROUGE by integrating additional stemmers and tokenizers to cover up to the 45 languages. To support more languages, and avoid dependency on varying implementations, we use a SentencePiece tokenizer (Kudo and Richardson, 2018) which, provided a vocabulary distribution file, is self-contained and has sensible fall-backs to unexpected words. Specifically, we used mT5's SentencePiece vocabulary.

For the same reason, we also evaluate with ChrF (Popović, 2015), which is a character-level n-gram overlap metrics and thus independent from tokenizers. BLEURT (Sellam et al., 2020; Pu et al., 2021) is a multilingual model-based evaluation metric for generation designed to compute the similarity between a pair of sentences i.e. a reference and a candidate. It finetunes RemBERT (Chung

[6] More details can be found at https://beta.openai.com/docs/model-index-for-researchers

[7] For Summarization, this prompt was *"Summarize the following article:"*, and for Data-to-Text it was *"Verbalize:"*. This was translated to the appropriate language.

et al., 2021) on synthetic sentence pairs and gold ratings. In contrast to the lexical metrics, BLEURT is meant to capture the non-trivial semantic similarities between two texts.

For brevity, the main text of this section focuses on the F-measure of ROUGE-L for English and SentencePiece-ROUGE-L for all other languages while the remaining results are in Appendix A. We additionally investigate the agreement between metrics in Section 5.[8]

4 Empirical Observations

Few-shot learning falls behind finetuning For many generation tasks, including multilingual summarization tasks, we observe a large gap between finetuning and few-shot results, indicating that finetuning will play an important role when it comes to maximizing automatic scores. On data-to-text, the few-shot results follow a similar trend as in summarization, but the gap to the best finetuned results shrinks drastically. Moreover, the finetuning result do not always follow a trend according to scale or architecture. We hypothesize that multiple tasks have saturated to the metrics. If this is the case, approaching them as few-shot generation tasks may still yield insights but it is no longer productive to use them to benchmark finetuned models.

Finetuned decoder-only PLMs can match encoder-decoder performance with scale In summarization, finetuned decoder-only PLMs, such as PaLM-540B, closely match or exceeds the best reported prior results on all English generation tasks. This demonstrates that PLMs can make up their architectural disadvantage through its vastly increased scale. While finetuning PLMs is computationally expensive, it serves as an important upper bound for few-shot predictions.

Multilingual generation capabilities are highly dependent on pretraining data The PLMs evaluated are mostly pretrained on English corpora: 99+% for T5, LongT5, ST-MoE; 90% for PaLM, LaMDA; contrarily mT5 is explicitly pretrained in a multilingual corpus.[9] PaLM achieves best results in 3 out of 4 English generation tasks which generate English text, even when the input is non-English. However, the much smaller mT5 bests the other models in 10 out of 14 non-English summarization tasks, and the relative difference between few-shot and finetuning is larger for non-English generation. This suggests that English-centric PLMs are better at processing non-English input than generating non-English output.

Analyzing the effects of input context length Tasks with long inputs suffer from models' limitation to process said inputs. Inputs are thus usually transformed (e.g. cropped, re-ranked, etc) to fit into the model. We found that a several of the evaluated tasks, such as WikiLingua and MLSum benefit from a longer input context in models even if the long-context model is smaller (i.e., LongT5 vs T5). In contrast, the performance is comparable for the rest of short-context tasks.

```
Given a natural language generation task:

Step 1:   Is few-shot learning (A) or finetuning (B)
best suited?
  i)    Inconsistent system rankings might indicate
        tasks are saturated. Few-shot learning will be
        more sensible to monitor in those cases.
  ii)   Encoder-decoder baselines are strong
        performers regardless of smaller scale.

Step 2-A: How best to benchmark few-shot learning?
  i)    Avoid curating prompts to individual PLMs.
  ii)   Randomly select exemplars for prompts.
  iii)  Control for output length outliers.

Step 3:   What is an efficient test set size?
  i)    Pick the smallest size that produces
        consistent system rankings.
  ii)   Evaluate more datasets rather than larger
        evaluations, adhering to (3.i).

Step 4:   What metrics are best suited for the task?
  i)    Effective automatic metrics should provide
        consistent system rankings.
  ii)   Overlap-based metrics are not calibrated for
        few-shot learning.
  iii)  Human evaluation is vital if comparing
        few-shot vs finetuning.
```

Figure 1: General recommendations when monitoring or benchmarking PLMs.

5 Deriving Evaluation Practices

Figure 1 summarizes the recommendations we developed from challenges we faced and our observed empirical results. These recommendations are best understood in the context of monitoring

[8]For ROUGE, we used the python implementation at https://github.com/google-research/google-research/tree/master/rouge at commit f935042 and whitespace-tokenized references and predictions before calling the library. For BLEURT, we used BLEURT-20 checkpoint from the library at https://github.com/google-research/bleurt and commit c6f2375.

[9]The language breakdown for GPT-3.5 is unknown.

Task	One-shot				Finetuning					
	PaLM 8B	PaLM 540B	LaMDA 137B	GPT-3.5 175B	PaLM 8B	PaLM 540B	ST-MoE 32B	T5 11B	mT5 11B	LongT5 3B
Data-To-Text										
E2E (en)	37.7	**46.6**	7.1	**46.6**	52.9	52.3	51.5	52.9	52.2	**53.1**
WebNLG (en)	45.3	**54.7**	8.4	54.6	56.8	**58.0**	56.4	50.8	47.7	**58.0**
ToTTo (en)	40.2	**50.7**	5.6	**51.9**	65.8	**67.5**	67.0	66.1	65.5	66.3
Czech Restaurant (cs)	16.9	34.1	3.3	**38.5**	**45.5**	**45.5**	40.7	45.4	39.4	44.8
WebNLG (ru)	16.8	**33.7**	4.5	33.3	40.9	40.5	28.2	41.2	41.1	**41.6**
English Generation										
XSum (en)	19.9	28.6	10.0	**34.0**	31.4	36.5	**38.3**	36.5	33.2	36.0
XLSum (en)	16.8	22.7	8.4	**27.9**	34.6	44.3	**45.4**	43.1	41.8	42.6
WikiLingua (en)	6.5	6.4	5.9	**7.7**	**8.0**	7.5	7.8	7.9	7.9	7.8
Crosslingual Generation										
WikiLingua (es → en)	6.5	6.1	5.0	**7.7**	7.7	7.6	7.3	7.8	7.6	**7.9**
WikiLingua (ru → en)	10.2	17.5	0.7	**18.9**	29.9	**35.7**	25.1	27.9	31.7	30.8
WikiLingua (tr → en)	10.1	20.0	7.7	**21.2**	31.1	**38.8**	31.5	26.8	36.7	28.2
WikiLingua (vi → en)	7.7	14.5	2.2	**16.2**	28.9	**32.9**	22.9	22.7	31.0	28.5
Multilingual Generation [SentencePiece-ROUGE-2]										
MLSum (es)	12.8	**14.3**	5.2	13.0	23.0	24.5	25.0	24.3	**25.7**	25.6
MLSum (de)	13.6	21.3	3.9	**22.6**	35.2	41.4	**44.1**	43.5	43.3	43.7
XLSum (ar)	12.2	**19.0**	10.8	18.0	36.2	39.9	15.7	15.2	**42.3**	6.2
XLSum (bn)	5.8	6.9	6.1	**11.7**	26.4	31.1	11.1	10.2	**36.5**	11.0
XLSum (ja)	11.3	15.1	5.4	**18.3**	38.7	42.5	4.5	4.5	**43.7**	4.6
XLSum (id)	16.8	**20.4**	9.0	20.1	35.5	**43.5**	41.1	41.6	**43.5**	40.8
XLSum (sw)	16.7	**24.5**	11.5	15.4	32.7	36.4	37.0	37.4	**40.7**	36.3
XLSum (ko)	16.1	**18.2**	7.9	17.6	33.8	37.3	20.3	19.5	**45.0**	19.9
XLSum (ru)	12.6	16.1	10.8	**19.1**	30.3	38.3	18.1	17.8	**38.6**	17.7
XLSum (te)	6.5	7.7	6.2	**13.1**	20.5	30.0	15.1	15.1	**33.5**	14.8
XLSum (th)	6.7	8.6	5.2	**13.3**	23.4	29.5	13.5	13.7	**34.3**	13.1
XLSum (tr)	15.2	**17.7**	8.0	16.8	33.3	**42.4**	30.3	30.4	42.3	29.7
XLSum (es)	15.7	**17.4**	8.3	16.9	25.2	**34.3**	31.9	32.5	33.9	32.3
XLSum (vi)	13.2	14.9	6.9	**15.4**	25.9	**41.5**	27.7	27.3	41.0	26.7
XLSum (hi)	10.0	12.1	9.3	**15.2**	37.7	**43.6**	13.7	2.3	43.5	2.3

Table 2: ROUGE-L and SentencePiece-ROUGE-L results on data-to-text and compression datasets. Best results in **bold**. Few-shot results lag behind finetuned results and the gap increases as tasks become more complex. The non-English performance mostly follows the trend that higher percentages of non-English pretraining data leads to better performance. Despite their much smaller size, encoder-decoder model frequently much larger decoder-only models after finetuning.

and benchmarking PLMs during training or inference.

Comparable few-shot learning evaluation As mentioned in Section 3, our design choices were made to ensure that results are comparable across PLMs. Primarily, prompts were deliberately kept extremely simple and all few-shot exemplars were randomly sampled. While highly curated prompts or methods like chain-of-thought prompting can increase the performance considerably (Wei et al., 2022b), it can also lead to overfitting to the particular model the prompt was developed on, in turn making a comparison to other models unfair and producing unrealistic expectations when people have single interactions with it.

Overlap-based metrics are not calibrated to evaluate few-shot learning Few-shot generation suffers from not being able to predict output length properly given the few exemplars provided. While encoder-decoder models utilize end-of-string tokens, these are not always learned during decoder-only pretraining. To circumvent this issue, researchers rely on PLMs match to the few-shot format provided e.g. line-breaks that separate exemplars. We observed PLMs fail to follow the format a significant number of times, producing the largest allowed length on occasion. In our experiments, we tried to avoid very long outputs by trimming outputs to the 95-percentile length seen in the targets.[10] Still, few-shot output lengths

[10]This simple method avoids discrepancies across PLMs which might have different maximum decoding lengths.

are on average 2-3 times the average target length while finetuned model's output average 80% the average target length, across all tasks. Overlap metrics used in generation are sensitive to length (Sun et al., 2019) making a natural disadvantage for few-shot learners. We do not recommend using overlap-based metrics to compare few-shot results without length normalization.

Computational costs can be decreased without sacrificing relative model performance The computational cost of evaluating large datasets, some with more than 10K examples, are prohibitive and perhaps unnecessary. To that end, we investigate if a model ranking can be produced, with a high degree of certainty, while only considering a random subset of the test set, saving compute cost to possibly evaluate on more tasks instead. To investigate this effect, we ran the following experiment: (1) Sample n datapoints from a dataset and all corresponding model scores. (2) Following Kocmi et al. (2021) and Graham et al. (2014), we perform Wilcoxon Rank Sum test (Wilcoxon, 1946) to assess the stability of the ranking. (3) Repeat steps 1&2 k times and record the fraction of runs in which models scores from any two models were not distinguishable from each other (those with a p-value of > 0.05). Since we are considering 10 model settings in this work, this experiment considers all 45 possible pairs.

The result shown in Figure 2 provides insight into the required number of data points to produce rankings. For most datasets, we can produce stable model rankings with only 500 examples, some with as little as 100. Tasks where models achieve very similar scores tend to require more test examples, since smaller score differences require more examples to be distinguishable from each other (Wei and Jia, 2021).[11]

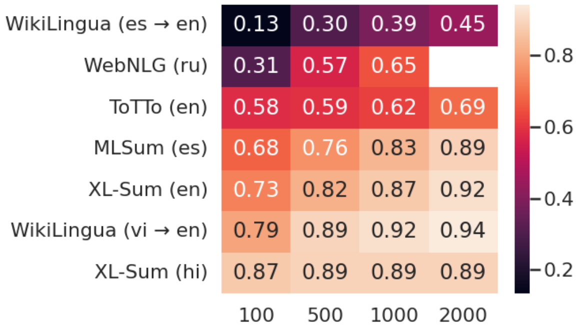

Figure 2: Empirical probability of p-value of Wilcoxon Rank Sum test < 0.05 for any combination between 1-shot and finetuned models.

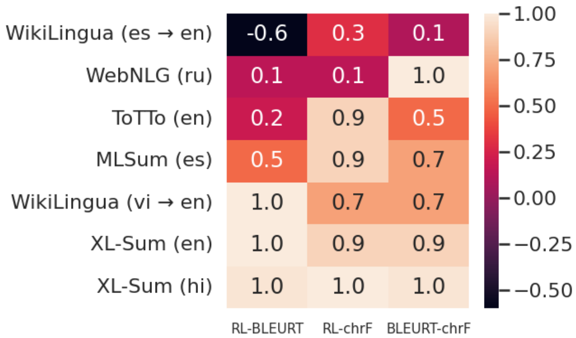

Figure 3: Spearman correlation coefficients between metrics: (SP)ROUGE-L, BLEURT and ChrF.

Analyzing metrics utility We use different automated metrics to evaluate the generation quality of the models. These metrics attempt to capture the similarity between system generated output and the reference text. While ROUGE and chrF account for the lexical overlap, BLEURT is meant to compute the semantic similarity. It is important to understand the agreement between these metrics. We compute the the system-level agreement via Spearman correlation coefficient (Spearman, 1987) between the scores given by the metrics to the fine-tuned set of models. Figure 3 shows the correlation between ROUGE-L (RL), BLEURT and ChrF. We observe that the metrics are highly correlated for most datasets. Similar to Figure 2, on the tasks where the models have similar performance, we notice less correlation among the metrics. Such tasks are may have either saturated performance, e.g., ToTTo (en) or all models perform poorly, e.g., Wikilingua (es-> en). Due to the small differences between models, metrics fail to produce the same rankings.

6 Discussion and Reality Check

In line with our goal to provide a "reality check" via empirical and theoretical research, and to reflect on the ways in which reported performance improvements are meaningful, we want to situate our findings in the context of the broader NLP community. Openly accessible APIs and publicly available large models have led to increased attention on large pretrained models, but they have also led to a "release-then-test" philosophy where models are released without extensive evaluations. While the findings we present in this paper do not

[11]Full results available in Appendix A.

solve this issue, agreeing on a shared evaluation process could lead to more realistic claims about model performance (and shortcomings), and allow for a more accurate monitoring of models during training.

What claims can we not make? Empirical findings demonstrate that incorporating generation into NLU tasks via Chain-of-Thought leads to better model performance (Wei et al., 2022b; Suzgun et al., 2022). Providing additional grounding via finetuning on instructions and aligning a model to human feedback leads to better task-specific performance without supervision (Wei et al., 2022a; Ouyang et al., 2022a). However, we lack the scientific methods to quantify these advances. While benchmarks provide an indication whether a model is performing better than a previous iteration, and projects like BIG-bench (Srivastava et al., 2022) and HELM (Liang et al., 2022) enable evaluation on a very wide range of possible tasks, they are also inherently limited.

When benchmarking models in few-shot settings, especially models for which little information about their training data is available, it is hard to disambiguate model performance from memorization, i.e. if the examples were seen during pretraining. Instruction tuning further blur the line between finetuning and few-shot, which can lead to very different outputs and are not fully comparable. It is thus near impossible to make claims about *why* a model is succeeding at one particular task without having access to its training data.

As mentioned earlier, the target of this work is to derive best practices for comparing models in generation settings with constrained computational budgets, for example when monitoring a training model or when trying to compare on many different tasks. Our findings are grounded in much prior work that finds that metrics have a very high agreement with human judgments on the system-level (e.g., Kocmi et al., 2021), but are essentially meaningless on the segment-level. For that reason, we cannot derive claims beyond these rankings about utility of a model or whether a particular model would actually produce useful outputs for a task. To derive such insights, we point to work on extrinsic evaluation which requires comprehensive human evaluations (e.g., Lee et al., 2022).

How can our findings be applied to improve the status quo? Since the generation capabilities of PLMs are currently not extensively monitored or evaluated, we set out to derive best practices for how these evaluations can look. We found that many of the "easy" tasks, on which finetuned models saturate the metrics, still yield insights for few-shot approaches. We further identified the tension between doing a computationally expensive full evaluation on a dataset and adding more evaluation sets for different tasks or languages. Our findings suggest that evaluation on small subsets of more tasks can be beneficial to the overall results.

To further motivate this suggestion, consider the following thought experiment: We have two tasks, A and B. At 500 examples, they have a risk of producing a "wrong" ranking of 10%. At 1,000 examples, they have a risk of producing a wrong ranking of 5%. These risks are not correlated, i.e., their covariance is 0. Given a computational budget of evaluating on 1,000 examples, the risk of only evaluating on one dataset is 5%, and the risk of producing two wrong ratings after evaluating on A and B is only 1%. While additional datasets introduce a larger risk of one individual dataset producing misleading results (18% in this case), one can easily expand this argument to a whole portfolio of tasks to hedge against individual dataset risk (Stuart and Markowitz, 1959). Many existing NLU benchmarks like BIG bench (Srivastava et al., 2022) already follow such a strategy and we believe that generation evaluation, especially considering the additional risk due to metrics, should follow this approach for the use cases discussed in this work. To further minimize the individual dataset risk, they can be switched out once they saturate or their sample sizes increased.

7 Conclusion

In this work, we produced an extensive evaluation of a diverse set of state-of-the-art pre-trained language models (PLMs) for 27 different multilingual generation tasks under few-shot learning and finetuning settings. We discuss empirical results that help inform practitioners which tasks, methods and metrics are suitable. We provide recommendations on how best to monitor conditional generation capabilities of PLMs, including how to fairly benchmark few-shot learning, automated metrics and their utility, and how to efficiently utilize computational resources. We hope that such findings and recommendations could positively influence natural language evaluation in future work.

8 Limitations

In this work, we have presented results that help inform us what tasks, methods and metrics are best suited for monitoring as well as methodologies and empirical information about the current set of models. We provide detailed information of how these results can be reproduced, to the extend that research have access to the PLMs in question, but these results have limitations, in order to reduce costs, many languages were not evaluated which might have left unforeseen patterns not discussed in this work. Moreover, few-shot learning, in particular, could exhibit large variance if different prompts were chosen, or a different set of exemplars chosen. Because of the high costs involved our work does not explore the performance difference when multiple sets of hyper-parameters were chosen.

On the conceptual level, we make the assumption that system-level improvements on our tasks translate to downstream usefulness. While prior work suggests that this is the case, tools like chatGPT have significantly expanded the possible application space beyond the realm of "typical" NLP tasks, and we don't know how well our findings generalize to this space of tasks.

9 Ethics Statement

This paper focuses on conditional generation tasks where models are free to generate long text sequences. Typical issues associated with text generation such as hallucinations, memorization of private information publicly available, toxic and discriminatory language, or sensitive generated content could and are likely to arise. measuring the extent to which these issues occur is a necessary and crucial additional dimension of model evaluation which we do not include in this work, which should be seen as supplemental.

References

Farhad Akhbardeh, Arkady Arkhangorodsky, Magdalena Biesialska, Ondřej Bojar, Rajen Chatterjee, Vishrav Chaudhary, Marta R. Costa-jussa, Cristina España-Bonet, Angela Fan, Christian Federmann, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Leonie Harter, Kenneth Heafield, Christopher Homan, Matthias Huck, Kwabena Amponsah-Kaakyire, Jungo Kasai, Daniel Khashabi, Kevin Knight, Tom Kocmi, Philipp Koehn, Nicholas Lourie, Christof Monz, Makoto Morishita, Masaaki Nagata, Ajay Nagesh, Toshiaki Nakazawa, Matteo Negri, Santanu Pal, Allahsera Auguste Tapo, Marco Turchi, Valentin Vydrin, and Marcos Zampieri. 2021. Findings of the 2021 conference on machine translation (WMT21). In *Proceedings of the Sixth Conference on Machine Translation*, pages 1–88, Online. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, T. J. Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher 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. *ArXiv*, abs/2005.14165.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek B Rao, Parker Barnes, Yi Tay, Noam M. Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Benton C. Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier García, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Díaz, Orhan Firat, Michele Catasta, Jason Wei, Kathleen S. Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways. *ArXiv*, abs/2204.02311.

Hyung Won Chung, Thibault Fevry, Henry Tsai, Melvin Johnson, and Sebastian Ruder. 2021. Rethinking embedding coupling in pre-trained language models. In *International Conference on Learning Representations*.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

Nan Du, Yanping Huang, Andrew M. Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, Barret Zoph, Liam Fedus, Maarten Bosma, Zongwei Zhou, Tao Wang, Yu Emma Wang, Kellie Webster, Marie Pellat, Kevin Robinson, Kathleen S. Meier-Hellstern, Toju Duke, Lucas Dixon, Kun Zhang, Quoc V. Le, Yonghui Wu, Z. Chen, and Claire Cui. 2022. Glam: Efficient scaling of language models with mixture-of-experts. In *International Conference on Machine Learning*.

Ondrej Dusek and Filip Jurvc'ivcek. 2019. Neural generation for czech: Data and baselines.

Ondřej Dušek, David M Howcroft, and Verena Rieser. 2019. Semantic Noise Matters for Neural Natural Language Generation. In *Proceedings of the 12th International Conference on Natural Language Generation (INLG 2019)*, pages 421–426, Tokyo, Japan.

Kawin Ethayarajh and Dan Jurafsky. 2020. Utility is in the eye of the user: A critique of NLP leaderboards. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4846–4853, Online. Association for Computational Linguistics.

Thiago Castro Ferreira, Claire Gardent, Nikolai Ilinykh, Chris van Der Lee, Simon Mille, Diego Moussallem, and Anastasia Shimorina. 2020. The 2020 Bilingual, Bi-Directional WebNLG+ Shared Task Overview and Evaluation Results (WebNLG+ 2020). In *Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+)*, Dublin/Virtual, Ireland.

Markus Freitag, George Foster, David Grangier, and Colin Cherry. 2020. Human-paraphrased references improve neural machine translation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1183–1192, Online. Association for Computational Linguistics.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. Creating training corpora for nlg micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 179–188. Association for Computational Linguistics.

Sebastian Gehrmann, Tosin Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Aremu Anuoluwapo, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna Clinciu, Dipanjan Das, Kaustubh D. Dhole, Wanyu Du, Esin Durmus, Ondřej Dušek, Chris Emezue, Varun Gangal, Cristina Garbacea, Tatsunori Hashimoto, Yufang Hou, Yacine Jernite, Harsh Jhamtani, Yangfeng Ji, Shailza Jolly, Mihir Kale, Dhruv Kumar, Faisal Ladhak, Aman Madaan, Mounica Maddela, Khyati Mahajan, Saad Mahamood, Bodhisattwa Prasad Majumder, Pedro Henrique Martins, Angelina McMillan-Major, Simon Mille, Emiel van Miltenburg, Moin Nadeem, Shashi Narayan, Vitaly Nikolaev, Rubungo Andre Niyongabo, Salomey Osei, Ankur Parikh, Laura Perez-Beltrachini, Niranjan Ramesh Rao, Vikas Raunak, Juan Diego Rodriguez, Sashank Santhanam, João Sedoc, Thibault Sellam, Samira Shaikh, Anastasia Shimorina, Marco Antonio Sobrevilla Cabezudo, Hendrik Strobelt, Nishant Subramani, Wei Xu, Diyi Yang, Akhila Yerukola, and Jiawei Zhou. 2021. The gem benchmark: Natural language generation, its evaluation and metrics.

Sebastian Gehrmann, Abhik Bhattacharjee, Abinaya Mahendiran, Alex Wang, Alexandros Papangelis, Aman Madaan, Angelina McMillan-Major, Anna Shvets, Ashish Upadhyay, Bingsheng Yao, et al. 2022a. Gemv2: Multilingual nlg benchmarking in a single line of code. *arXiv preprint arXiv:2206.11249*.

Sebastian Gehrmann, Elizabeth Clark, and Thibault Sellam. 2022b. Repairing the cracked foundation: A survey of obstacles in evaluation practices for generated text. *ArXiv*, abs/2202.06935.

Tanya Goyal, Junyi Jessy Li, and Greg Durrett. 2022. News summarization and evaluation in the era of GPT-3. *CoRR*, abs/2209.12356.

Yvette Graham, Timothy Baldwin, Alistair Moffat, and Justin Zobel. 2014. Is machine translation getting better over time? In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 443–451, Gothenburg, Sweden. Association for Computational Linguistics.

Mandy Guo, Joshua Ainslie, David C. Uthus, Santiago Ontañón, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2021. LongT5: Efficient text-to-text transformer for long sequences. *CoRR*, abs/2112.07916.

Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. 2021. XL-sum: Large-scale multilingual abstractive summarization for 44 languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4693–4703, Online. Association for Computational Linguistics.

Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494, Online. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Faisal Ladhak, Esin Durmus, Claire Cardie, and Kathleen McKeown. 2020. WikiLingua: A new benchmark dataset for cross-lingual abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4034–4048, Online. Association for Computational Linguistics.

Mina Lee, Megha Srivastava, Amelia Hardy, John Thickstun, Esin Durmus, Ashwin Paranjape, Ines Gerard-Ursin, Xiang Lisa Li, Faisal Ladhak, Frieda Rong, Rose E. Wang, Minae Kwon, Joon Sung Park, Hancheng Cao, Tony Lee, Rishi Bommasani, Michael S. Bernstein, and Percy Liang. 2022. Evaluating human-language model interaction. *CoRR*, abs/2212.09746.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Annual Meeting of the Association for Computational Linguistics*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020b. 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.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D. Manning, Christopher R'e, Diana Acosta-Navas, Drew A. Hudson, E. Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue Wang, Keshav Santhanam, Laurel J. Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan S. Kim, Neel Guha, Niladri S. Chatterji, O. Khattab, Peter Henderson, Qian Huang, Ryan Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas F. Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. 2022. Holistic evaluation of language models. *ArXiv*, abs/2211.09110.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Nitika Mathur, Johnny Wei, Markus Freitag, Qingsong Ma, and Ondřej Bojar. 2020. Results of the WMT20 metrics shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 688–725, Online. Association for Computational Linguistics.

Simon Mille, Kaustubh D. Dhole, Saad Mahamood, Laura Perez-Beltrachini, Varun Gangal, Mihir Sanjay Kale, Emiel van Miltenburg, and Sebastian Gehrmann. 2021. Automatic construction of evaluation suites for natural language generation datasets. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 201–206, Saarbrücken, Germany. Association for Computational Linguistics.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022a. Training language models to follow instructions with human feedback. *CoRR*, abs/2203.02155.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. 2022b. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155.

Ankur P. Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. Totto: A controlled table-to-text generation dataset. *ArXiv*, abs/2004.14373.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Amy Pu, Hyung Won Chung, Ankur Parikh, Sebastian Gehrmann, and Thibault Sellam. 2021. Learning compact metrics for MT. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 751–762, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John F. J. Mellor, Irina Higgins, Antonia Creswell, Nathan McAleese, Amy Wu, Erich Elsen, Siddhant M. Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, L. Sifre, Lena Martens, Xiang Lorraine

Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, N. K. Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Tobias Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d'Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew G. Johnson, Blake A. Hechtman, Laura Weidinger, Iason Gabriel, William S. Isaac, Edward Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem W. Ayoub, Jeff Stanway, L. L. Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. 2021. Scaling language models: Methods, analysis & insights from training gopher. *ArXiv*, abs/2112.11446.

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

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

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*.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. 2020. MLSUM: the multilingual summarization corpus. *CoRR*, abs/2004.14900.

Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. Bleurt: Learning robust metrics for text generation. In *Annual Meeting of the Association for Computational Linguistics*.

C. Spearman. 1987. The proof and measurement of association between two things. by c. spearman, 1904. *The American journal of psychology*, 100 3-4:441–71.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ameet Rahane, Anantharaman S. Iyer, Anders Andreassen, Andrea Santilli, Andreas Stuhlmüller, Andrew M. Dai, Andrew La, Andrew K. Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakas, and et al. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *CoRR*, abs/2206.04615.

Alan L. Stuart and Harry M. Markowitz. 1959. Portfolio selection: Efficient diversification of investments. *A Quarterly Journal of Operations Research*, 10:253.

Simeng Sun, Ori Shapira, Ido Dagan, and Ani Nenkova. 2019. How to compare summarizers without target length? pitfalls, solutions and re-examination of the neural summarization literature. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 21–29, Minneapolis, Minnesota. Association for Computational Linguistics.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. 2022. Challenging big-bench tasks and whether chain-of-thought can solve them. *CoRR*, abs/2210.09261.

Yi Tay, Mostafa Dehghani, Samira Abnar, Hyung Won Chung, William Fedus, Jinfeng Rao, Sharan Narang, Vinh Q Tran, Dani Yogatama, and Donald Metzler. 2022. Scaling laws vs model architectures: How does inductive bias influence scaling? *arXiv preprint arXiv:2207.10551*.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam M. Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, Yaguang Li, Hongrae Lee, Huaixiu Zheng, Amin Ghafouri, Marcelo Menegali, Yanping Huang, Maxim Krikun, Dmitry Lepikhin, James Qin, Dehao Chen, Yuanzhong Xu, Zhifeng Chen, Adam Roberts, Maarten Bosma, Yanqi Zhou, Chung-Ching Chang, I. A. Krivokon, Willard James Rusch, Marc Pickett, Kathleen S. Meier-Hellstern, Meredith Ringel Morris, Tulsee Doshi, Renelito Delos Santos, Toju Duke, Johnny Hartz Søraker, Ben Zevenbergen, Vinodkumar Prabhakaran, Mark Díaz, Ben Hutchinson, Kristen Olson, Alejandra Molina, Erin Hoffman-John, Josh Lee, Lora Aroyo, Ravindran Rajakumar, Alena Butryna, Matthew Lamm, V. O. Kuzmina, Joseph Fenton, Aaron Cohen, Rachel Bernstein, Ray Kurzweil, Blaise Aguera-Arcas, Claire Cui, Marian Croak, Ed Chi, and Quoc Le. 2022. Lamda: Language models for dialog applications. *ArXiv*, abs/2201.08239.

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*, abs/1905.00537.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022a. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations,* *ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Huai hsin Chi, Quoc Le, and Denny Zhou. 2022b. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903.

Johnny Wei and Robin Jia. 2021. The statistical advantage of automatic NLG metrics at the system level. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6840–6854, Online. Association for Computational Linguistics.

Frank Wilcoxon. 1946. Individual comparisons of grouped data by ranking methods. *Journal of economic entomology*, 39(2):269–270.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Barret Zoph, Irwan Bello, Sameer Kumar, Nan Du, Yanping Huang, Jeff Dean, Noam Shazeer, and William Fedus. 2022. St-moe: Designing stable and transferable sparse expert models.

A Additional empirical results

Table 3, Table 4 and Table 5 report ROUGE-2 and BLEURT and ChrF results respectively for all tasks. These results are in line with the discussed results in 4

B Technical details

Finetuning and inference was done in the **t5x** framework for public and closed access models. Few-shot learning task methodology is well described in 3, for public access models inference was done via their respective public API, whilst all other models were loaded from the standard checkpoint into TPU accelerators and inference was done on batches of 64. Finetuning was carried out in TPU accelerators, for PaLM we used a constant learning rate of 5×10^{-5}, 20x smaller than during pretraining and reset the optimizer's (Adafactor) accumulators, for T5, mT5 and LongT5 we used a constant learning rate of 1×10^{-4}.

C Computational Cost and Environmental Impact

In our work we benchmark 27 generation tasks which require a substantial amount of computational resources. Inference on PLMs is exceedingly more efficient than finetuning. We report the number of test examples across all tasks to be 194,844. Inference over this number of examples times 10 models evaluated amounts to 2 million inference examples. Finetuning on the other hand, requires all parameters to be trained and training dataset sizes are considerably larger. We estimate the compute usage for finetuning to be 256 TPU v3 chips for 200 hours. One of the goals of this work is to encourage benchmarking in the future to avoid these costs by more efficiently selecting smaller test size and persuade researchers to only evaluate finetuning approaches when suitable.

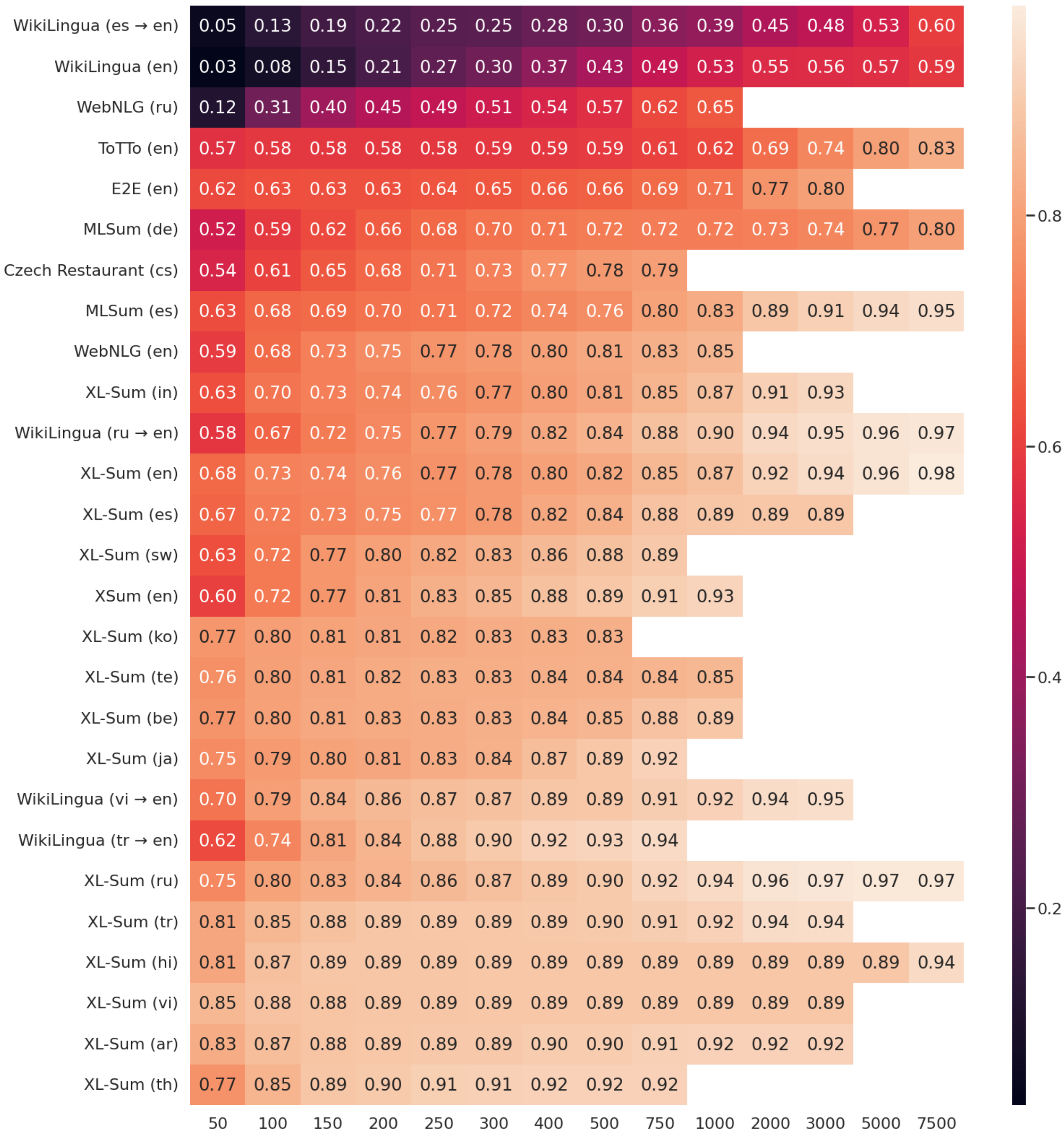

Figure 4: Empirical probability of p-value of Wilcoxon Rank Sum test < 0.05 for any combination between 1-shot and finetuned models.

Task	One shot				Finetuning					
	PaLM 8B	PaLM 540B	LaMDA 137B	GPT-3.5 175B	PaLM 8B	PaLM 540B	ST-MoE 32B	T5 11B	mT5 11B	LongT5 3B
Data-To-Text										
E2E (en)	26.7	37.3	4.2	37.9	45.7	45.3	44.2	45.2	45.5	46.3
WebNLG (en)	33.8	45.8	5.4	46.0	47.7	49.2	47.6	39.6	35.8	48.8
ToTTo (en)	26.4	37.8	3.2	38.1	53.9	55.9	55.2	54.1	53.3	54.5
Czech Restaurant (cs)	7.9	18.1	0.9	22.3	30.2	30.6	25.4	28.8	25.0	29.9
WebNLG (ru)	4.9	16.5	1.2	16.8	22.4	23.4	13.0	23.1	23.2	24.2
English Generation										
XSum (en)	8.0	14.4	3.4	19.9	16.3	21.2	22.8	21.0	17.5	20.8
XLSum (en)	6.5	11.7	2.7	17.0	19.6	29.5	30.6	28.0	26.5	27.4
WikiLingua (en)	0.7	0.4	0.7	0.9	0.4	0.4	0.4	0.4	0.4	0.4
Crosslingual Generation										
WikiLingua (es → en)	0.7	0.5	0.4	0.6	0.4	0.4	0.3	0.4	0.4	0.4
WikiLingua (ru → en)	3.1	6.8	0.1	7.8	14.0	18.7	12.0	12.7	15.1	14.4
WikiLingua (tr → en)	3.1	8.7	2.1	10.1	16.6	23.0	17.7	13.8	20.8	14.3
WikiLingua (vi → en)	2.4	5.5	0.4	6.8	13.4	0.4	10.3	9.7	14.8	13.2
Multilingual Generation [SentencePiece-ROUGE-2]										
MLSum (es)	3.7	4.5	0.7	4.9	10.7	0.7	13.1	12.1	13.5	13.6
MLSum (de)	8.8	16.8	1.2	16.9	26.9	33.4	36.5	36.1	35.9	36.3
XLSum (ar)	4.5	11.7	2.4	9.6	25.8	30.0	1.9	2.0	32.1	0.6
XLSum (bn)	1.0	1.8	0.5	2.7	18.5	23.5	0.2	0.1	29.4	0.1
XLSum (ja)	4.0	6.7	0.3	9.6	27.1	31.8	0.1	0.1	31.5	0.1
XLSum (id)	7.2	11.3	3.7	12.6	25.0	33.0	30.5	30.7	32.7	30.3
XLSum (sw)	7.9	16.2	6.5	6.6	22.7	26.8	27.8	27.6	31.3	26.8
XLSum (ko)	6.9	9.6	1.6	9.4	23.0	26.7	4.1	3.7	34.9	4.0
XLSum (ru)	6.0	9.2	3.7	10.9	20.8	29.4	4.5	6.1	29.6	6.1
XLSum (te)	2.4	3.3	1.1	4.8	13.3	22.7	3.2	3.2	26.5	3.3
XLSum (th)	2.9	4.0	0.3	6.2	16.4	22.5	2.4	2.5	26.9	2.4
XLSum (tr)	7.5	10.5	3.2	10.7	23.7	32.7	17.6	17.8	32.2	17.7
XLSum (es)	5.8	9.0	3.1	9.6	14.2	23.7	20.1	20.6	23.0	20.6
XLSum (vi)	4.8	6.8	1.5	7.5	20.2	35.9	11.9	13.2	35.5	13.1
XLSum (hi)	4.4	6.4	1.8	7.0	29.0	35.7	1.8	0.0	35.4	0.0

Table 3: ROUGE-2 and SentencePiece-ROUGE-2 results in data-to-text, English and multilingual generation datasets.

Task	One shot				Finetuning					
	PaLM 8B	PaLM 540B	LaMDA 137B	GPT-3.5 175B	PaLM 8B	PaLM 540B	ST-MoE 32B	T5 11B	mT5 11B	LongT5 3B
Data-To-Text										
E2E (en)	46.1	57.8	15.1	57.8	62.5	61.9	61.1	62.1	60.9	61.8
WebNLG (en)	47.5	61.8	17.0	61.8	63.6	65.2	64.1	59.4	55.8	65.4
ToTTo (en)	43.5	55.8	12.6	55.2	67.5	69.4	68.3	67.3	66.8	67.7
Czech Restaurant (cs)	17.6	35.6	7.9	41.5	48.0	48.1	36.6	38.2	44.1	40.1
WebNLG (ru)	21.8	45.5	15.3	45.3	62.7	62.6	24.4	31.8	63.5	32.1
English Generation										
XSum (en)	24.6	32.7	18.5	37.6	34.4	38.9	41.3	39.3	36.8	39.0
XLSum (en)	23.8	29.9	13.9	33.5	35.8	45.8	47.0	46.0	43.9	44.4
WikiLingua (en)	14.7	14.3	15.4	15.8	15.1	14.6	15.2	15.8	15.8	15.3
Crosslingual Generation										
WikiLingua (es $\rightarrow$ en)	16.8	13.2	10.2	14.6	14.2	14.9	13.0	15.1	14.8	15.7
WikiLingua (ru $\rightarrow$ en)	19.1	21.5	1.5	22.5	30.6	35.9	24.0	28.8	34.3	33.3
WikiLingua (tr $\rightarrow$ en)	19.3	24.6	12.3	26.7	34.4	39.4	32.6	30.8	39.0	32.7
WikiLingua (vi $\rightarrow$ en)	16.4	19.9	4.4	21.3	31.8	14.2	23.2	27.6	32.3	32.5
Multilingual Generation										
MLSum (es)	21.3	22.9	5.3	20.6	28.0	18.4	29.1	28.7	30.7	30.3
MLSum (de)	28.9	37.0	5.7	34.4	41.9	48.8	50.9	50.5	50.1	51.5
XLSum (ar)	14.2	22.7	11.4	24.4	35.0	39.6	0.2	0.2	41.6	0.1
XLSum (bn)	10.3	12.7	4.5	17.5	28.6	34.0	0.0	0.0	37.8	0.0
XLSum (ja)	8.8	11.6	1.2	13.8	26.0	31.3	0.8	0.9	30.6	0.9
XLSum (id)	21.0	26.0	9.0	26.2	36.8	45.3	43.2	42.3	43.0	43.4
XLSum (sw)	24.0	33.0	15.0	21.5	36.2	42.0	40.1	41.1	44.4	41.6
XLSum (ko)	6.9	9.4	1.6	10.0	18.0	21.8	1.4	1.2	27.5	1.4
XLSum (ru)	15.0	19.8	9.4	26.5	29.1	38.6	14.4	20.1	40.3	19.9
XLSum (te)	11.3	13.6	4.7	16.8	18.0	29.8	0.3	0.2	30.4	0.3
XLSum (th)	14.7	16.8	4.4	21.5	27.1	33.4	0.3	0.3	33.9	0.3
XLSum (tr)	20.3	24.9	6.2	24.5	32.7	43.1	31.2	33.1	42.6	33.8
XLSum (es)	19.0	22.9	7.3	22.0	24.5	33.4	31.5	31.9	32.6	32.8
XLSum (vi)	10.9	13.1	2.4	14.2	21.8	37.1	16.9	20.2	34.3	21.1
XLSum (hi)	12.2	15.1	6.6	18.8	33.2	39.6	0.2	0.0	39.1	0.0

Table 4: ChrF results in data-to-text, English and multilingual generation datasets.

Task	One shot				Finetuning					
	PaLM 8B	PaLM 540B	LaMDA 137B	GPT-3.5 175B	PaLM 8B	PaLM 540B	ST-MoE 32B	T5 11B	mT5 11B	LongT5 3B
Data-To-Text										
E2E (en)	60.0	71.8	44.2	72.3	76.5	75.8	75.1	76.4	75.9	76.2
WebNLG (en)	62.3	74.5	43.5	74.7	75.8	76.8	75.6	71.2	67.8	76.3
ToTTo (en)	56.9	69.4	33.1	69.5	76.8	77.9	77.0	76.6	76.8	76.7
Czech Restaurant (cs)	34.7	66.8	32.2	72.2	75.8	74.4	48.8	51.8	72.9	48.8
WebNLG (ru)	39.2	67.8	19.7	66.9	77.5	78.0	25.9	29.8	78.4	29.9
English Generation										
XSum (en)	43.0	46.9	28.2	53.4	51.0	55.5	58.5	56.4	53.2	56.4
XLSum (en)	32.7	41.1	22.0	51.1	52.6	61.9	63.0	61.8	60.3	60.8
WikiLingua (en)	33.3	34.0	27.9	34.3	32.2	32.2	31.3	32.4	32.6	32.1
Crosslingual Generation										
WikiLingua (es → en)	32.9	33.7	16.8	33.4	32.3	32.6	31.0	32.5	33.0	32.5
WikiLingua (ru → en)	38.8	43.3	6.4	45.9	50.6	54.4	46.4	49.1	52.3	51.5
WikiLingua (tr → en)	39.3	44.0	19.3	46.4	49.2	54.4	48.6	45.4	52.8	46.8
WikiLingua (vi → en)	35.6	38.2	5.7	40.8	50.6	32.8	45.4	45.5	51.4	50.0
Multilingual Generation										
MLSum (es)	21.0	21.5	-1.3	26.7	30.6	6.3	25.9	24.2	32.0	27.0
MLSum (de)	39.4	50.1	4.6	49.3	57.8	63.4	62.5	61.6	61.5	61.4
XLSum (ar)	19.8	28.5	2.5	27.7	44.6	50.1	2.8	3.5	53.0	4.3
XLSum (bn)	31.8	41.8	0.2	27.4	46.6	57.1	2.9	3.6	62.7	2.8
XLSum (ja)	28.1	31.4	-1.2	34.7	47.0	52.2	-0.3	-0.3	53.0	-0.3
XLSum (id)	41.2	47.4	9.5	53.8	58.7	68.0	61.4	65.5	66.8	65.6
XLSum (sw)	25.5	36.3	14.3	24.0	45.8	52.9	48.6	50.2	59.1	48.9
XLSum (ko)	25.6	31.6	-0.3	33.0	40.5	47.1	0.8	1.6	54.6	1.4
XLSum (ru)	30.1	37.3	3.2	33.0	47.9	59.6	14.2	16.7	58.0	17.0
XLSum (te)	29.6	35.0	6.5	22.7	32.0	49.1	10.9	11.5	51.6	12.4
XLSum (th)	22.0	27.2	-0.1	16.3	31.9	43.6	-1.1	-0.9	46.0	-1.1
XLSum (tr)	30.8	34.5	3.3	40.0	49.8	63.8	21.4	26.4	62.5	26.3
XLSum (es)	21.2	26.3	0.0	30.6	31.5	46.2	33.3	36.1	45.2	35.7
XLSum (vi)	14.5	14.5	-1.6	16.4	24.7	46.5	-4.0	-4.6	45.0	-4.5
XLSum (hi)	33.9	40.4	7.0	33.7	50.7	57.5	5.7	4.6	57.3	4.6

Table 5: BLEURT results in data-to-text, English and multilingual generation datasets.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
We added a section on Limitations.

☑ A2. Did you discuss any potential risks of your work?
We discuss in the ethics section the risks of autoregressive text generation, in particular how it can produce misinformation or sensitive data.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
The introduction summarizes the paper's main claims.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

We used artifacts throughout

☑ B1. Did you cite the creators of artifacts you used?
We used generation datasets found through the GEM benchmark (gem-benchmark.com). We cite the researchers that released these data in section 3.1.

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Licenses: MLSum (Scialom et al., 2020) – MIT License https://github.com/ThomasScialom/MLSUM/blob/master/LICE
WikiLingua(Ladhak et al., 2020) – cc-by-3.0 https://gem-benchmark.com/data_cards/wiki_lingua
XSum(Narayan et al., 2018) – cc-by-sa-4.0 https://gem-benchmark.com/data_cards/xsum
*Clean E2E NLG(Novikova et al., 2017; Duseket al., 2019) – cc-by-sa-4.0 https://gem-benchmark.com/data_cards/e2e_*
Czech Restaurants (Dusek and Jurvcivcek, 2019) – cc-by-sa-4.0https://gem-benchmark.com/data_cards/cs_restaurants
*WebNLG 2020(Gardent et al., 2017; Ferreiraet al., 2020) – cc-by-nc-4.0 https://gem-benchmark.com/data_cards/web_*
ToTTo(Parikh et al., 2020) – cc-by-sa-3.0 https://gem-benchmark.com/data_cards/totto
XL-Sum(Hasan et al., 2021) – cc-by-nc-sa-4.0 https://gem-benchmark.com/data_cards/xlsum

☑ 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)?
All the data mentioned above has intended research purposes which is consistent with this work's use. We mention this in the paper

☒ 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?
No data was collected. The data has been revised outside this work by community efforts such as GEM.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
We provided information on the languages covered by each dataset in Table 1, as well as the publicly available information on model's pretraining data language distribution.

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

☑ 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 provided information on the statistic of the datasets used in Table 1,

C ☑ Did you run computational experiments?

Section 4 and 5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Number of parameters per model is reported in section 3.2. Computing infrastructure is mentioned in the appendix B. The total computational budget is discussed in its own section.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Hyper parameter selection is discussed in appendix B. Experimental setup is discussed in the methodology section.

☑ 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?
Most of the contributions for this paper are statistics of empirical results and method to obtain them.

☑ 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.)?
We provide this information in the main text, in footnotes.

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.