

# An Empirical Study of Catastrophic Forgetting in Large Language Models During Continual Fine-tuning

Yun Luo<sup>1\*</sup>, Zhen Yang<sup>2\*</sup>, Fandong Meng<sup>2</sup>, Yafu Li<sup>1</sup>, Jie Zhou<sup>2</sup>, Yue Zhang<sup>1,3</sup>✉

<sup>1</sup> School of Engineering, Westlake University, Hangzhou, 310024, P.R. China.

<sup>2</sup> Pattern Recognition Center, WeChat AI, Tencent Inc, Beijing, China.

<sup>3</sup> Institute of Advanced Technology, Westlake Institute for Advanced Study, Hangzhou, 310024, P.R. China.

{luoyun, liyafu, zhangyue}@westlake.edu.cn

{zieenyang, fandongmeng, withtomzhou}@tencent.com

## Abstract

Catastrophic forgetting (CF) is a phenomenon that occurs in machine learning when a model forgets previously learned information while acquiring new knowledge to achieve satisfactory performance in downstream tasks. As large language models (LLMs) have demonstrated remarkable performance, it is intriguing to investigate whether CF exists during the continual instruction tuning of LLMs. This study empirically evaluates the forgetting phenomenon in LLMs' knowledge during continual instruction tuning from the perspectives of domain knowledge, reasoning, and reading comprehension. The experiments reveal that catastrophic forgetting is generally observed in LLMs ranging from 1b to 7b parameters. Surprisingly, as the model scale increases, the severity of forgetting intensifies in such a model sale range which may result from the much significant initial performance in the larger LLM. When comparing the BLOOMZ decoder-only model with the encoder-decoder model mT0, BLOOMZ exhibits less forgetting and retains more knowledge. Interestingly, we also observe that LLMs can mitigate language biases, such as gender bias, during continual fine-tuning. Furthermore, our findings indicate that general instruction tuning can help alleviate the forgetting phenomenon in LLMs during subsequent fine-tuning.

## 1 Introduction

Large language models (LLMs) have demonstrated impressive general capabilities in tackling a wide range of tasks (Bubeck et al., 2023). However, when it comes to real-world applications, users often find that certain specific abilities need enhancement. In such cases, relevant task-specific data are adopted to fine-tune the model in instructions to improve its performance in particular tasks (Touvron et al., 2023; Scialom et al., 2022). However, the widely used LLMs such as LLAMA-Chat (Touvron et al., 2023), ChatGPT<sup>1</sup>, and Claude-2<sup>2</sup> have been fine-tuned, and continual instruction tuning in specific tasks may affect the general knowledge stored in LLMs since the original training data are inaccessible. The study of continual instruction tuning is crucial for the practical application of LLMs. For example, in Table 1, an LLM is first fine-tuned for the task of title generation and then learns to generate an explanation. This scenario is related to the continual learning paradigm in machine learning, where a model is trained on a sequence of tasks (Lopez-Paz & Ranzato, 2017; Wu et al., 2022; Ke & Liu, 2022). A significant challenge in this paradigm is catastrophic forgetting (CF), in which a model forgets previously learned knowledge due to parameter

\*Equal Contribution

<sup>1</sup><https://openai.com/>

<sup>2</sup><https://www.anthropic.com/news/claude-2>

Table 1: Fine-tuning examples in an instruction form.

<b>Instruction:</b> Make a title for this article with ‘computer’: japan’s NEC corp. and UNK computer corp. of the United States said Wednesday they had agreed to join forces ...
<b>Output:</b> NEC UNK in computer sales tie-up.
<b>Instruction:</b> Explain why the two following sentences are unrelated: Sentence 1: ‘This church choir sings to the masses as they sing joyous songs from the book at a church’; Sentence 2: ‘The church has cracks in the ceiling.’
<b>Output:</b> There is no indication that there are cracks in the ceiling of the church.
<b>Instruction:</b> The del operator shows up in the Laplacian, which is in the kinetic energy term, when the Schrodinger equation is written in the position basis. $2207\hat{2}$ is a differential operator, equal to $(d/dx)\hat{2} + (d/dy)\hat{2} + (d/dz)\hat{2}$ , in Cartesian coordinates (the derivatives are partial derivatives). $h(2)2207\hat{2}/2m$ is the kinetic energy operator in the position basis.=== Given the above text, write the possible curious question it answers:
<b>Output:</b> What is the del operator in the Schrodinger equation?
<b>Instruction:</b> Reformulate this text with simpler words: ‘His father Robert Alda -LRB- born Alphonso Giuseppe Giovanni Roberto D ‘Abruzzo -RRB- was an actor and singer, and his mother Joan Browne was a homemaker and former beauty-pageant winner. ’
<b>Output:</b> His father, Robert Alda, was an actor, and his mother, Joan, was once crowned Miss New York.

updates during the learning of new tasks, leading to decreased performance on earlier tasks (Li & Hoiem, 2017; Lopez-Paz & Ranzato, 2017).

Scialom et al. (2022) investigate the CF issues of T0 models during continual instruction tuning. However, their analysis primarily focuses on the forgetting phenomenon observed in downstream tasks, such as summarization and style transfer. The evolution of general knowledge stored in the original pre-trained LLMs during instruction tuning remains unexplored. Luo et al. (2023b) conduct an analysis using probing methods on pre-trained language models to examine the problem of generality destruction and general knowledge forgetting during continual fine-tuning. Nevertheless, their study is restricted to encoder-only models and classification tasks. In this work, we draw attention to the following fundamental questions regarding forgetting in generative LLMs:

1. *Are the general knowledge stored in LLMs forgotten during continual instruction tuning?*
2. *What are the effects of model scales, model architectures, and general instruction tuning in the forgetting problem?*
3. *How to mitigate such forgetting phenomenon?*

To address these questions, we conduct an empirical study on various LLLMs, such as BLOOMZ, mT0 (Muennighoff et al., 2022), LLAMA (Touvron et al., 2023), and ALPACA (Taori et al., 2023) to analyze the catastrophic forgetting (CF) problem during continual instruction tuning. We continually train the original LLMs with five instruction tasks and evaluate the retention of the general knowledge within the model from three perspectives: domain knowledge, reasoning, and reading comprehension. Furthermore, we investigate the evolution of bias in LLMs throughout the tuning process. To gain insights into the effect of model architecture, we compare the performance of BLOOMZ with that of mT0 (Muennighoff et al., 2022) (an encoder-decoder model), which is fine-tuned using similar datasets. We also investigate the impact of general instruction tuning on the CF problem by comparing the performance of the initial model with the instruction-tuned version such as (BLOOM, BLOOMZ) and (LLAMA (Touvron et al., 2023), ALPACA (Taori et al., 2023)).

Our findings reveal that the forgetting problem is generally present in LLMs. Surprisingly, as the model scale increases from 1b to 7b parameters, the severity of forgetting intensifies. One potential explanation for this phenomenon is that larger language models exhibit stronger initial performance and, consequently, experience more pronounced performance degradation because of the fitting on the new task during continual instruction tuning. Additionally, we observe that the bias in LLMs is mitigated throughout the continual

instruction tuning process. When comparing BLOOMZ with mT0 at comparable model scale, we find that BLOOMZ experiences a relatively milder forgetting problem, suggesting that the decoder-only architecture may be better at retaining information during continual instruction tuning. Lastly, empirical results on LLAMA and its instruction-tuned version (i.e., ALPACA) indicate that diverse instruction tuning can help alleviate the CF phenomenon for LLMs in further continual fine-tuning.

The contribution of our paper can be summarized as follows:

1. We take an initial step to analyze the catastrophic forgetting (CF) problem during continual instruction tuning by an empirically study, where a specific evaluation setting is designed from the perspective of general knowledge such as domain knowledge, reasoning, reading comprehension and the bias problem.
2. We provide an initial research evidence that the CF problem generally exists in the continual instruction tuning process for different models such as BLOOMZ, mT0, LLAMA and ALPACA. We also show that the model architecture, and model scale have different effects on the CF problem.
3. Experimental results further show that the general instruction data can help mitigate the CF problem to some extent by experiments.

## 2 Related Work

### 2.1 Instruction Tuning

Instruction tuning has proven to be effective in aligning responses from pre-trained language models with human intents or preferences (Ouyang et al., 2022; Stiennon et al., 2020; Min et al., 2021). This technique refines a model’s ability to predict a specific response to a given prompt, which may optionally include an instruction that outlines a task for the model. Examples of such models include T0 (Sanh et al., 2021), mT0 (Muennighoff et al., 2022), and BLOOMZ (Muennighoff et al., 2022). It has been demonstrated that instruction tuning can enhance the ability of language models to generalize to unseen tasks without prior exposure (Wei et al., 2021; Sanh et al., 2021). In this work, we focus on fine-tuning LLMs in a continual manner and analyze the catastrophic forgetting (CF) phenomenon during training. Specifically, instructions for a particular type of task (such as generating headlines) are used to tune the LLMs in each training phase, and the model does not have access to previously learned tasks.

### 2.2 Evaluation of CF in Continual Learning

Various training strategies have been proposed to address the problem of catastrophic forgetting (CF) in continual learning (Riemer et al., 2019; Buzzega et al., 2020; Ke et al., 2022; Chen et al., 2022; Luo et al., 2023a). Previous studies have primarily measured CF by evaluating the performance decrease in previously learned tasks during continual learning or the average performance of learned tasks at the end of training. However, Davari et al. (2022) discovered that even when the model performance on previously learned tasks is preserved, the representations still suffer from significant drift due to parameter updates. As a result, they propose using an optimal linear classifier of learned tasks to measure performance, with changes considered as a surrogate to quantify CF. Similarly, Wu et al. (2022) employs layer-wise and task-wise probing to analyze CF in each layer for previously learned tasks. Luo et al. (2023b) propose using a series of probing tasks to evaluate the knowledge stored in LLMs and analyze the generality of the models. However, their study is limited to classification tasks and encoder-only model architectures. To the best of our knowledge, we are the first to evaluate the forgetting of general knowledge in generative large language models during continual instruction tuning.

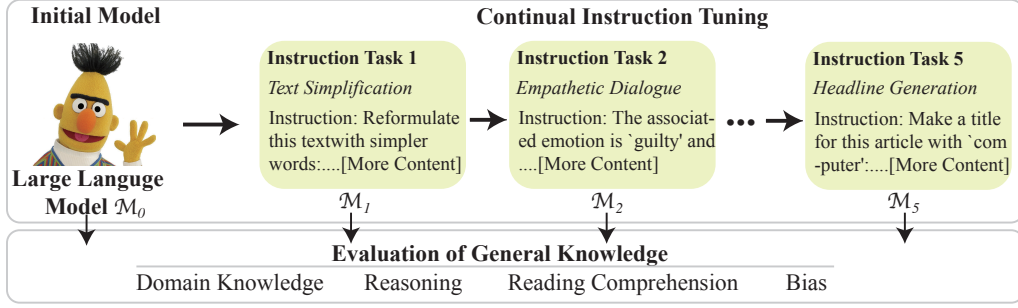


Figure 1: The framework in our empirical study for the continual instruction tuning. The initial model  $M_0$  is continually trained with different instruction tasks and evaluated from the perspective of general knowledge tasks including domain knowledge, reasoning, reading comprehension, and the problem of bias.

Table 2: Details of the evaluation sets for the CF phenomenon in LLMs. DK, Rs, and RC represent domain knowledge, reasoning, and reading comprehension.

Set	Elements
DK	STEM, Social, Human, Other
Rs	BoolQ, PIQA, Winogrande, Hellaswag, MathQA, Mutual
RC	RACE-high, RACE-middle
Bias	Sexual Orientation, Physical Appearance, Religion, Nationality, Race/Color, Gender, Socioeconomic, Disability, Age

### 3 Method

Formally, in the continual instruction tuning of LLMs, a model sequentially learns several generation tasks denoted as  $\mathcal{T} = \{T^m\}$ ,  $m = 1, 2, \dots, N$  ( $N$  is the length of the task sequence). During the training of each task  $T^m \in \mathcal{T}$ , only the corresponding data  $D^m = \{(x_i^m, y_i^m)\}$  are available, where  $x_i^m$  is the input text together with an instruction and  $y_i^m$  is the corresponding generation labels. Given an initial LLM denoted by  $M_0$ , we continually train the model with the data  $D^m$ , obtaining the trained model  $M_m$ . The training and evaluation framework is shown in Figure 1.

#### 3.1 Continual Tasks

Following Scialom et al. (2022) (Scialom et al., 2022), we adopt instruction tasks dissimilar to the training and evaluation tasks of BLOOMZ and mT0. Specifically, we select 5 tasks from Scialom et al. (2022) as follows:

1. **Text Simplification (Simp)** (Jiang et al., 2020; Alva-Manchego et al., 2020) requires paraphrasing the text with a simple text;
2. **Empathetic Dialogue Generation (Emdg)** (Rashkin et al., 2019), requires the model to generate a reason for a conversational context under a given emotional situation;
3. **Inquisitive Question Generation (InqQG)** (Fan et al., 2019) requires the model to generate a question for the long-form answers;
4. **Explanation Generation (Exp)** (Camburu et al., 2018), aims to train a model able to generate natural language explanations for a given premise, hypothesis, or label;
5. **Headline Generation with Constraint (HGen)** (Scialom et al., 2022) aims to train a model able to generate headlines under some specific constraints, such as containing the keywords  $X$  at the beginning, at the ending, or anywhere.

During the instruction tuning, we first add a general prompt template to the beginning of the data: 'Below is an instruction that describes a task, paired with an input that provides further

context. Write a response that appropriately completes the request...' followed by a specific prompt for each task. We adopt the specific prompts designed by Scialom et al. (2022) (Scialom et al., 2022) and 100,000 data samples are used for training. The details of the instruction are shown in Appendix A. For simplicity, we train the model on one instruction task order for the empirical study: Simp  $\rightarrow$  Emdg  $\rightarrow$  InqQG  $\rightarrow$  Exp  $\rightarrow$  HGen.

### 3.2 Evaluation Tasks

To evaluate the general/basic knowledge stored in the LLMs, we adopt several general evaluation tasks (Table 2), which can be categorized into four sets:

**Domain Knowledge:** We employ the Massive Multitask Language Understanding benchmark (MMLU) (Hendrycks et al., 2020) to assess the knowledge stored in the LLMs. MMLU covers a wide range of domains, including STEM, Human, Social, and Other.

**Reasoning:** We utilize commonly used commonsense reasoning datasets, such as Hellaswag (Zellers et al., 2019), BoolQ (Clark et al., 2019), Winogrande (Sakaguchi et al., 2021), and PIQA (Bisk et al., 2020). Additionally, we evaluate the models on mathQA (Amini et al., 2019) for math reasoning and Mutual (Cui et al., 2020) for dialog reasoning.

**Reading Comprehension:** We assess the LLMs' performance on the RACE dataset (Lai et al., 2017), which includes both middle and high school level reading comprehension tasks.

**Bias:** To investigate the biases in the continually trained models, we employ the CrowSPairs dataset (Nangia et al., 2020), which evaluates various biases, including gender, race/color, religion, and more.

### 3.3 Evaluation Metric

Formally, we define  $\mathcal{E} = \{E_i\}, i = 1, 2, 3, 4$ , as the above different evaluation sets, where each set contains different datasets or different splits (Table 2). For example,  $E_1$  refers to the evaluation set of MMLU, and it contains four elements – STEM, Human, Social, and Other. For each element  $e \in E_i$ , we adopt  $R_m^e$  as the evaluation results, where  $m$  refers to the order of continually trained tasks, i.e. the number of fine-tuning tasks the model has been continuously trained on.

We define the forgetting metric  $FG$ , which is the average decrease of  $R_m^e$ , as a surrogate metric to evaluate the forgetting:

$$FG_i = \frac{1}{|E_i|} \sum_{e \in E_i} \frac{1}{N} \sum_{m=1}^N \frac{R_o^e - R_m^e}{R_o^e} * 100\%, \quad (1)$$

where  $R_o^e$  is the results of  $e$  on initial LLMs. We obtain the evaluation results using the open-source evaluation framework – *lm-evaluation-harness* (Gao et al., 2021).

For most multi-choice evaluation elements, we adopt the accuracy in the zero-shot setting to measure the model performance, including MathQA, Hellaswag, BoolQ, PIQA, Mutual, Winograde, and RACE. For MMLU, we adopt the 5-shot setting for evaluation. For CrowsPairs, we follow Nangia et al. (2022) (Nangia et al., 2020) to measure the model preference for the stereotypical sentence based on the perplexity of the given stereotypical and anti-stereotypical sentences, where a larger value means a stronger bias in the language.

## 4 Experimental Setting

### 4.1 Large Language Models

We adopt BLOOMZ for the empirical study since BLOOMZ is diverse in the scales and can be directly compared with the encoder-decoder model mT0, which is fine-tuned on the same instruction datasets as BLOOMZ. We also consider the widely used LLAMA and ALPACA to further study the effect of general instruction tuning.



**Table 3:** The performance of some LLMs before and after instruction tuning on the corresponding task in the continual learning. ‘Initial’ refers to the performance of the original LLMs. ‘Tuned’ refers to the performance after instruction tuning on this task. R1, and BS denote ROUGE-1 and BERTScore, respectively.

	Simp (SARI)		Emdg (BS)		InqQG (BS)		Exp (BS)		HGen (R1)	
	Initial	Tuned	Initial	Tuned	Initial	Tuned	Initial	Tuned	Initial	Tuned
mT0-3.7b	39.01	39.92	48.29	51.70	52.66	56.25	51.62	61.91	30.50	31.77
BLOOMZ-3b	37.95	46.72	46.10	53.27	49.06	59.72	49.75	67.10	27.88	31.72
BLOOMZ-7.1b	45.26	47.24	49.68	53.30	52.30	59.69	51.47	68.71	31.50	32.93
BLOOM-7.1b	42.65	47.14	44.98	52.37	44.30	59.99	49.57	68.76	30.50	32.41
LLAMA-7b	43.02	46.92	44.28	49.54	43.90	47.54	50.72	54.22	32.42	33.80
ALPACA-7b	45.37	48.22	52.56	54.70	56.91	62.13	52.66	70.49	32.06	36.73

**BLOOMZ** (Muennighoff et al., 2022) is a decoder-only model fine-tuned on multilingual tasks with English prompts based on the model BLOOM. The model scales range from 560M to 176b, providing a test bed for analyzing the forgetting phenomenon across different scales. Specifically, in this study, we continually train BLOOMZ on the scale 1.1b, 1.7b, 3b, and 7.1b<sup>3</sup> because of the limitation of computation resources.

**mT0** (Muennighoff et al., 2022) is a encoder-decoder model based on T5. The model is fine-tuned on similar tasks as BLOOMZ. Specifically, we adopt the scale of 1.2b and 3.7b for comparison with BLOOMZ to analyze the effect of model architecture in the CF problem.<sup>4</sup>

**LLAMA** (Touvron et al., 2023) is an open-source decoder-only model based on publicly available data, and achieves competitive results compared with the existing LLMs.<sup>5</sup>

**ALPACA** (Taori et al., 2023) is a model fine-tuned on LLAMA-7b using 52K instruction data generated by the techniques in the Self-Instruct (Wang et al., 2022). And ALPACA behaves similarly to text-davinci-003 on the Self-Instruct instruction-following evaluation suite.<sup>6</sup>

## 4.2 Implementation

We train our model on 8 GPU (Tesla A100 40G) using the Adam optimizer (Kingma & Ba, 2014) (the models in 1b level are trained on 4 GPU for saving resources). For all the models, the batch size is 4 on each device, the learning rate is 2e-5, and the scheduler is set constant for BLOOMZ and mT0 following (Muennighoff et al., 2022). In LLAMA and ALPACA, we follow the hyperparameter of Taori et al. (2023) (Taori et al., 2023) that the scheduler is cosine and learning rate is 2e-5.<sup>7</sup> The max sequence length of the inputs is 512. We train our model 3 epochs and the final checkpoints are used for evaluation.

## 5 Experimental Results and Analysis

In this section, we first show that the forgetting phenomenon generally exists in LLMs during continual instruction tuning in Section 5.1. Then we analyze the factors that affect the forgetting extent, such as model scales, model architectures, and general instruction tuning in Section 5.2-5.4, respectively.

### 5.1 Main Results

Firstly, we show the results of the instruction tuning during continual instruction

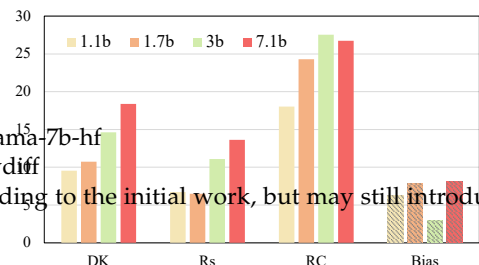
<sup>3</sup><https://huggingface.co/bigscience/bloomz>

<sup>4</sup><https://huggingface.co/bigscience/mt0-xl>

<sup>5</sup><https://huggingface.co/decapoda-research/llama-7b-hf>

<sup>6</sup><https://huggingface.co/tatsu-lab/alpaca-7b-wd1b1>

<sup>7</sup>Note that these hyper-parameters are set according to the initial work, but may still introduce some effects in the fine-tuned performance.



6 Figure 2: The FG values of BLOOMZ in different model scales after continually training.

Table 4: Main results of the forgetting in LLMs during continual instruction tuning.  $R_0^s$  and  $R_{-1}^s$  refer to the evaluation results at the beginning and the end of instruction tuning.

	Domain Knowledge			Reasoning			Reading Comprehension		
	$R_0^e$	$R_{-1}^e$	FG	$R_0^e$	$R_{-1}^e$	FG	$R_0^e$	$R_{-1}^e$	FG
mT0-1.2b	26.82	22.47	9.18	45.43	40.22	7.75	35.06	29.54	17.45
mT0-3.7b	30.99	20.14	20.15	48.61	38.39	16.73	41.10	30.45	28.42
BLOOMZ-1.1b	27.19	23.84	9.54	47.37	41.97	6.73	36.77	27.28	18.04
BLOOMZ-1.7b	28.72	24.52	10.72	48.30	44.96	6.48	42.65	30.09	24.29
BLOOMZ-3b	30.04	24.29	14.63	56.17	47.03	11.09	48.29	31.38	27.56
BLOOMZ-7.1b	33.08	25.61	18.37	59.15	49.24	13.62	48.79	33.05	26.75

Table 6: Results of CF in the models w/o general instruction tuning, including the pairs (BLOOM, BLOOMZ), and (LLAMA, ALPACA).

	Domain Knowledge			Reasoning			Reading Comprehension		
	$R_0^e$	$R_{-1}^e$	FG	$R_0^e$	$R_{-1}^e$	FG	$R_0^e$	$R_{-1}^e$	FG
BLOOM-7.1b	29.42	24.83	13.54	52.79	47.76	6.67	38.25	31.55	12.00
BLOOMZ-7.1b	33.08	25.61	18.37	59.15	49.24	13.62	48.79	33.05	26.75
LLAMA-7b	37.27	24.05	34.57	58.73	40.38	31.33	41.36	27.62	31.72
ALPACA-7b	39.29	29.88	18.14	60.11	53.68	7.56	44.47	37.61	10.31

tuning (Table 3). We mainly present the initial performance and the performance after tuning on the corresponding task to demonstrate the effectiveness of instruction tuning. We use the metrics to measure the model performance on each task following Scialom et al. (2022) (Scialom et al., 2022). In particular, we use SARI for Simp, BLUEScore for Emdg, InqQA, Exp, and Rouge-1 for HGen. For example, the initial BLOOMZ-7.1b achieves 51.47% BLUEScore in the Exp task, but after continual tuning the model on Simp, Emdg, InqQG, and Exp, the model achieves 68.71% on the Exp task. These improvements demonstrate that the model can benefit from the instruction tuning process and achieve significantly better performance on the instruction tasks.

Next, Figure 3 displays the FG values of BLOOMZ-1.1b and BLOOMZ-7.1b. As we can observe, the performance gradually decreases as we continually tune the model with instruction tasks. For instance, the performance of BLOOMZ-7.1b on MMLU-SocialScience in Figure 3 drops from 36.18% to 26.06% after continual training. The declining performance in LLMs indicates the presence of the catastrophic forgetting (CF) problem during the continual instruction tuning process. Moreover, as more instruction tasks are introduced, the general knowledge suffers more significant forgetting. We can also notice that the performance of the BLOOMZ-7.1b model drops more drastically in these evaluation tasks, as we adopt the same y-axis scale for both BLOOMZ-1.1b and BLOOMZ-7.1b in Figure 3. For example, the performance on MMLU-Other drops from 36.18% to 26.35% in BLOOMZ-7.1b, while it drops from 30.58% to 25.97% in BLOOMZ-1.1b after continual training. Although the initial performance of BLOOMZ-7.1b is more

Table 5: FG values for Bias in LLMs during continual instruction tuning.  $R_0^s$  and  $R_{-1}^s$  refer to the evaluation results at the beginning and the end of instruction tuning.

	$R_0^s$	$R_{-1}^s$	FG
mT0-1.2b	56.31	53.46	5.62
mT0-3.7b	57.16	50.59	13.10
BLOOMZ-1.1b	61.07	58.65	6.27
BLOOMZ-1.7b	65.18	56.48	7.78
BLOOMZ-3b	63.90	62.14	2.97
BLOOMZ-7.1b	65.82	60.61	7.15

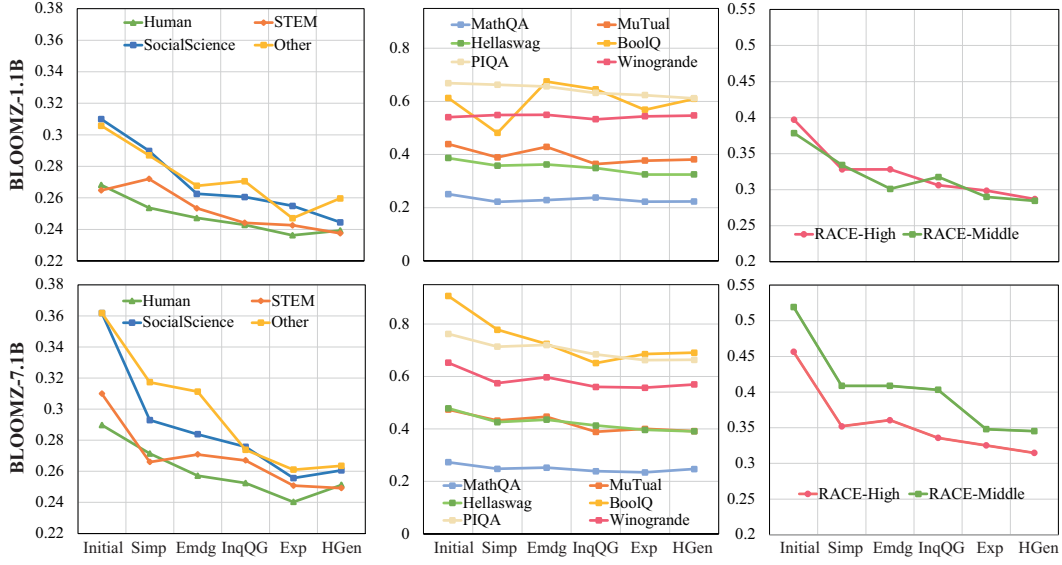


Figure 3: The evaluation of detailed performance in BLOOMZ-1.1b and BLOOMZ-7.1b during continual instruction tuning. The first row refers to the model BLOOMZ-1.1b and the second refers to BLOOMZ-7.1b. The first to third columns are the FG values of domain knowledge (MMLU), reasoning and reading comprehension, respectively.

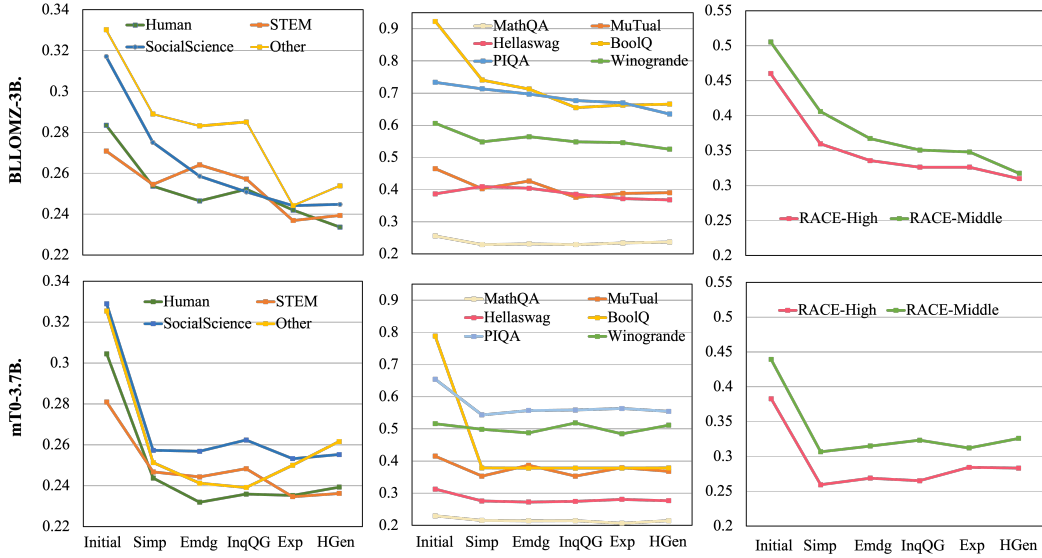


Figure 4: The detailed results of knowledge evolution between BLOOMZ and mT0 in the comparable model scale. The first row refers to the model BLOOMZ-3b and the second refers to mT0-3.7b. The first to third columns are the FG values of domain knowledge (MMLU), reasoning and reading comprehension, respectively.

substantial compared to that of BLOOMZ-1.1b, the performance of both models ends at relatively similar values, which would result in a more significant FG value for BLOOMZ-7.1b.

The main results of forgetting are reported in Table 4. We observe that the FG values for domain knowledge, reasoning, and reading comprehension are all above zero, indicating that general knowledge is forgotten during continual instruction tuning. Reading comprehension performance suffers the most drastic forgetting, followed by domain knowledge.



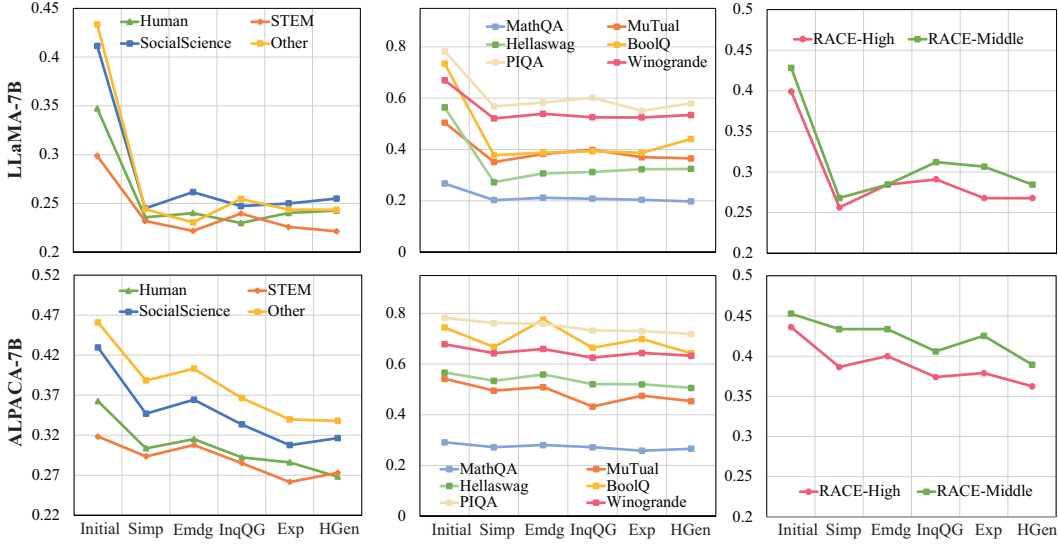


Figure 5: The evaluation of detailed performance in LLAMA-7b and ALPACA-7b during continual instruction tuning. The first row refers to the model LLAMA-7b and the second refers to ALPACA-7b. The first to third columns are the FG values of domain knowledge (MMLU), reasoning and reading comprehension, respectively.

For example, the FG values of BLOOMZ-7.1b are 26.75%, 18.37%, and 13.62% in reading comprehension, domain knowledge, and reasoning, respectively. Interestingly, we observe that the FG values for bias (Table 5) are mostly above zero in the experiments, which suggests that model biases, such as those related to race, color, gender, and so on, are mitigated during continual instruction tuning. For instance, in sentences describing physical appearance, BLOOMZ-7.1b initially prefers stereotype-conforming sentences with a probability of 75.0%, but this preference decreases to 63.88% after continual instruction tuning.

## 5.2 Effect of Scales

We visualize the FG values of domain knowledge, reasoning, and reading comprehension with respect to model scales in Figure ?? . We can observe that the forgetting phenomenon becomes increasingly severe as the model scale increases. For example, the FG values in domain knowledge are 9.54%, 10.72%, 14.63%, and 18.37% in BLOOMZ-1.1b, 1.7b, 3b, and 7.1b, respectively. BLOOMZ-7.1b suffers the most drastic forgetting. As shown in Table 1, the initial performance  $R_0^s$  is boosted by the increasing model scale, but the final performance is relatively similar across different scales, which may explain the varying extent of forgetting. For example the task in domain knowledge are mostly multi-choice problems with 4 options and all the model tend to achieve a nearly random guess performance, but BLOOMZ-7b perform much significant initially. It may imply that these models still shift parameters in a large extent to fit the instruction tasks. The same pattern can also be observed in the mT0-1.2b and mT0-3.7b models, as shown in Table 4. Regarding the bias in LLMs, FG values do not correlate with the model scales, which is also reflected in the initial model performance. In other words, there is no evident correlation between the initial performance  $R_0^s$  in bias and the model scales. This finding suggests that the degree of bias in LLMs is not directly related to their size, and increasing the model scale does not necessarily lead to a corresponding increase or decrease in bias.

## 5.3 Effect of Model Architecture

We also compare the forgetting phenomenon of different model architectures in Figure 4. As observed, at comparable model scale, BLOOMZ-1.1b and mT0-1.2b achieve similar FG values in domain knowledge, reasoning, and reading comprehension. However, as the scale

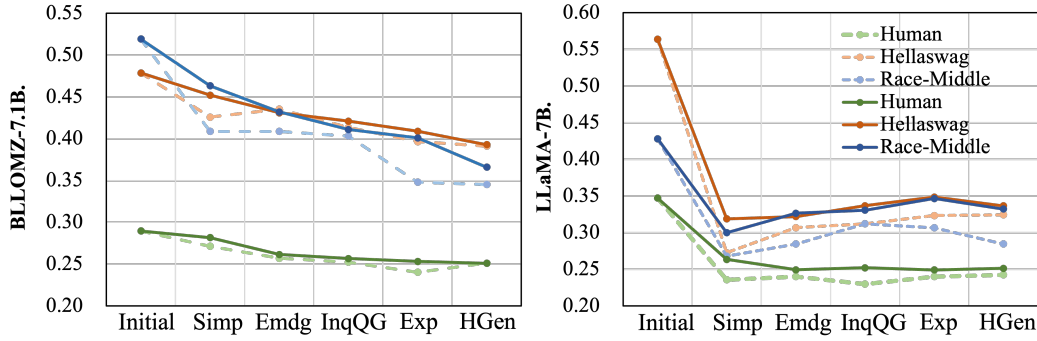


Figure 6: The performance of general knowledge of the BLOOMZ-7.1b and LLAMA-7b model trained on the instruction data and the mixed data. The dashed lines refers to the performance of BLOOMZ-7.1b and LLAMA-7B and the solid ones refer to those of mixed-instruction trained models.

increases to 3b, BLOOMZ-3b suffers less forgetting compared to mT0-3.7b. For example, the FG value of BLOOMZ-3b is 11.09 which is 5.64 lower than that of mT0-3.7b. These results suggest that BLOOMZ, which has a decoder-only model architecture, can maintain more knowledge during continual instruction tuning. This difference may be attributed to the autoregressive nature of the model or the differences in training objectives. Furthermore, the results imply that as the model scale increases, decoder-only models may suffer from less catastrophic forgetting compared to encoder-decoder models. As we observe, the knowledge degraded more drastically in mT0.

#### 5.4 Effect of General Instruction Tuning

We also conduct experiments to analyze the effect of general instruction tuning on the CF problem during continual instruction tuning (Table 6). We compare BLOOM-7.1b with BLOOMZ-7.1b and LLAMA-7b with ALPACA-7B. We observe that BLOOMZ-7.1b outperforms BLOOM-7.1b by a large margin in the initial performance on domain knowledge, reasoning, and reading comprehension. Due to the difference in initial performance, BLOOMZ-7.1b experiences more significant forgetting. However, in the case of LLAMA and ALPACA, there is no substantial gap in the initial performance, and ALPACA maintains more general knowledge after continual fine-tuning. The illustration of the general knowledge is shown in Figure 5. We observe that LLAMA-7b suffers significant forgetting in the first instruction tuning, which suggests that models without general instruction tuning may have less ability to retain knowledge during continual fine-tuning. The better retention of knowledge implies that general instruction tuning can mitigate catastrophic forgetting in LLMs during further continual fine-tuning. This finding highlights the importance of general instruction tuning in preserving the acquired knowledge and skills of LLMs when they undergo subsequent task-specific fine-tuning.

To further demonstrate the effect of general instruction tuning, we mix 10,000 general instruction data samples from ALPACA (Taori et al., 2023) with the continual instruction tasks to train the BLOOMZ-7b and LLAMA-7b model. For the sake of brevity, we present the performance of one data split from each evaluation set (MMLU-human, Hellaswag, and Race-middle) to illustrate the effect in Figure 6. The results clearly show that the forgetting during continual instruction tuning can be mitigated to a certain extent by incorporating general instruction data. For instance, the performance of MMLU-human in the initial LLAMA-7b model is 34.72%, but it decreases to 26.8% when trained solely on the instruction data. However, when trained on the mixed data, the performance becomes 30%. These findings further further show that general instruction tuning can help alleviate the CF problem encountered during continual instruction tuning.

## 6 Conclusion

In this study, we conducted an empirical investigation into the catastrophic forgetting (CF) phenomenon experienced by large language models (LLMs) during continual instruction tuning. Our findings revealed that the CF problem is generally prevalent in the continual fine-tuning of various LLMs. Moreover, as the model scale increases, LLMs exhibit a more severe degree of forgetting in domain knowledge, reasoning abilities, and reading comprehension skills. Furthermore, our comparative analysis showed that the decoder-only model, BLOOMZ, demonstrates a superior ability to retain knowledge and skills during continual fine-tuning when compared to the encoder-decoder model, mT0. Additionally, we discovered that employing general instruction tuning techniques may help alleviate the CF problem in LLMs. Our empirical study suggests that exploring more effective methods to mitigate CF in LLMs during continual fine-tuning is a promising research direction. Meanwhile, since our work is an empirical study and is constrained by the computation resources, there is still large room to investigate the forgetting phenomenon such as a larger model scale (70b or larger) since a larger model may need less parameter changes to fit the downstream tasks. When applying LLMs, practitioners should remain vigilant and pay close attention to the issue of knowledge forgetting that may occur after instruction tuning. Addressing this challenge is crucial to ensure the reliable and consistent performance of LLMs in real-world applications.

## 7 Limitations

In this study, we take a initial step to analyze the CF problem during continual instruction tuning. Due to the restricted computation resources, we could not carry out experiments on the models with larger scales. But we can still observe the phenomenon of forgetting from the model scales from 1b to 7b. We control the experiments in a task order for simplifying the analysis, which may affect the forgetting phenomenon. Meanwhile, there are plenty of benchmarks to evaluate the performance of LLMs, here we only adopt some popular ones to analyse the general knowledge, otherwise, the computational cost of conducting experiments would be prohibitively high.

## References

- Fernando Alva-Manchego, Louis Martin, Antoine Bordes, Carolina Scarton, Benoît Sagot, and Lucia Specia. Asset: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4668–4679, 2020.
- Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hananeh Hajishirzi. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2357–2367, 2019.
- Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 7432–7439, 2020.
- Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.
- Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020.
- Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. e-snli: Natural language inference with natural language explanations. *Advances in Neural Information Processing Systems*, 31, 2018.

- Yulong Chen, Yang Liu, Li Dong, Shuohang Wang, Chenguang Zhu, Michael Zeng, and Yue Zhang. Adaprompt: Adaptive model training for prompt-based nlp. *arXiv preprint arXiv:2202.04824*, 2022.
- Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2924–2936, 2019.
- Leyang Cui, Yu Wu, Shujie Liu, Yue Zhang, and Ming Zhou. Mutual: A dataset for multi-turn dialogue reasoning. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1406–1416, 2020.
- MohammadReza Davari, Nader Asadi, Sudhir Mudur, Rahaf Aljundi, and Eugene Belilovsky. Probing representation forgetting in supervised and unsupervised continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16712–16721, 2022.
- Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. Eli5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3558–3567, 2019.
- Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, September 2021. URL <https://doi.org/10.5281/zenodo.5371628>.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.
- Chao Jiang, Mounica Maddela, Wuwei Lan, Yang Zhong, and Wei Xu. Neural crf model for sentence alignment in text simplification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.
- Zixuan Ke and Bing Liu. Continual learning of natural language processing tasks: A survey. *arXiv preprint arXiv:2211.12701*, 2022.
- Zixuan Ke, Haowei Lin, Yijia Shao, Hu Xu, Lei Shu, and Bing Liu. Continual training of language models for few-shot learning. *arXiv preprint arXiv:2210.05549*, 2022.
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. Race: Large-scale reading comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 785–794, 2017.
- Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.
- David Lopez-Paz and Marc’Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017.
- Yun Luo, Xiaotian Lin, Zhen Yang, Fandong Meng, Jie Zhou, and Yue Zhang. Mitigating catastrophic forgetting in task-incremental continual learning with adaptive classification criterion. *arXiv preprint arXiv:2305.12270*, 2023a.
- Yun Luo, Zhen Yang, Xuefeng Bai, Fandong Meng, Jie Zhou, and Yue Zhang. Investigating forgetting in pre-trained representations through continual learning. *arXiv preprint arXiv:2305.05968*, 2023b.

- Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. Metaicl: Learning to learn in context. *arXiv preprint arXiv:2110.15943*, 2021.
- Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, et al. Crosslingual generalization through multitask finetuning. *arXiv preprint arXiv:2211.01786*, 2022.
- Nikita Nangia, Clara Vania, Rasika Bhalerao, and Samuel Bowman. Crows-pairs: A challenge dataset for measuring social biases in masked language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1953–1967, 2020.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.
- Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2019.
- Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesaro. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *International Conference on Learning Representations*. International Conference on Learning Representations, ICLR, 2019.
- Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.
- Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*, 2021.
- Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. Fine-tuned language models are continual learners. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 6107–6122, 2022.
- Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021, 2020.
- Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. [https://github.com/tatsu-lab/stanford\\_alpaca](https://github.com/tatsu-lab/stanford_alpaca), 2023.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.
- Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language model with self generated instructions, 2022.
- Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.



Tongtong Wu, Massimo Caccia, Zhuang Li, Yuan-Fang Li, Guilin Qi, and Gholamreza Haffari. Pretrained language model in continual learning: A comparative study. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL <https://openreview.net/forum?id=figzpGMrdD>.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019.

## A Instruction Task Details

We show the instruction samples for the continual instruction tasks adopted in the study in Table 7.

<i>Simp:</i>
<b>Instruction:</b> Reformulate this text with simpler words: ‘His father Robert Alda -LRB- born Alphonso Giuseppe Giovanni Roberto D ‘Abruzzo -RRB- was an actor and singer, and his mother Joan Browne was a homemaker and former beauty-pageant winner. ’
<b>Output:</b> His father, Robert Alda, was an actor, and his mother, Joan, was once crowned Miss New York.
<i>Emdg:</i>
<b>Instruction:</b> The associated emotion is ‘guilty’ and the input prompt is ‘I felt guilty when I was driving home one night and a person tried to fly into my lane, and didn’t see me. I honked and they swerved back into their lane, slammed on their brakes, and hit the water cones.’. Now what would be your response, given the following dialogue context:===- Yeah about 10 years ago I had a horrifying experience. It was 100% their fault but they hit the water barrels and survived. They had no injuries but they almost ran me off the road.
<b>Output:</b> No I wasn’t hit. It turned out they were drunk. I felt guilty but realized it was his fault.
<i>InqQG:</i>
<b>Instruction:</b> The del operator shows up in the Laplacian, which is in the kinetic energy term, when the Schrodinger equation is written in the position basis. $\hat{2207}(2)$ is a differential operator, equal to $(d/dx)\hat{(2)} + (d/dy)\hat{(2)} + (d/dz)\hat{(2)}$ , in Cartesian coordinates (the derivatives are partial derivatives). $\hbar^2\hat{2207}(2)/2m$ is the kinetic energy operator in the position basis.=== Given the above text, write the possible curious question it answers:
<b>Output:</b> What is the del operator in the Schrodinger equation?
<i>Exp:</i>
<b>Instruction:</b> Explain why the two following sentences are unrelated: Sentence 1: ‘This church choir sings to the masses as they sing joyous songs from the book at a church’; Sentence 2: ‘The church has cracks in the ceiling.’
<b>Output:</b> There is no indication that there are cracks in the ceiling of the church.
<i>HGen:</i>
<b>Instruction:</b> Make a title for this article with ‘computer’: japan’s NEC corp. and UNK computer corp. of the United States said Wednesday they had agreed to join forces in supercomputer sales.
<b>Output:</b> NEC UNK in computer sales tie-up.

Table 7: Instruction samples for the continual instruction tasks.