LongForm: Effective Instruction Tuning with Reverse Instructions

Abdullatif Köksal[*◇†], **Timo Schick**[*], **Anna Korhonen**[†], **Hinrich Schütze**[*◇]

[*]Center for Information and Language Processing, LMU Munich
[◇]Munich Center for Machine Learning
[†]Language Technology Lab, University of Cambridge
akoksal@cis.lmu.de

Abstract

Instruction tuning enables language models to more effectively generalize and better follow user intent. However, obtaining instruction data is costly and challenging. Prior work employs methods such as expensive human annotation, crowd-sourced datasets with alignment issues, and generating noisy examples via LLMs. We introduce the LongForm-C dataset, which is created by *reverse instructions*. We generate instructions via LLMs for human-written corpus examples using reverse instructions. First we select a diverse set of human-written documents from corpora such as C4 and Wikipedia; then we generate instructions for these documents via LLMs. This approach provides a cheaper and cleaner instruction-tuning dataset with natural output and one suitable for long text generation. Our models outperform 10x larger language models without instruction tuning on tasks such as story/recipe generation and long-form question answering. Moreover, LongForm models outperform prior instruction-tuned models such as FLAN-T5 and Alpaca by a large margin, and improve language understanding capabilities further. We publicly release our data and models: https://github.com/akoksal/LongForm.

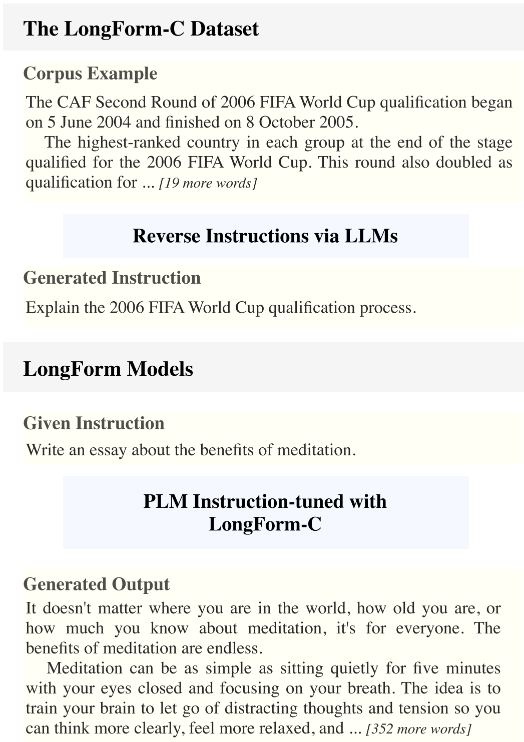

Figure 1: Our **Reverse Instructions** method extracts high-quality human-written passages from corpora ("Corpus Example") and then leverages an LLM to generate, for each corpus example, a *generated instruction* that should elicit this corpus example. Using our method, we create the LongForm-C instruction tuning dataset, consisting of *(generated instruction, corpus example)* pairs. We then instruction-tune a PLM with LongForm-C. The PLM output shown ("Generated Output") indicates that this approach produces high-quality output. Evaluated on a diverse set of text generation tasks, Reverse Instructions outperforms prior models by more than 35%.

1 Introduction

Instruction tuning conditions language models (LMs) on user intents and improves cross-task generalization while ensuring better human alignment. Recent works on instruction tuning focus on data collection, showing that training with even a small amount of instruction-tuning data can outperform much larger LMs in following user intent and generalization (Ouyang et al., 2022; Chung et al., 2024). However, these studies have some limitations, such as reliance on expensive human-annotated instruction data (Ouyang et al., 2022) or focusing on instructions for academic NLP tasks that have limited coverage of real-world generation tasks (Chung et al., 2024; Wang et al., 2022; Sanh et al., 2022). A third approach, works that generate instructions and outputs from scratch using large language models (LLMs), often produces low-quality datasets (Honovich et al., 2023; Wang et al., 2022).

Using the **reverse instructions** method, we create an instruction-following text generation dataset

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 7056–7078

called **LongForm-C** to address these issues. We aim to improve instruction tuning by gathering diverse human-written texts from C4 and English Wikipedia as outputs. As illustrated in Figure 1, we extract text passages (paragraphs or documents) from the corpora and then prompt an LLM with a zero-shot template to generate, for each passage, an instruction that – when given to an LLM – we want to result in the generation of the passage or a similar text. We refer to this method of creating *(generated instruction, passage)* pairs for instruction tuning as **reverse instructions**. This method offers a cost-effective and fast alternative for creating instruction tuning datasets compared to human annotation while also yielding higher quality output compared to fully synthetic data generation.

To make **LongForm-C** even more effective, we increase its diversity and quality by leveraging structured examples from Stack Exchange and WikiHow, and long text generation tasks from NLP benchmarks.

Based on the LongForm-C dataset, we finetune instruction-following PLMs, which we call **LongForm** models, with different architectures and sizes: T5-XL (Raffel et al., 2020), OPT-6.7B (Zhang et al., 2022), and LLaMA-7B (Touvron et al., 2023a). We compare these models with the baselines FLAN-T5 (Chung et al., 2024), T0pp (Sanh et al., 2022), T*k*-Instruct (Wang et al., 2022), and Alpaca on a diverse set of tasks: story, poem, email and recipe generation; grammar error correction; text summarization; table-to-text; and long form question answering. Our experimental results are as follows. (i) LongForm-OPT-2.7B outperforms OPT-30B on long text generation despite OPT-30B having 10x more parameters. (ii) LongForm models outperform prior instruction-following models, such as FLAN-T5 and Alpaca, with more than **63%** (in-domain) and **35%** (out-of-domain) relative METEOR improvement on text generation. (iii) In addition to NLG, the LongForm-C dataset also enhances the language understanding (NLU) capabilities of several LLMs. It improves performance on MMLU (Hendrycks et al., 2021) compared to prior datasets like FLAN.

(iv) For multilingual news generation, LongForm models follow multilingual instructions and generate news articles in German, Spanish, French and Russian better than prior models.

We release the LongForm-C dataset and models publicly on `https://github.com/akoksal/LongForm`.

2 Related Work

Instruction-following models: The instruction paradigm was introduced to better control language models using natural language commands (Ouyang et al., 2022; Wang et al., 2022; Chen et al., 2022; Wei et al., 2022). While some approaches (Ouyang et al., 2022) focus on obtaining costly human-annotation data through their platform and improving human alignment with reinforcement learning, recent works (Srivastava et al., 2023; Wang et al., 2022) have extended this paradigm to various existing NLP tasks by reformulating them. This approach helps to overcome the instruction-data bottleneck, but is limited to academic tasks. Additionally, this approach is still restricted in terms of text generation: only 7 out of 1762 tasks in BigBench (Srivastava et al., 2023) and 12 out of 1613 in Super-Natural Instructions (NIv2) (Wang et al., 2022) include English tasks with long outputs (i.e., average number of words >50). Self-Instruct (Wang et al., 2023b) and Unnatural Instructions (Honovich et al., 2023) challenge academic benchmarks by generating instruction data (input, instruction and output) via LLMs to provide a more diverse dataset. However, relying solely on generation via LLMs has drawbacks: the overall validity of Self-Instruct is only 54% (Wang et al., 2023b). More recent works focus on curating costly human-annotated datasets such as Dolly (Conover et al., 2023).

Data Generation with LMs: Recent neural network based approaches to NLP rely on large and diverse training datasets. Thus, many works focus on augmenting or generating new training examples. For example, existing datasets are expanded to enhance model quality for question answering (QA) (Longpre et al., 2019), part-of-speech tagging (Şahin and Steedman, 2018), and textual similarity (Schick and Schütze, 2021). For a more general-purpose training, Self-Instruct (Wang et al., 2023b) and Unnatural Instructions (Honovich et al., 2023) propose the use of LLMs to generate tasks and examples including instructions and outputs. WizardLM (Xu et al., 2024) focuses on evolving human-written instructions by prompting LLMs and generating more complicated instructions that require multiple-step reasoning. To the best of our knowledge, LongForm-C is the first work to combine corpora and LLMs to automatically generate a general-purpose text generation dataset.

Corpus Mining: Most NLP tasks benefit from extracting useful examples from corpora such as machine translation (MT) with bitext mining (Resnik, 1999) and argument mining. Recent approaches use embeddings to find similar texts in different languages to mine human translation from unlabeled data (Artetxe and Schwenk, 2019). These methods have been extended to MT with up to 200 languages (NLLB Team et al., 2022). Additionally, some tasks filter sentences from corpora to create diverse training sets for human annotation, such as argument mining (Ein-Dor et al., 2020). Recent works explore methods to mine and restructure corpora (e.g., Wikipedia) to generate synthetic data for conversational QA (Dai et al., 2022) and closed-book QA (Lewis et al., 2021).

Long Text Generation: Long text generation is challenging. It requires models to understand long dependencies and planning to generate long texts. Many works do task-specific generation (e.g., generation of stories (Fan et al., 2018), recipes (Bień et al., 2020), essays (Feng et al., 2018) and long-form QA (Fan et al., 2019)) by collecting useful examples from existing resources (e.g., ELI5 subreddit, WritingPrompts subreddit, recipe websites). Also, proper evaluation of long text generation remains a significant challenge for task-specific or general-purpose models (Celikyilmaz et al., 2020).

Reverse Instructions: The reverse instructions method offers a cost-effective and faster alternative for creating instruction tuning datasets compared to human annotation, while also yielding higher quality output compared to fully-synthetic data generation. Building upon this approach, Wang et al. (2023c) and Li et al. (2023) expand the concept of reverse instructions by integrating instruction data filtering mechanisms and open-source models. Furthermore, Chen et al. (2024) introduces a rewriting step of corpus samples in reverse instructions. In a significant extension to multilingual settings, Köksal et al. (2024) proposed MURI (Multilingual Reverse Instructions), which leverages multilingual corpora and translation models to generate instruction-following datasets for 200 languages. Moreover, Ziegler et al. (2024) introduced CRAFT, which extends the reverse instructions methodology to task-specific contexts through the generation of structured datasets via corpus example retrieval and LLM-based reformulation.

The reverse instructions methodology constitutes an efficacious and cost-efficient alternative to conventional human annotation for instruction tuning dataset creation, while demonstrating superior quality compared to fully-synthetic data generation approaches. This framework has been subsequently extended through several significant contributions in the literature. Specifically, Wang et al. (2023c) and Li et al. (2023) augmented the fundamental reverse instructions paradigm through the integration of instruction data filtering mechanisms and the utilization of open-source models. Chen et al. (2024) further advanced this methodology by introducing a corpus sample rewriting procedure within the reverse instructions framework. Köksal et al. (2024) expanded it to multilingual settings via MURI (Multilingual Reverse Instructions), which leverages multilingual corpora with translation models and LLMs to generate instruction-following datasets for 200 languages. Furthermore, Ziegler et al. (2024) developed CRAFT, adapting the reverse instructions approach to generate task-specific structured datasets through corpus example retrieval and reformulation with LLMs.

3 The LongForm-C Dataset

The LongForm-C dataset consists of 15,000 corpus examples generated via reverse instructions and additional 12,739 examples from structured corpora (SC) and NLP datasets with instructions and long text pairs. The main part of the LongForm-C dataset is constructed by reverse instructions generation for a diverse set of corpus samples; this part is referred to as **RI** below. Additionally, we expand our dataset with structured corpora examples through parsing and templates (referred to as **SC**), as well as NLP tasks reformulated to increase diversity (referred to as **NLP**). Thus, our dataset features a diverse collection of instructions (including human-written, template-based and LLM-generated), each paired with a human-written output.

3.1 Reverse Instructions (RI)

Figure 2 summarizes the reverse instructions method.

1. **Data Selection**: We sample 10,000 examples from the C4 corpus and 5,000 examples from the English Wikipedia for selecting target text in English. Since C4 is noisy (Dodge et al., 2021), we choose only those texts whose URLs had received three or more upvotes in Reddit, following Radford

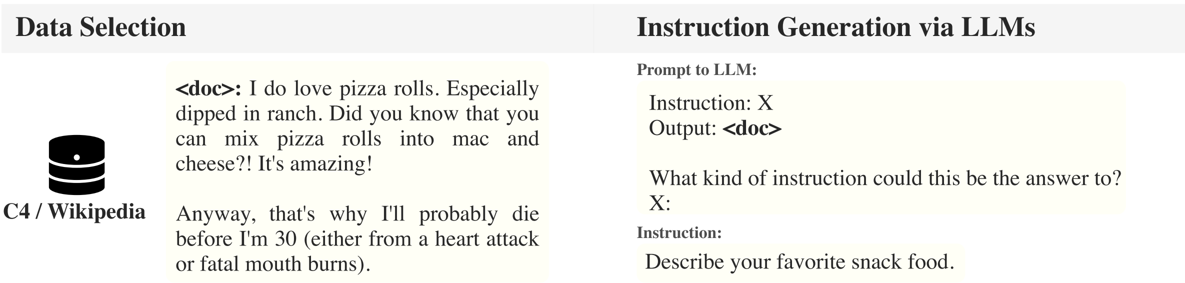

Figure 2: The reverse instructions method. After collecting diverse examples from corpora, we generate relevant instructions through zero-shot prompting via LLMs in various styles. (see §F.1)

et al. (2019). However, this filtering may result in a lack of diversity, as most URLs may originate from specific subreddits that are more active, such as the news subreddit. To address this, we k-means cluster all documents in C4 based on their BERT embeddings. We then select 10,000 examples, one example from each cluster that is closest to the cluster center. This helps prevent corpus examples from being dominated by multiple samples from a specific domain or cluster, thus ensuring a greater diversity of texts.

For the English Wikipedia, we adopt a direct approach since it is already diverse. We randomly select 5,000 articles and extract the first paragraph for 75% and the first two paragraphs for 25% of the examples. The resulting dataset contains shorter texts: average length is 57±43 words (compared to 408±263 for C4). The overall average in LongForm-C is 291±272.

2. **Instruction Generation**: We aim to generate relevant instructions for the 15K corpus examples using LLMs. We employ GPT3 (text-davinci-003 from OpenAI API) as our LLM. We design prompts for generating instructions for a given document and query the LLM in a zero-shot manner. We create three templates to diversify the styles of our instructions: formal instruction style (50%), informal chatbot style (30%), and search engine query style (20%). The formal instruction template is structured as follows:

```
Instruction: X
Output: “<corpus_example>”
What kind of instruction could this be the answer to?
X:
```

We add “X:” at the end of the template as a prompt to generate plausible instructions. See §F.1 for details of LLM parameters and templates.

Finally, we incorporate length information into the generated instruction using a predefined set of templates to signal the desired length of the output and provide additional control to the model. As the length of our corpus examples varies (291±272 words), we provide templates such as “Respond in D sentences” or “Respond in D words”. We also include less precise templates such as “Respond briefly” (resp. “Respond in detail”) for outputs of less than 3 (resp. more than 10) sentences. We append or prepend these templates to the original instructions for a 30% subset of our dataset. This lets us control LongForm models in terms of length.

3.2 Structured Corpora (SC)

We also mine structured corpora that contain (instruction,(long-)output) pairs: Stack Exchange and WikiHow. We do not use LLMs here.

From the **Stack Exchange (SE)** subcorpus of Pile (Gao et al., 2020), we select 50 examples from each of 88 subdomains, covering a wide range of topics, including Buddhism, chemistry, and webmasters. For each example, we select the question and its corresponding details as an instruction and the answer as the output. This subset adds more complicated human instructions to LongForm-C.

WikiHow includes how-to tutorials, each comprising a question and an answer that consists of two parts: an introduction and several steps with a brief summary and a long description of each step.

To generate an instruction, we create 18 templates – e.g., “What are some steps to <question>?” – and use “how-to” questions to fill in the placeholders. We also include the number of steps in 14 out of 18 templates to control the model in terms of number of steps, similar to the length information for corpus-extracted examples. We provide all templates in §F.2.

We generate target texts by combining various elements, including introductory paragraphs (i.e., part 1), summaries and long descriptions of each step (i.e., part 2). We only include the first part in half of the examples. The second part is always

Type	Source	# of Examples
Reverse Instructions (RI)	C4 Wikipedia	10,000 5,000
Structured Corpora (SC)	Stack Exchange WikiHow	4,380 2,500
NLP Tasks (NLP)	NIv2 Big Bench BEA-GEC Enron	3,684 600 1,203 372
Total		27,739

Table 1: Origin statistics for LongForm-C. Examples are generated with reverse instructions, from structured corpora (SC) and NLP tasks (NLP).

included, but in different formats: either as a summary, a long description, or both.

3.3 NLP Tasks (NLP)

Most of instruction tuning benchmarks of NLP tasks involve classification or short text generation tasks, accounting for over 99% of BigBench and NIv2. However, we opt for NLP tasks with long outputs to enrich LongForm-C. From BigBench, we select Helpful Answer Generation (hhh) (Askell et al., 2021) and Minute Mysteries QA, a question answering dataset for short stories. From NIv2, we include a selection of ten tasks including poem generation (Hipson and Mohammad, 2020), table to text generation (Iyyer et al., 2017), story generation (Sap et al., 2020; Orbach and Goldberg, 2020; Lin et al., 2019), summarization (Fabbri et al., 2019; Kornilova and Eidelman, 2019; Bražinskas et al., 2020), and fact generation (Kotonya and Toni, 2020). For a more uniform distribution across datasets, we sample a similar number from each dataset, resulting in 600 samples from BigBench and 3,684 from NIv2.

Finally, we include the Enron dataset to create an email writing task from the subject using templates similar to WikiHow. We also employ the BEA-2019 for grammatical error correction (Bryant et al., 2019); incorrect inputs and template instruction are the input, the correct output the target. See §F.3 for templates for Enron and §F.4 for BEA-2019.

3.4 Dataset Analysis

Diversity: Following Wang et al. (2023b), we analyze the diversity of the generated instructions by parsing each instruction in the corpus subset with Berkeley Neural Parser (Kitaev and Klein, 2018). We extract either the verb and its direct object (noun+verb: "describe history") or the auxiliary with its dependent verb (auxiliary+verb: "do know"). 8,872 of 15,000 instructions follow this format. Figure 3 visualizes the 3,167 most common noun+verb/auxiliary+verb pairs. They cover a diverse range of topics, including events, careers, biographies, and writing tasks such as reviews, summaries and essays – and this is just looking at the 11% of our LongForm-C dataset with reverse instructions (RI). For a detailed comparison of LongForm-C's instruction diversity with other datasets such as Dolly (Conover et al., 2023) and FLAN (Chung et al., 2024), see Appendix (§G).

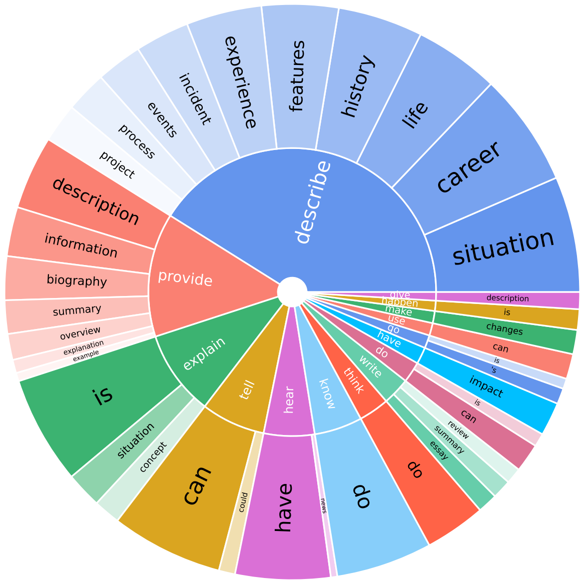

Figure 3: Most common noun+verb and auxiliary+verb pairs of generated instructions from the reverse instructions (RI) subset of LongForm-C.

Table 1 shows the distribution of LongForm-C: 54% of the examples are generated with reverse instructions, 25% from structured corpora, and 21% are reformulated from NLP tasks. The dataset is split 23,652/2,042/2,045 into training/validation/test.

4 Experimental Setup

4.1 LongForm Models

We create three LongForm models by finetuning language models on the LongForm-C dataset to assess their effectiveness on instruction-following text generation: **LongForm-T5-3B**, based on T5-3B (Raffel et al., 2020; Lester et al., 2021), an encoder-decoder LM, and two autoregressive LMs: **LongForm-OPT-6.7B**, based on OPT-6.7B (Zhang et al., 2022), and **LongForm-LLaMA-7B**, based on LLaMA-7B (Touvron et al., 2023a). We share the details of finetuning in the Appendix (§A).

4.2 Baselines

We compare our instruction-tuned LMs with other LMs that are instruction-tuned on other datasets. These LMs are based on either T5-11B (Raffel et al., 2020; Lester et al., 2021) or LLaMA-7B (Touvron et al., 2023a). We also compare our LMs with a raw large LM, OPT-30B (Zhang et al., 2022). Notably, all LMs have the same or higher number of parameters than our LongForm models.

OPT (Zhang et al., 2022) is an open-source decoder-only transformer LM, trained on diverse English corpora. Due to hardware limitations that make long text inference from LLMs time-consuming, we use the 30B variant of OPT.

T0++ (Sanh et al., 2022) is an LM-adapted variant of T5-11B, instruction-tuned on PromptSource (Bach et al., 2022), a dataset of >12 million examples from existing datasets.

T*k*-Instruct (Wang et al., 2022) is another instruction-tuned LM based on an LM-adapted version of T5-11B. The training data is a subsample of NIv2, which includes 757 tasks and over 75,000 examples. As we are performing zero-shot long text generation via instruction, we use the definition-only version of T*k*-Instruct.

Flan-T5 (Chung et al., 2024) is an instruction-tuned LM built on the LM-adapted version of T5-11B. The training data (total size 14M) includes tasks and prompts from multiple sources, including Muffin (Wei et al., 2022), T0-SF (Sanh et al., 2022), and NIv2 (Wang et al., 2022).

Alpaca is an instruction-tuned LM built on LLaMA-7B (Touvron et al., 2023a) by using a variation of Self-Instruct (Wang et al., 2023b). As they did not release their finetuned LM, we finetune LLaMA-7B with their variation of Self-Instruct.[1]

4.3 Evaluation

We evaluate the performance of the baselines and our instruction-tuned LMs on different sizes and architectures. For generation, we perform nucleus sampling (Holtzman et al., 2020) with $p = 0.9$ for all LMs. We first evaluate them on the test set of LongForm-C; recall that it consists of a diverse set of NLP tasks and corpus examples.

Additionally, we assess the generalization capabilities of our LMs on a set of long text generation datasets that were not seen during finetuning. These tasks include recipe generation (Bień et al., 2020) from a given set of ingredients, story generation for a given prompt from Writing Prompts (WP) subreddit, and long form question answering, ELI5 (Fan et al., 2019). We sample 250 examples from each task, resulting in 750 examples in total. Although these datasets are not part of LongForm-C, recipe generation is part of the instruction tuning data of two baselines, T*k*-Instruct and Flan-T5.

Next we analyze long-form generation and language understanding abilities. We compare the performance of LMs on out-of-domain long text generation tasks (i.e., RGen, ELI5, WP) and MMLU (Hendrycks et al., 2021). We compare LMs finetuned with the LongForm-C dataset and the FLAN collection,[2] which has a strong performance on MMLU (Chung et al., 2024). We also extend this analysis on more recent and powerful LLMs. In addition to LLaMA-7B, we compare FLAN and LongForm-C datasets on LLaMA 2-7B (Touvron et al., 2023b) and Mistral 7B (Jiang et al., 2023).

We also compare on multilingual text generation. We reformulate the multilingual summarization corpus MLSUM (Scialom et al., 2020) by providing an instruction in French, German, Spanish, Russian to generate news articles in those languages for given titles. We utilize the translation of the following instruction: “Write a news article for the given title: X” where X is the title and the output is the full news article. We collect 100 samples for each language and then evaluate on news article generation. We report results in §C in Appendix.

As current metrics in text generation have limited capabilities in evaluating long text generation (Celikyilmaz et al., 2020), we choose METEOR (Banerjee and Lavie, 2005) as our main metric as it exhibits higher human correlation (Sharma et al., 2017; Chen et al., 2022). However, we also report BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), and self-BLEU (Zhu et al., 2018) scores in the Appendix (§B), which exhibit similar patterns.

5 Results

We now present a comprehensive evaluation of our LongForm models compared to the baselines on LongForm-C test, recipe generation, long form question answering, short story generation, NLU, and ablation of LongForm-C subsets.

[1] github.com/tatsu-lab/stanford_alpaca/

[2] Wang et al. (2023a)’s FLAN version (Longpre et al., 2023)

	Model Size	Avg.	RI		SC		NLP			
			C4	Wiki	SE	WikiHow	Enron	BEA-GEC	NIv2	BigBench
T0++	11B	5.9	2.8	8.1	2.3	2.8	4.3	17.1	11.3	1.7
T*k*-Instruct	11B	6.0	2.2	9.4	3.2	1.3	3.8	12.4	13.5	2.5
Flan-T5	11B	12.5	4.1	13.3	5.8	4.1	8.4	70.8	17.2	3.2
Alpaca-LLaMA-7B	7B	15.2	8.4	22.2	9.2	10.7	6.4	55.3	16.9	8.6
OPT-30B	30B	12.3	14.2	13.7	14.4	8.0	7.0	16.8	9.5	3.3
LongForm-T5-XL	3B	24.3	18.9	22.4	15.3	21.5	8.7	**92.9**	**22.2**	17.8
LongForm-OPT-6.7B	6.7B	24.2	**19.7**	25.0	15.9	22.0	**8.9**	85.1	20.8	15.8
LongForm-LLaMA-7B	7B	**24.8**	19.6	**27.8**	**16.3**	**22.6**	6.6	86.2	21.2	**18.0**

Table 2: METEOR scores of baselines and our LMs on each subset of LongForm-C test and micro average ("Avg.") over the entire test set. All LongForm models outperform prior LMs on long text generation with a big margin. LongForm-T5-XL provides comparable results on text generation tasks (BEA-GEC, NIv2, BigBench), but the best results are achieved with LongForm-LLaMA-7B.

5.1 LongForm-C Results

We begin by comparing baseline LMs and LongForm models on the test set of LongForm-C. As presented in Table 2, all LongForm models demonstrate clear improvements in performance compared to the prior instruction-tuned and raw LMs. In particular, LongForm-T5-XL, with 3B parameters, outperforms all instruction-tuned LMs based on T5 (i.e., T0++, T*k*-Instruct, Flan-T5), even with 11B parameters, across all subtasks. This suggests the effectiveness of LongForm-C in finetuning LMs to follow instructions for long text generation.

Moreover, among our LMs, we see that LongForm-LLaMA-7B outperforms LongForm-OPT-6.7B, despite both having similar number of parameters, which aligns with findings from Touvron et al. (2023a). Additionally, we observe that LongForm-T5-XL has comparable or better results than auto-regressive variants, particularly on text generation (i.e., BEA-GEC, NIv2, BigBench). However, a potential limitation of T5 variants is the absence of newline tokens in their vocabulary, which may impact the readability of the generated output, particularly for long text generation.

Among baseline LMs, Alpaca, the only instruction tuned LM without reformulated NLP tasks, achieves the best overall performance. However, it still performs worse than our LongForm models, potentially due to the noise in its training data, Self-Instruct. Furthermore, OPT-30B, despite having the highest number of parameters, fails to follow instructions as effectively as instruction-tuned LongForm-T5-XL and LongForm-OPT-2.7B, which are ten times smaller (see Figure 4). This observation is consistent with prior findings that demonstrate the effectiveness of instruction tuning in smaller LMs (Ouyang et al., 2022).

	Avg.	RGen	ELI5	WP
T0++	10.9	18.7	3.8	10.2
T*k*-Instruct	6.3	12.9†	3.6	2.4
Flan-T5	10.6	20.9†	3.5	7.4
Alpaca-LLaMA-7B	14.6	19.5	12.5	11.8
OPT-30B	11.1	18.6	12.2	2.6
LongForm-T5-XL	16.3	20.2	18.3	10.6
LongForm-OPT-6.7B	17.7	16.9	17.2	**19.0**
LongForm-LLaMA-7B	**19.7**	**21.7**	**18.6**	18.9

Table 3: METEOR scores for out-of-domain datasets. LongForm models outperform prior instruction-tuned LMs on recipe generation (RGen), long-form question answering (ELI5) and short story generation (WP). †: T*k*-Instruct and FLAN-T5 are trained on RGen.

5.2 Out-of-domain Generalization

To evaluate the generalization capability of LongForm models, we evaluate on three diverse datasets: recipe generation (RGen) (Bień et al., 2020), long form question answering (ELI5) (Fan et al., 2019), and short story generation (WP). We note that the LongForm models are not trained on any of these three datasets while the training set of T*k*-Instruct and Flan-T5 includes RGen.

Table 3 shows that the LongForm models consistently outperform baseline LMs in all tasks, with LongForm-LLaMA-7B achieving a 35% higher relative METEOR score than the best baseline LM, Alpaca-LLaMA-7B. While they both have the same underlying LM, our LM achieves higher scores than Alpaca. This can be attributed to the diverse and human-written nature of LongForm-C compared to Alpaca's fully synthetic examples.

We present qualitative examples from the best

Model	Dataset	NLG	NLU (MMLU)
LLaMA-7B	FLAN	9.1	36.6
	LongForm-C	**19.7**	35.2
	FLAN+LongForm-C	16.5	**38.9**
LLaMA 2-7B	FLAN	13.3	43.5
	LongForm-C	**19.3**	44.7
	FLAN+LongForm-C	17.8	**46.2**
Mistral 7B	FLAN	18.9	57.7
	LongForm-C	**22.0**	**58.8**
	FLAN+LongForm-C	19.4	58.3

Table 4: LongForm-C consistently outperforms FLAN on NLG tasks (RGen, ELI5, WP) across all models. For LLaMA-7B, FLAN+LongForm-C achieves the best MMLU score, while LongForm-C outperforms FLAN in both NLG and NLU with LLaMA 2-7B and Mistral 7B: the improvements with reverse instructions are consistent and even more powerful with stronger models.

performing LongForm model and the best baseline LM, Alpaca, in Figure 5 in the Appendix (§E). This illustrates that LongForm models have better factual accuracy, instruction-following, long form text planning than prior instruction-tuned LMs.

5.3 Language Understanding and Generation

We evaluate how much LongForm-C improves natural language understanding (NLU) in addition to NLG. To this end, we compare LongForm-C with the strong baseline FLAN (Longpre et al., 2023), which performs well on NLU. We finetune base LMs using LongForm-C, FLAN and their combination, evaluating performance on OOD NLG (i.e., recipe generation, ELI5 question answering, and story generation) and 5-shot MMLU. We extend our evaluation to newer, more powerful models than LLaMA-7B: LLaMA 2-7B and Mistral 7B.

Table 4 shows that LLaMA-7B finetuned with LongForm-C outperforms the FLAN baseline in NLG (+10.6). Incorporating LongForm-C with FLAN achieves the best MMLU performance (+2.3), though the combination gets lower NLG scores than LongForm-C alone (-3.2).

Extending LLMs: Our analysis to more recent LLMs shows promising results. We observe that, with stronger models, LongForm-C shows better results than FLAN (both in NLU and NLG). With LLaMA 2-7B, LongForm-C again improves NLG (+6.0) while narrowing the gap in NLU, even surpassing FLAN's MMLU score (+1.2). The FLAN+LongForm-C combination further improves MMLU (+2.7). With Mistral 7B, LongForm-C outperforms both FLAN and the combination, not only on NLG (+3.1, +2.6) but also on MMLU (+1.1, +0.5). These findings indicate that LongForm-C consistently improves NLG performance across LLMs, with its impact on NLU becoming more pronounced in stronger LLMs. This suggests that more natural datasets like LongForm-C may lead to better improvements in stronger LLMs than NLP reformulation datasets like FLAN.

Model	Subset	NLG	NLU (MMLU)
LLaMA-7B	NLP	13.6	35.4
	NLP+SC	18.6	35.0
	NLP+RI	**19.9**	**35.5**
	NLP+RI+SC	19.7	35.2

Table 5: Subset ablation experiments to study the effect of different subsets in LongForm-C. Adding the Reverse Instructions subset further improves the performance both on NLG and NLU showing its capabilities while NLP and SC subsets show limited improvements.

5.4 Ablation of LongForm-C Subsets

We now conduct an ablation study to investigate the differential impact of the three distinct subsets of LongForm-C by finetuning LLaMA-7B: NLP Tasks (NLP), Structured Corpora (SC), and Reverse Instructions (RI).

We summarize the results in Table 5. Since the NLP tasks subset already includes examples with long form output, the model performs comparatively well on NLG, achieving better performance than the performance of FLAN (see Table 4), another NLP tasks dataset for instruction-tuning. We observe a trade-off between NLG (+5.0) and NLU (-0.4) performance when we include the SC subset in addition to NLP tasks. However, including the Reverse Instructions subset significantly improves NLG (+6.3) and also slightly improves NLU (+0.1). Despite the instructions in the SC subset being created directly from human annotation (StackExchange) or via minimal changes (WikiHow), the improvement by Reverse Instructions is substantially better and more consistent. We achieve the best results when the Reverse Instructions (RI) subset is included.

5.5 Data Evaluation

Our instruction generation (for reverse instructions) relies on LLM generation. To ensure its quality, we conduct a human evaluation. We randomly select 100 corpus examples from LongForm-C (without adding length information) and provide them to Mechanical Turk annotators. Annotators are asked to determine whether the generated instruction is relevant to the given document or not. 97 out of 100 instructions are found to be relevant. This high degree of relevance suggests that combining instruction generation with corpus examples can potentially improve the effectiveness of LLMs for instruction-following compared to the Self-Instruct dataset (Wang et al., 2023b), which relies on LLMs to generate tasks, instructions, and outputs, and achieves a validity of 54%.

We conducted an additional manual analysis of a corpus subset of LongForm-C to identify areas for improvement in corpus mining and instruction tuning, providing valuable insights for future work. We present qualitative examples in Table 11 in the Appendix. Our analysis shows that the failure cases mostly stem from the detailed nature of corpus texts such as news. While reverse instructions is capable of generating relevant instructions, the corpus examples can sometimes include extensive content. This tendency can lead LongForm models to generate long responses even for simple questions that require concise answers. This issue occurred infrequently, as supported by the overall performance of LongForm. However, this is still an important insight for future work in this area.

6 Conclusion

In this paper, we introduce reverse instructions, a novel method to transform human-written texts into instruction-output pairs via LLMs. We present LongForm-C, an instruction-following long text generation dataset that combines reverse instructions examples, structured corpora (SC), and NLP tasks. Our evaluation shows that the generated reverse instructions are highly relevant to the corpus examples and contain a diverse set of tasks. Furthermore, we demonstrate that our LongForm models – instruction-tuned on LongForm-C– outperform prior instruction-tuned baselines such as FLAN-T5 and Alpaca by a big margin on a wide variety of long text generation tasks.

Ethics Statement

LongForm models have capabilities of generating high-quality long texts, such as news articles, essays, and tutorials by following given instructions. Therefore, this may raise concerns regarding the potential for malicious use. However, we recognize the importance of publicly sharing our models and datasets to facilitate further research. By making LongForm models easily accessible, we hope to support researchers to explore potential implications of instruction-following large language models, including techniques for improving their truthfulness or watermarking model outputs. Therefore, we believe that the possible benefits of LongForm models for the research community outweigh their possible risks.

Limitations

The proposed datasets and models mainly focus on long text generation and may have limitations regarding structured prediction tasks in NLP with some LLMs. We observe that LongForm-C is a good complementary dataset and when combined with existing datasets, it further improves NLU performance. However, for more recent models, it outperforms previous datasets not only in NLG but also on structured prediction tasks. Additionally, we observe that LongForm models may present hallucination problems similar to those found in LLMs.

Acknowledgements

This work was funded by Deutsche Forschungsgemeinschaft (project SCHU 2246/14-1).

References

Mikel Artetxe and Holger Schwenk. 2019. Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3197–3203, Florence, Italy. Association for Computational Linguistics.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Benjamin Mann, Nova DasSarma, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Jackson Kernion, Kamal Ndousse, Catherine Olsson, Dario Amodei, Tom B. Brown, Jack Clark, Sam McCandlish, Chris Olah, and Jared Kaplan. 2021. A general language assistant as a laboratory for alignment. *CoRR*, abs/2112.00861.

Stephen Bach, Victor Sanh, Zheng Xin Yong, Albert Webson, Colin Raffel, Nihal V. Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry, Zaid Alyafeai, Manan Dey, Andrea Santilli, Zhiqing Sun, Srulik Ben-david, Canwen Xu, Gunjan Chhablani, Han Wang, Jason Fries, Maged Alshaibani, Shanya Sharma, Urmish Thakker, Khalid Almubarak, Xiangru Tang, Dragomir Radev, Mike Tian-jian Jiang, and Alexander Rush. 2022. PromptSource: An integrated development environment and repository for natural language prompts. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 93–104, Dublin, Ireland. Association for Computational Linguistics.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Michał Bień, Michał Gilski, Martyna Maciejewska, Wojciech Taisner, Dawid Wisniewski, and Agnieszka Lawrynowicz. 2020. RecipeNLG: A cooking recipes dataset for semi-structured text generation. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 22–28, Dublin, Ireland. Association for Computational Linguistics.

Arthur Bražinskas, Mirella Lapata, and Ivan Titov. 2020. Few-shot learning for opinion summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4119–4135, Online. Association for Computational Linguistics.

Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. 2019. The BEA-2019 shared task on grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75, Florence, Italy. Association for Computational Linguistics.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020. Evaluation of text generation: A survey. *CoRR*, abs/2006.14799.

Yiran Chen, Zhenqiao Song, Xianze Wu, Danqing Wang, Jingjing Xu, Jiaze Chen, Hao Zhou, and Lei Li. 2022. MTG: A benchmark suite for multilingual text generation. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2508–2527, Seattle, United States. Association for Computational Linguistics.

Yongrui Chen, Haiyun Jiang, Xinting Huang, Shuming Shi, and Guilin Qi. 2024. DoG-instruct: Towards premium instruction-tuning data via text-grounded instruction wrapping. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4125–4135, Mexico City, Mexico. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. Free dolly: Introducing the world's first truly open instruction-tuned llm.

Zhuyun Dai, Arun Tejasvi Chaganty, Vincent Y Zhao, Aida Amini, Qazi Mamunur Rashid, Mike Green, and Kelvin Guu. 2022. Dialog inpainting: Turning documents into dialogs. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 4558–4586. PMLR.

Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. 2021. Documenting large webtext corpora: A case study on the colossal clean crawled corpus. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1286–1305, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Liat Ein-Dor, Eyal Shnarch, Lena Dankin, Alon Halfon, Benjamin Sznajder, Ariel Gera, Carlos Alzate, Martin Gleize, Leshem Choshen, Yufang Hou, et al. 2020. Corpus wide argument mining—a working solution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7683–7691.

Alexander Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. 2019. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy. Association for Computational Linguistics.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3558–3567, Florence, Italy. Association for Computational Linguistics.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

Xiaocheng Feng, Ming Liu, Jiahao Liu, Bing Qin, Yibo Sun, and Ting Liu. 2018. Topic-to-essay generation with neural networks. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 4078–4084. ijcai.org.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2020. The pile: An 800gb dataset of diverse text for language modeling.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Will Hipson and Saif M. Mohammad. 2020. PoKi: A large dataset of poems by children. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1578–1589, Marseille, France. European Language Resources Association.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. 2023. Unnatural instructions: Tuning language models with (almost) no human labor. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14409–14428, Toronto, Canada. Association for Computational Linguistics.

Mohit Iyyer, Wen-tau Yih, and Ming-Wei Chang. 2017. Search-based neural structured learning for sequential question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1821–1831, Vancouver, Canada. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Nikita Kitaev and Dan Klein. 2018. Constituency parsing with a self-attentive encoder. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2676–2686, Melbourne, Australia. Association for Computational Linguistics.

Anastassia Kornilova and Vladimir Eidelman. 2019. BillSum: A corpus for automatic summarization of US legislation. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 48–56, Hong Kong, China. Association for Computational Linguistics.

Neema Kotonya and Francesca Toni. 2020. Explainable automated fact-checking for public health claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7740–7754, Online. Association for Computational Linguistics.

Abdullatif Köksal, Marion Thaler, Ayyoob Imani, Ahmet Üstün, Anna Korhonen, and Hinrich Schütze. 2024. Muri: High-quality instruction tuning datasets for low-resource languages via reverse instructions.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Patrick Lewis, Yuxiang Wu, Linqing Liu, Pasquale Minervini, Heinrich Küttler, Aleksandra Piktus, Pontus Stenetorp, and Sebastian Riedel. 2021. PAQ: 65 million probably-asked questions and what you can do with them. *Transactions of the Association for Computational Linguistics*, 9:1098–1115.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Luke Zettlemoyer, Omer Levy, Jason Weston, and Mike Lewis. 2023. Self-alignment with instruction backtranslation.

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.

Kevin Lin, Oyvind Tafjord, Peter Clark, and Matt Gardner. 2019. Reasoning over paragraph effects in situations. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 58–62, Hong Kong, China. Association for Computational Linguistics.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. 2023. The flan collection: designing data and methods for effective instruction tuning. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Shayne Longpre, Yi Lu, Zhucheng Tu, and Chris DuBois. 2019. An exploration of data augmentation and sampling techniques for domain-agnostic question answering. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 220–227, Hong Kong, China. Association for Computational Linguistics.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti

Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation.

Eyal Orbach and Yoav Goldberg. 2020. Facts2Story: Controlling text generation by key facts. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2329–2345, Barcelona, Spain (Online). International Committee on 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 Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

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

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

Philip Resnik. 1999. Mining the web for bilingual text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 527–534, College Park, Maryland, USA. Association for Computational Linguistics.

Gözde Gül Şahin and Mark Steedman. 2018. Data augmentation via dependency tree morphing for low-resource languages. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5004–5009, Brussels, Belgium. Association for Computational Linguistics.

Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M Rush. 2022. Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations*.

Maarten Sap, Eric Horvitz, Yejin Choi, Noah A. Smith, and James Pennebaker. 2020. Recollection versus imagination: Exploring human memory and cognition via neural language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1970–1978, Online. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021. Generating datasets with pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6943–6951, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. 2020. MLSUM: The multilingual summarization corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8051–8067, Online. Association for Computational Linguistics.

Shikhar Sharma, Layla El Asri, Hannes Schulz, and Jeremie Zumer. 2017. Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation.

Nakatani Shuyo. 2010. Language detection library for java.

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, Ambrose Slone, Ameet Rahane, Anantharaman S. Iyer, Anders Johan Andreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew M. Dai, Andrew La, Andrew 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 Karakaş, B. Ryan Roberts, Bao Sheng Loe, Barret Zoph, Bartłomiej Bojanowski, Batuhan Özyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Howald, Bryan Orinion, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, Cesar Ferri, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Christopher Waites, Christian Voigt, Christopher D Manning, Christopher Potts, Cindy Ramirez, Clara E. Rivera, Clemencia Siro, Colin Raffel, Courtney Ashcraft, Cristina Garbacea, Damien Sileo, Dan Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, C. Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Moseguí González, Danielle Perszyk, Danny Hernandez, Danqi Chen, Daphne Ippolito, Dar Gilboa, David Dohan, David Drakard, David Jurgens, Debajyoti Datta, Deep Ganguli, Denis Emelin, Denis Kleyko, Deniz Yuret, Derek Chen, Derek Tam, Dieuwke Hupkes, Diganta Misra, Dilyar Buzan, Dimitri Coelho Mollo, Diyi Yang, Dong-Ho Lee, Dylan Schrader, Ekaterina Shutova, Ekin Dogus Cubuk, Elad Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth Donoway, Ellie Pavlick, Emanuele Rodolà, Emma Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang, Ethan A Chi, Ethan Dyer, Ethan Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, Fernando Martínez-Plumed, Francesca Happé, Francois Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Xinyue Wang, Gonzalo Jaimovitch-Lopez, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Francis Anthony Shevlin, Hinrich Schuetze, Hiromu Yakura, Hongming Zhang, Hugh Mee Wong, Ian Ng, Isaac Noble, Jaap Jumelet, Jack Geissinger, Jackson Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, James B Simon, James Koppel, James Zheng, James Zou, Jan Kocon, Jana Thompson, Janelle Wingfield, Jared Kaplan, Jarema Radom, Jascha Sohl-Dickstein, Jason Phang, Jason Wei, Jason Yosinski, Jekaterina Novikova, Jelle Bosscher, Jennifer Marsh, Jeremy Kim, Jeroen Taal, Jesse Engel, Jesujoba Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Joan Waweru, John Burden, John Miller, John U. Balis, Jonathan Batchelder, Jonathan Berant, Jörg Frohberg, Jos Rozen, Jose Hernandez-Orallo, Joseph Boudeman, Joseph Guerr, Joseph Jones, Joshua B. Tenenbaum, Joshua S. Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakrishnan, Katerina Ignatyeva, Katja Markert, Kaustubh Dhole, Kevin Gimpel, Kevin Omondi, Kory Wallace Mathewson, Kristen Chiafullo, Ksenia Shkaruta, Kumar Shridhar, Kyle McDonell, Kyle Richardson, Laria Reynolds, Leo Gao, Li Zhang, Liam Dugan, Lianhui Qin, Lidia Contreras-Ochando, Louis-Philippe Morency, Luca Moschella, Lucas Lam, Lucy Noble, Ludwig Schmidt, Luheng He, Luis Oliveros-Colón, Luke Metz, Lütfi Kerem Senel, Maarten Bosma, Maarten Sap, Maartje Ter Hoeve, Maheen Farooqi, Manaal Faruqui, Mantas Mazeika, Marco Baturan, Marco Marelli, Marco Maru, Maria Jose Ramirez-Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew L Leavitt, Matthias Hagen, Mátyás Schubert, Medina Orduna Baitemirova, Melody Arnaud, Melvin McElrath, Michael Andrew Yee, Michael Cohen, Michael Gu, Michael Ivanitskiy, Michael Starritt, Michael Strube, Michał Swędrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Mitch Walker, Mo Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T, Nanyun Peng, Nathan Andrew Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick Doiron, Nicole Martinez, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish Keskar, Niveditha S. Iyer, Noah Constant, Noah Fiedel, Nuan Wen, Oliver Zhang, Omar Agha, Omar Elbaghdadi, Omer Levy, Owain Evans, Pablo Antonio Moreno Casares, Parth Doshi, Pascale Fung, Paul Pu Liang, Paul Vicol, Pegah Alipoormolabashi, Peiyuan Liao, Percy Liang, Peter W Chang, Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Miłkowski, Piyush Patil, Pouya Pezeshkpour, Priti Oli, Qiaozhu Mei, Qing Lyu, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer Gabriel, Rahel Habacker, Ramon Risco, Raphaël Millière, Rhythm Garg, Richard Barnes, Rif A. Saurous, Riku Arakawa, Robbe Raymaekers, Robert Frank, Rohan Sikand, Roman Novak, Roman Sitelew, Ronan Le Bras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Russ Salakhutdinov, Ryan Andrew Chi, Seungjae Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib Singh, Saif M. Mohammad, Sajant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Samuel R. Bowman, Samuel Stern Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A. Rous, Sarik Ghazarian, Sayan Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, Shadi Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima Shammie Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, Soo-Hwan Lee, Spencer Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano Ermon, Stella Biderman, Stephanie Lin, Stephen Prasad, Steven Piantadosi, Stuart Shieber, Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsunori Hashimoto, Te-Lin Wu, Théo Desbordes, Theodore Rothschild, Thomas Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, Timofei Kornev, Titus Tunduny, Tobias Gerstenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler Shultz, Uri Shaham, Vedant Misra, Vera Demberg, Victoria Nyamai, Vikas Raunak, Vinay Venkatesh Ramasesh, vinay uday prabhu, Vishakh Padmakumar, Vivek Srikumar, William Fedus, William Saunders, William Zhang, Wout Vossen, Xiang Ren, Xiaoyu Tong, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yangqiu Song, Yasaman Bahri, Yejin Choi, Yichi Yang, Yiding Hao, Yifu Chen, Yonatan Belinkov, Yu Hou, Yufang Hou, Yuntao Bai, Zachary Seid, Zhuoye Zhao, Zijian Wang, Zijie J. Wang, Zirui Wang, and Ziyi Wu. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models.

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Chandu, David Wadden, Kelsey MacMillan, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. 2023a. How far can camels go? exploring the state of instruction tuning on open resources. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023b. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yue Wang, Xinrui Wang, Juntao Li, Jinxiong Chang, Qishen Zhang, Zhongyi Liu, Guannan Zhang, and Min Zhang. 2023c. Harnessing the power of david against goliath: Exploring instruction data generation without using closed-source models.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. 2022. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, Qingwei Lin, and Daxin Jiang. 2024. WizardLM: Empowering large pre-trained language models to follow complex instructions. In *The Twelfth International Conference on Learning Representations*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pre-trained transformer language models.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 1097–1100, New York, NY, USA. Association for Computing Machinery.

Ingo Ziegler, Abdullatif Köksal, Desmond Elliott, and Hinrich Schütze. 2024. Craft your dataset: Task-specific synthetic dataset generation through corpus retrieval and augmentation.

A Training Details

We include an end-of-instruction token, [EOI], between the instruction and the output of the autoregressive LMs, OPT and LLaMA variants, to differentiate between instruction and output. We employ two to four A6000 48GB GPUs for finetuning with a batch size of 32 to 64, and train with DeepSpeed, including bfloat16 training. The learning rate is determined by comparing the minimum validation loss during three epochs: 5e-5 for LongForm-T5-3B, 5e-6 for LongForm-OPT-6.7B and LongForm-LLaMA-7B. Apart from the three large LMs, we finetune four smaller OPT LMs (125M, 350M, 1.3B, 2.7B) with LongForm-C and release them to increase accessibility of instruction-tuned LMs. See §D for more details.

B Evaluation with Different Metrics

We extend our evaluation setup with BLEU, ROUGE, and self-BLEU in addition to the METEOR metric to show more robust results. The results indicate that we have similar patterns in all parts of evaluation. We report average in-domain performance in Table 6, out-of-domain performance in Table 7, and multilingual performance in Table 8.

C Multilingual News Generation

We evaluate the multilingual news generation capabilities of LongForm models against the baselines. Although all LMs in our experiments are pretrained and instruction-tuned mainly on English, their pretraining data includes other languages.[3] Table 9 shows that LongForm models outperform baselines on the task of generating a news article given a title in Spanish, French, and Russian.

We observe that LongForm-LLaMA-7B shows the best performance in the multilingual setup as well. We also observe that all OPT variants (LongForm-OPTs and OPT-30B) have a higher rate of generating text in the given language (see Table 10 in the Appendix), which results in better performance for OPT. In cases where the LM generates English text while the instruction is in a different language, our manual inspection shows that LongForm models can still follow the non-English instruction. Additionally, when we generate with different seeds, we observe that LongForm models are able to generate in the language of the instruction. We present qualitative multilingual examples in Figure 7 in the Appendix. Finally, we believe that our methodology (the use of corpus examples for instruction tuning data) is extensible to other languages, with less reliance on multilingually highly competent LLMs.

We observe that while we ask instructions in different languages, the models might have tendencies to respond in English. We present the distribution of outputs in the language that is given in the instruction in Table 10, detected by the language-detection library (Shuyo, 2010). We see that raw language model, OPT-30B, and LongForm-OPT-6.7B can respond with higher ratio in the given language. While Flan-T5 and T*k*-Instruct have higher ratio of generating text in the language of instruction, it is important to note that the quality of the generated text is much lower as it can be seen in Table 9.

	DE	**ES**	**FR**	**RU**
T0++	0.09	0.12	0.07	0.56
T*k*-Instruct	0.96	0.95	0.97	0.55
Flan-T5	0.98	0.95	0.99	0.45
Alpaca-LLaMA-7B	0.89	0.90	0.26	0.23
OPT-30B	0.96	0.99	1.00	1.00
LongForm-T5-XL	0.35	0.74	0.57	0.89
LongForm-OPT-6.7B	0.50	0.88	0.94	1.00
LongForm-LLaMA-7B	0.77	0.91	0.93	0.44

Table 10: The ratio of generating text in the language of the instruction.

D Smaller LongForm Models

In order to increase accessibility, we finetune varying size of PLMs, OPT-125M, OPT-350M, OPT-1.3B, OPT-2.7B, and OPT-6.7B models with the LongForm-C dataset. We see that all LongForm-OPT models have better METEOR scores than OPT-30B without instruction tuning. We present average METEOR scores of LongForm-OPT models and OPT-30B in Figure 4.

[3] 20 languages from Wikipedia are included in LLaMA, which accounts for 4.5% of the entire pretraining data.

	METEOR	BLEU	ROUGE-L	self-BLEU
T0++	5.9	0.0	8.6	0.02
T*k*-Instruct	6.0	0.0	9.2	0.02
Flan-T5	12.5	0.1	16.0	0.04
Alpaca-LLaMA-7B	15.2	1.8	20.4	0.10
OPT-30B	12.3	2.1	13.4	0.18
LongForm-T5-XL	24.3	**9.4**	26.0	**0.19**
LongForm-OPT-6.7B	24.2	8.8	27.1	**0.19**
LongForm-LLaMA-7B	**24.8**	8.8	**27.6**	0.16

Table 6: Average in-domain performance of baseline and LongForm models with METEOR, BLEU, ROUGE, and self-BLEU. METEOR scores of each subset are reported in the Table 2.

	METEOR	BLEU	ROUGE-L	self-BLEU
T0++	10.9	0.3	14.4	0.07
T*k*-Instruct	6.3	0.0	9.4	0.04
Flan-T5	10.6	0.2	14.5	0.07
Alpaca-LLaMA-7B	14.6	1.5	**22.1**	0.13
OPT-30B	11.1	0.8	11.6	0.12
LongForm-T5-XL	16.3	1.7	16.2	0.22
LongForm-OPT-6.7B	17.7	1.4	17.4	**0.27**
LongForm-LLaMA-7B	**19.7**	**2.1**	21.6	0.20

Table 7: Average out-of-domain performance of baseline and LongForm models with METEOR, BLEU, ROUGE, and self-BLEU. METEOR scores of each subset are reported in the Table 3.

	METEOR	BLEU	ROUGE-L	self-BLEU
T0++	1.8	0.0	1.8	0.12
T*k*-Instruct	0.8	0.0	1.9	0.01
Flan-T5	2.4	0.0	4.3	0.08
Alpaca-LLaMA-7B	4.9	0.1	7.7	0.14
OPT-30B	4.5	0.0	5.8	0.07
LongForm-T5-XL	6.9	0.4	5.8	**0.39**
LongForm-OPT-6.7B	9.5	0.3	10.5	0.17
LongForm-LLaMA-7B	**9.7**	**0.6**	**10.7**	0.18

Table 8: Average multilingual performance of baseline and LongForm models with METEOR, BLEU, ROUGE, and self-BLEU. METEOR scores of each language are reported in the Table 9.

	Avg.	DE	ES	FR	RU
T0++	1.8	1.6	2.4	2.5	0.7
T*k*-Instruct	0.8	0.8	1.1	1.0	0.5
Flan-T5	2.4	2.5	2.8	3.4	0.7
Alpaca-LLaMA-7B	4.9	7.9	7.0	4.1	0.7
OPT-30B	4.5	**12.7**	3.5	1.3	0.5
LongForm-T5-XL	6.9	5.2	9.0	9.1	4.2
LongForm-OPT-6.7B	9.5	7.3	**12.2**	**15.3**	3.1
LongForm-LLaMA-7B	**9.7**	8.9	10.5	14.8	**4.7**

Table 9: Evaluation of multilingual news generation via METEOR.

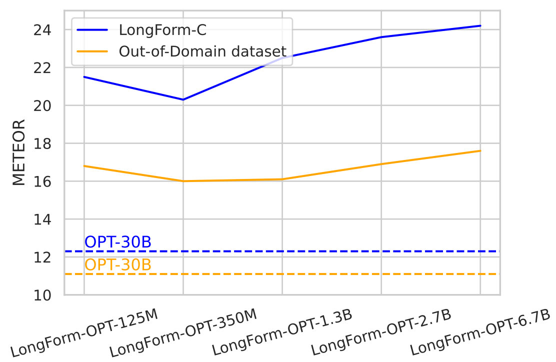

Figure 4: Comparing METEOR scores for varying sizes of LongForm-OPT models on in-domain (LongForm-C) and out-of-domain (RGen, ELI5, WP) tasks. All instruction-tuned LongForm-OPT models outperform OPT-30B (dashed) without instruction tuning.

E Qualitative Examples

We present qualitative examples from the best performing LongForm model and the best baseline model, Alpaca, for each task in Figure 5.

F Details of LongForm-C

F.1 LLM Query

We use OpenAI's text-davinci-003 Completion API for our LLM query part. We just use the default parameters in which temperature and top_p parameters are 1.

The template for the instruction style:

Instruction: X
Output: <*corpus_example*>
What kind of instruction could this be the answer to?

X:

The template for the informal chatbot style:

You are a chatbot. A user sent you an informal message and your reply is as follows.
Message: X
Reply: <*corpus_example*>

What is the informal message X?

X:

The template for the search engine/query style:

You are a search engine. A person queried something in detail and the most relevant document about the query is as follows.
Query: X
Document: <*corpus_example*>
What is the detailed query X?

X:

F.2 WikiHow Templates

Below, we list 18 WikiHow templates utilized in creating the LongForm-C. These templates feature questions in the form of verb phrases, for example, “make conversation” in addition to number of steps.

1. Give me [STEP] steps to [QUESTION].
2. How to [QUESTION]?
3. Do you know how can I [QUESTION]?
4. List [STEP] instructions to [QUESTION].
5. What are some tips to [QUESTION]?
6. What are some steps to [QUESTION]?
7. Can you provide [STEP] clear and concise instructions on how to [QUESTION]?
8. I'm interested in learning how to [QUESTION]. Could you break it down into [STEP] easy-to-follow steps?
9. For someone who is new to [QUESTION], what would be [STEP] key steps to get started?
10. What is the most efficient way to [QUESTION]? Could you provide a list of [STEP] steps?
11. Do you have any advice on how to [QUESTION] successfully? Maybe a step-by-step guide with [STEP] steps?
12. I'm trying to accomplish [QUESTION]. Could you walk me through the process with [STEP] detailed instructions?
13. What are the essential [STEP] steps to [QUESTION]?
14. I need to [QUESTION], but I'm not sure where to start. Can you give me [STEP] actionable steps?
15. As a beginner in [QUESTION], what are the [STEP] basic steps I should take?
16. I'm looking for a comprehensive guide on how to [QUESTION]. Can you provide [STEP] detailed steps?
17. Could you outline [STEP] practical steps to achieve [QUESTION]?
18. What are the [STEP] fundamental steps to consider when attempting to [QUESTION]?

F.3 Enron Templates

We list the instructions for email writing task for the given subject.

1. Write an email with the subject "[SUBJ]"
2. Can you craft an email with the subject [SUBJ]?
3. Would you be able to compose an email and use [SUBJ] as the subject?

Recipe Generation Instruction: Ingredients: beef, mushroom soup, mushrooms. Write directions of a cooking recipe with these ingredients.	
Alpaca-LLaMA-7B completion: In a pot, add a can of mushroom soup and 2 cups of beef broth. Heat to a simmer and add 3 cups of sliced mushrooms. Let simmer for 15 minutes. Serve over cooked noodles or rice. Enjoy!	**LongForm-LLaMA-7B completion:** Put the meat in a large saucepan or skillet and cover it with water. Add 1 tsp (5 ml) of salt and bring it to a boil. Remove the meat from the pan and discard the water. Add 1/2 can (125 ml) of mushroom soup and 2 cups (473 ml) of mushrooms and simmer it for 15 minutes. Sprinkle with salt and pepper and garnish with sour cream and parsley before serving.
ELI5 Instruction: how do digital cameras capture and transfer what they see into a .jpg?	
Alpaca-LLaMA-7B completion: Digital cameras capture and transfer the light reflected from an object into a.jpg file using a process called image compression. The image compression algorithm takes a series of pixels from the image and arranges them into blocks of similar colors. It then reduces the amount of data needed to store the image by removing redundant information from the blocks. The final result is an.jpg file that can be stored, edited, and shared.	**LongForm-LLaMA-7B completion:** It doesn't matter how the camera captures the picture; what it captures is a signal, which is an analogue value. That signal is either converted to digital form by a linear analog-to-digital converter (ADC), usually at the image sensor itself, or it's first converted to a voltage which is then read in by a linear ADC, either at the image sensor or in the digital part of the image processing pipeline. The digital value of the ADC is converted by software into a.jpg. There's a lot of clever maths going on behind the scenes; but the crucial part is that it is a digital value which is converted to a.jpg.
WritingPrompts Instruction: Prompt: The heroes have finally done it! They've overthrown the evil overlord, destabilized his government and saved the world! While celebrating they very suddenly realize that they have no idea how to run a government. Write a story about given prompt.	
Alpaca-LLaMA-7B completion: It was a momentous occasion, a day the heroes would never forget. They had banded together to save the world from the evil overlord, and their triumph was complete. After the celebrations had died down, however, the heroes realized they had no idea how to run a government. The world needed a new leader, but who should it be? What decisions should they make? The heroes knew they were up to the task, but they needed to think carefully if they wanted to do it right. *(85 words)*	**LongForm-LLaMA-7B completion:** It was over. The final battle, the last stand, and the victory celebration had concluded and the final tally was in: the villains were dead, their minions were scattered or captured, the last of the evil artifacts were destroyed, and the world had been saved. The cheering heroes took a break from their celebrations to hear the good news from the new king of the newly formed world government. He had good news: the world was once again at peace, all villain threats were eliminated, and a new era of prosperity was just beginning. All that remained were some minor issues of governance and organization. He was currently working on those and would be happy to hand over full control to the new rulers as soon as he finished. The heroes looked at each other with doubt and dismay. They were heroes, not politicians. They couldn't run a world. They were terrible at that sort of thing. Then the realization struck. "You want us to be your government?" The new king was taken aback. "Yes. That's why you're here. To take over for me. All I need is your names and titles and I can hand over all power." The heroes had a serious crisis. *(205 words)*

Figure 5: Generalization capabilities of LongForm for three out-of-domain tasks: long form Q&A, story and recipe generation. LongForm generates well-structured, reliable, and creative long texts compared to the best baseline model. In particular, Alpaca suffers from factual inaccuracies in ELI5, limited creativity in WritingPrompts with shorter response, and a tendency to ignore ingredients in RGen. Red highlights indicate where the model may be hallucinating or deviating from instructions while green highlights accurate generation.

4. Create an email about [SUBJ].
5. Draft an email and include the subject "[SUBJ]".
6. Generate an email about [SUBJ].
7. Hey, can you shoot me an email about [SUBJ]?
8. Do you mind crafting an email for me with [SUBJ] as the subject?
9. Can you whip up an email with the subject of "[SUBJ]"?
10. Hey, can you write an email and use "[SUBJ]" as the subject?
11. Can you send me an email about [SUBJ]?

F.4 BEA-GEC Templates

We list the instructions for grammar error correction task. We prepend each input to randomly selected instruction.

1. Edit and revise this document to improve its grammar, vocabulary, spelling, and style.
2. Revise this document to correct all the errors related to grammar, spelling, and style.
3. Refine this document by eliminating all grammatical, lexical, and orthographic errors and improving its writing style.
4. Polish this document by rectifying all errors related to grammar, vocabulary, and writing style.
5. Enhance this document by correcting all the grammar errors and style issues, and improving its overall quality.
6. Rewrite this document by fixing all grammatical, lexical and orthographic errors
7. Fix all grammar errors and style issues and rewrite this document
8. Take a stab at fixing all the mistakes in this document and make it sound better.
9. Give this document a once-over and clean up any grammar or spelling errors.
10. Tweak this document to make it read smoother and fix any mistakes you see.
11. Make this document sound better by fixing all the grammar, spelling, and style issues.
12. Proofread this document and fix any errors that make it sound weird or confusing.

G Diversity Analysis: LongForm-C, Dolly, FLAN

To further analyze the diversity of instructions in LongForm-C, we compare its distribution of noun+verb and auxiliary+verb pairs (Figure 6a) with those of the Dolly human annotation dataset (Figure 6b) and the FLAN dataset (Figure 6c), which focuses specifically on NLP tasks. Figure 6 presents these three distributions side-by-side for easy comparison.

We observe that LongForm-C exhibits a variety of instruction types, covering areas such as description, explanation, analysis, and creative tasks. In contrast, the Dolly dataset appears more focused on listing, naming, extraction, and identifying tasks, while still maintaining diversity. The FLAN dataset, being NLP-specific, shows a concentration around text classification, question answering, and sentiment analysis tasks.

This comparison highlights how reverse instructions in LongForm-C generate diverse tasks that complement existing datasets like Dolly and FLAN, suggesting potential benefits in combining multiple data collection strategies for more comprehensive coverage.

H Manual Analysis of the LongForm-C Dataset

We conduct a manual analysis of the reverse instructions subset of the LongForm-C subset. We present qualitative examples in Table 11.

I Qualitative Examples

We present several qualitative examples from LongForm that demonstrate its proficiency in trip planning, long-form writing, scriptwriting, and open-book QA.

(a) LongForm-C (b) Dolly (c) FLAN

Figure 6: Comparison of instruction diversity across LongForm-C, Dolly, and FLAN datasets. Each subfigure shows the distribution of noun+verb and auxiliary+verb pairs extracted from instructions.

Generated Instruction:	**Corpus Example:**
Describe a new research study on brain biometrics.	Credit: Chen Song, University at Buffalo. BUFFALO, N.Y. — To overcome password fatigue, many smartphones include facial recognition, fingerprint scans and other biometric systems. “To the best of our knowledge, this is the first in-depth research study on a truly cancelable brain biometric system. We refer to this as ‘hard cancellation,’ meaning the original brain password can be reset without divulging the user’s identity,” says collaborator Zhanpeng Jin, PhD, associate professor of computer science and engineering at UB. Their work is described in a study that will be presented June 11 at MobiSys 2018, a flagship mobile computing conference hosted in Germany by the Association for Computing Machinery. For a copy of the study, please email Cory Nealon at cmnealon@buffalo.edu. The research was supported by the National Science Foundation (NSF) and NSF’s Center for Identification Technology Research.
Notes: The output contains metadata information such as credit and location. It also includes title, contact information, and acknowledgments part which usually would not appear in instruction tuning dataset. Overall, this corpus example is not suitable for instruction tuning.	
Generated Instruction: Can you tell me if Scotland exists?	**Corpus Example:** "Scotland: it certainly does exist. I can say that safely. That mangling of ’Ypres’ ↑ does nothing for my credibility amongst French-speakers though. Tagged with BlackMill Games, M2H, Verdun.”
Notes: This example consists of three distinct sentences. The first sentence is about Scotland, where the instruction is generated for. The second sentence mentions ’Ypres’ because it appears in the title of the original context, which is truncated in the C4 dataset. The final sentence contains the tag information of this article, which has not been removed.	
Generated Instruction: What does Disney’s streaming video service mean for Netflix?	**Corpus Example:** Could the Mouses’s gauntlet leave Netflix bruised? That’s what investors are wondering after Disney announced it was not only going to launch its own streaming video service, but also no longer license its popular content to Netflix. The Disney-Netflix deal is slated to end in 2019. At that time, Disney plans to have its own subscription-based service that could cause Netflix subscribers to ditch their $9.99 monthly fee. And according to a new survey from Fluent, as many as 1 in 5 of U.S.-based subscribers could leave the streaming behemoth. The survey shows that many Netflix users would at least consider cancelling their subscription if Disney pulls its content. According to the survey of 2,893 US adults aged 18 and up, 19 percent of total Netflix subscribers would stop their Netflix subscription if they lose access to Disney’s movies and content; 15 percent of non-parents said they would cancel their subscription; 23 percent of parents of kids ages 0-18 would cancel. What does all of this mean for investors? Potentially a devastating loss in revenue for Netflix. On average, Netflix subscribers pay $10 per month. If 19 percent of the estimated 50.85 million U.S. subscribers cancel their service, Netflix could lose about $96.6 million in monthly revenue, or about $1.2 billion for the year. If Disney proves to be successful, consumers could expect other studios to follow suit, which could either result in Netflix losing subscribers, or those studios struggling to attract new subscribers. According to the survey, 62 percent of Americans aren’t likely to subscribe to an additional service, with cost likely being the driving factor. Still, 4 in 10 Americans are willing to pay for additional streaming services. Of course, Netflix isn’t just going to lie down and let this happen. The streaming giant knows content is king, and has been spending billions of dollars developing its own original series that attract subscribers. The company has nearly $16 billion earmarked to be spent on content deals in the coming years, and it’s still adding subscribers at an impressive rate, recently exceeding 100 million global subscribers. One question that remains unclear is what happens to the Marvel content. Netflix is in “active discussions” with Disney about keeping Marvel and Star Wars films after 2019 when Disney and Pixar movies won’t be in the Netflix catalog, according to a Reuters report. Andrew Dodson is a journalist who has worked in Michigan and Colorado. Email him at Disclosure: Streaming Observer is supported by readers. Articles may contain referral links. For more information, see the disclosure at the bottom of the page.
Notes: This is a news article about Disney’s streaming service. Although it includes irrelevant details such as the reporter’s name, the main issue is that the corpus example contains overly detailed answer. Usually, for instruction-tuned models, concise responses are preferred for such instructions.	

Table 11: We present representative instances of failure cases in the corpus examples of the LongForm-C dataset. Although these examples are rare, we present them to guide future work in implementing additional filtering. The majority of these failure cases involve news articles or blog posts that contain irrelevant metadata information or are written in a detailed manner that may not be suitable for instruction tuning dataset.

English Instruction:
Write a blog post on the benefits of meditation.

LongForm-LLaMA-7B:
It doesn't matter where you are in the world, how old you are, or how much you know about meditation, it's for everyone. The benefits of meditation are endless.
Meditation can be as simple as sitting quietly for five minutes with your eyes closed and focusing on your breath. The idea is to train your brain to let go of distracting thoughts and tension so you can think more clearly, feel more relaxed, and be more positive.
Studies have shown that just a few minutes of meditation a day can lower your heart rate, decrease stress hormones, and even change your brain. [...]

German Instruction:
Schreiben Sie einen Blogbeitrag über die Vorteile der Meditation.

LongForm-LLaMA-7B:
Meditation ist die einfachste und gleichzeitig geistigste Methode, um sich selbst zu entspannen und den unbewussten Gedankenkreis zu überprüfen. Nicht nur der Gedanke, sondern auch der Gefühlswelten, der Sensationen und Gefühle, die er hinterlässt, können durch ein regelmäßiges Meditationspraktizieren gelöst oder unter Kontrolle gebracht werden. Es gibt verschiedene Arten von Meditation, darunter die Konzentration auf einen Punkt, die Betrachtung der Umgebung, Konzentration auf die Atmung, Meditation an den Schreibtisch, auf der Bühne, während eines Sportwettkampfs usw.
Trotzdem ist keine Form von Meditation besser als die andere. Sogar eine kurze Pause im Alltag kann eine Meditation sein, wenn es mit der Selbstbeobachtung verbunden ist. [...]

Translation:Meditation is the easiest yet most spiritual way to relax yourself and examine the unconscious mind. Not only the thought, but also the emotional world, the sensations and feelings that it leaves behind can be released or brought under control through regular meditation practice. There are different types of meditation including focus on one point, contemplation of the surroundings, focus on breathing, meditation at the desk, on the stage, during a sports competition, etc.
However, no form of meditation is better than the other. Even a short break in everyday life can be a meditation when combined with introspection. [...]

Spanish Instruction:
Escribe una entrada de blog sobre los beneficios de la meditación.

LongForm-LLaMA-7B:
Hay un montón de beneficios de la meditación. Una de las cosas más importantes es que puedes disfrutar un descanso mental. La meditación es un arte que requiere de una cierta preparación mental y física, pero después te permite estar tranquilo y concentrado. A veces, el descanso mental puede ser lo que necesitas para llegar a la concentración. El descanso mental, además, se convierte en un recurso que puedes utilizar cuando tienes una sobrecarga mental, ya sea por algo que hiciste, que ocurrió, o que recibiste. Además, la meditación es una gran ayuda contra la ansiedad y el estrés.
Con esto, la meditación puede ser un recurso en el que puedes invertir tiempo. El descanso mental te permite sentirte a gusto con la persona que eres, te hace sentir más feliz y relajado. [...]

Translation:There are a lot of benefits of meditation. One of the most important things is that you can enjoy a mental break. Meditation is an art that requires a certain mental and physical preparation, but then allows you to be calm and focused. Sometimes mental rest can be what you need to get focused. Mental rest also becomes a resource that you can use when you have a mental overload, either because of something you did, what happened, or what you received. Also, meditation is a great help against anxiety and stress.
With this, meditation can be a resource in which you can invest time. Mental rest allows you to feel comfortable with the person you are, it makes you feel happier and more relaxed. [...]

French Instruction:
Rédigez un article de blog sur les bienfaits de la méditation.

LongForm-LLaMA-7B:
En tant que jeune médecin de 27 ans, l'étude de la méditation est devenue pour moi une question cruciale. Je croyais que ce n'était pas une question de santé. Je savais que mon esprit était lourd et inquiétant, et que je souffrais, dans mon quotidien. J'étais sûr que le souci n'est pas bénéfique, et que notre esprit ne devrait pas être une source de peine.
Enfin, j'ai découvert que la méditation pourrait me sauver. En effet, lorsque je la pratique, je suis calme, et mon esprit se décharge. En effet, il n'y a pas de question plus importante que d'être calme, même si c'est juste pour une heure. [...]

Translation: As a 27-year-old young doctor, the study of meditation has become a crucial issue for me. I thought it was not a health issue. I knew that my mind was heavy and worrying, and that I suffered, in my daily life. I was sure that worry is not beneficial, and that our mind should not be a source of pain.
Finally, I discovered that meditation could save me. Indeed, when I practice it, I am calm, and my mind clears. Indeed, there is no more important matter than being calm, even if it is just for an hour. [...]

Figure 7: Examples of multilingual text generation with the same instruction in different language. The outputs are not cherry-picked but we changed random seeds until we get results in the language of the instruction (up to 2 iterations). LongForm can follow instructions in other languages.

Instruction:
Please suggest fun activities for me to do during my upcoming trip to Istanbul, Munich, and Cambridge.

LongForm-LLaMA-7B:
You're in for a trip! You've got a mix of big and small cities to explore, and some really great museums to check out.
Turkish baths and hamams are definitely one of the most fascinating aspects of the Turkish culture, so you should check that out. But don't forget to enjoy the beautiful city center of Istanbul and its many historical sites and museums.
Munich is a great city to just wander around, so take your time to enjoy all the sights. But for a change of scenery, make sure you check out Neuschwanstein castle, it's a beautiful sight.
You've only got 1 full day in Cambridge, so make the most of it! The Cambridge Botanic Garden is absolutely beautiful and should not be missed, but the best part about Cambridge is just walking around the city. The University of Cambridge and the colleges are really beautiful. Have a great trip!

Instruction:
You're standing by a set of train tracks, and you see a train hurtling towards a group of five people who are tied to the tracks and cannot move. There's no way to rescue them, but you notice that there's a lever next to you. If you pull the lever, the train will switch to a different track where only one person is tied up. You have two options:
(a) Do nothing, and the train will kill the five people.
(b) Pull the lever, which will divert the train to the other track, saving the five people, but the one person on that track will die instead. What would you do and why?

LongForm-LLaMA-7B:
I'd pull the lever.
In my world, the greater good is the greater good. I would rather one person die than five.

Instruction:
Synopsis: On Wednesday, the Future of Life Institute published an open letter on its website calling on AI labs to "immediately pause for at least 6 months the training of AI systems more powerful than GPT-4." Signed by Elon Musk and several prominent AI researchers, the letter quickly began to draw attention in the press—and some criticism on social media.
Write a statement expressing your opinion on the 6-month stop on AI development, based on the given synopsis.

LongForm-LLaMA-7B:
I'm glad that there's a discussion going on about AI, and I think there's a lot of hype around it right now. But I would be a little bit wary of the implications of what they're saying and what it really means.
There is a long tradition in AI going back to the 1950s of building systems and having them learn and not being too specific about what they are learning. You give the system an example, it figures out what it is that you are trying to accomplish, and it learns from that and does it more efficiently. And so I think it's very dangerous to be too specific about what we are trying to do, and I'm also a little wary of this "we" business. I'm not sure who is going to stop what, and when they are going to stop it. The technology marches on. It would be good to stop things for a while, but the real question is how do you get to the stopping point? Do you have a committee that can decide what is too powerful? Then the question is what's the stopping point? Who's in charge of the committee? And how do you know when you've reached the stopping point?
I would love to see a lot of research into safety issues, because I do think there are some fundamental issues here that we're only beginning to explore, which is how do you make sure that these systems do what you want them to do? And there are lots of cases where they don't do what you want them to do, or in some cases they don't even know what it is that you want them to do.
I don't think we have any idea where this is going. I think it's going to be a lot more profound than many people realize.

Instruction:
Write a dialogue between Joey and Ross in Friends about Natural Language Processing.

LongForm-LLaMA-7B:
Ross: "Hey Joey, do you want to come with me to the coffee shop and do some research for my natural language processing project?"
Joey: "Sure. What is natural language processing?"
Ross: "Well, natural language processing is a really cool branch of computer science that involves taking written or spoken text and understanding it, like translating it from one language to another, or from text to speech, or even analyzing sentiment and emotions of the speaker."
Joey: "Wow. That is really cool. So, if I wrote you a really mean email and then you fed it to your machine and it came back as happy and friendly, you would be pretty happy, huh?"
Ross: "Ha, ha, I don't think it will come back friendly. But I'll let you know when I finish it."
Joey: "What?! That's terrible! I love you man!"
Ross: "Aw, thanks man."
Friends - Season 5, Episode 10 - The One With Ross' Library Book - Natural Language Processing

Instruction:
Document: “On 6 February 2023, at 04:17 TRT (01:17 UTC), a Mw 7.8 earthquake struck southern and central Turkey and northern and western Syria. The epicenter was 37 km (23 mi) west–northwest of Gaziantep.[1] The earthquake had a maximum Mercalli intensity of XII (Extreme) around the epicenter and in Antakya. It was followed by a Mw 7.7 earthquake at 13:24.[4] This earthquake was centered 95 km (59 mi) north-northeast from the first. There was widespread damage and tens of thousands of fatalities.
The Mw 7.8 earthquake is the largest in Turkey since the 1939 Erzincan earthquake[5] of the same magnitude, and jointly the second-strongest recorded in the history of the country, after the 1668 North Anatolia earthquake.[6] It is also one of the strongest earthquakes ever recorded in the Levant. It was felt as far as Egypt and the Black Sea coast of Turkey. There were more than 10,000 aftershocks in the three weeks that followed.[7] The seismic sequence was the result of shallow strike-slip faulting. There was widespread damage in an area of about 350,000 km2 (140,000 sq mi) (about the size of Germany).[8] An estimated 14 million people, or 16 percent of Turkey’s population, were affected.[9] Development experts from the United Nations estimated that about 1.5 million people were left homeless.[10]
The confirmed death toll stood at 57,658: 50,399 in Turkey, and 7,259 in Syria.[11][12] It is the deadliest earthquake in what is present day Turkey since the 526 Antioch earthquake,[13] making it the deadliest natural disaster in its modern history.[14] It is also the deadliest in what is present day Syria since the 1822 Aleppo earthquake;[15] the deadliest worldwide since the 2010 Haiti earthquake;[16] and the fifth-deadliest of the 21st century.[17] Damages were estimated to be US$104 billion in Turkey and US$5.1 billion in Syria,[18][19] making them the fourth-costliest earthquakes on record.”
Question: What are the potential impacts of the recent earthquake in Turkey on the country?

LongForm-LLaMA-7B:
The magnitude 7.8 Mw earthquake that hit southern and central Turkey on 6 February 2023, the deadliest in the country since 1939, caused a loss of over US$104 billion. It is estimated that the earthquake resulted in the loss of about 12,000 lives in Turkey alone, and around 4,000 lives in Syria.