StoryWars: A Dataset and Instruction Tuning Baselines for Collaborative Story Understanding and Generation

Yulun Du and **Lydia Chilton**
Columbia University
New York City, New York, USA
{yulundu, chilton}@cs.columbia.edu

Abstract

Collaborative stories, which are texts created through the collaborative efforts of multiple authors with different writing styles and intentions, pose unique challenges for NLP models. Understanding and generating such stories remains an underexplored area due to the lack of open-domain corpora. To address this, we introduce StoryWars, a new dataset of over 40,000 collaborative stories written by 9,400 different authors from an online platform. We design 12 task types, comprising 7 understanding and 5 generation task types, on StoryWars, deriving 101 diverse story-related tasks in total as a multi-task benchmark covering all fully-supervised, few-shot, and zero-shot scenarios. Furthermore, we present our instruction-tuned model, InstructStory, for the story tasks showing that instruction tuning, in addition to achieving superior results in zero-shot and few-shot scenarios, can also obtain the best performance on the fully-supervised tasks in StoryWars, establishing strong multi-task benchmark performances on StoryWars.[1]

1 Introduction

Storytelling is crucial due to its vital role in human experience, history, and culture dating back to the earliest days of humanity. Humans possess the unique storytelling ability to structure a sequence of events, whether factual, fictional or a mixture of both, and create a coherent narrative that conveys a big picture while also including intricate details. Current story generation systems usually mimic this ability by starting with a plot then crafting the story. This can be done by linearly expanding (Peng et al., 2018, Yao et al., 2019, Martin et al., 2017) or hierarchically developing (Xu et al., 2018, Fan et al., 2018, Fan et al., 2019, Rashkin et al. 2020, Goldfarb-Tarrant et al., 2020) the story based on the given plot. Collaborative storytelling is distinctly challenging because there is no predetermined plot or story outline of events. Instead, collaborative stories are created through the collective efforts of multiple authors. Each author contributes a section sequentially, while also attempting to express their own personal intentions within the context of the jointly crafted and jointly owned story. It is a more challenging problem as it requires not only the ability to generate text, but also the capability to understand the previous context and contributions written by other authors.

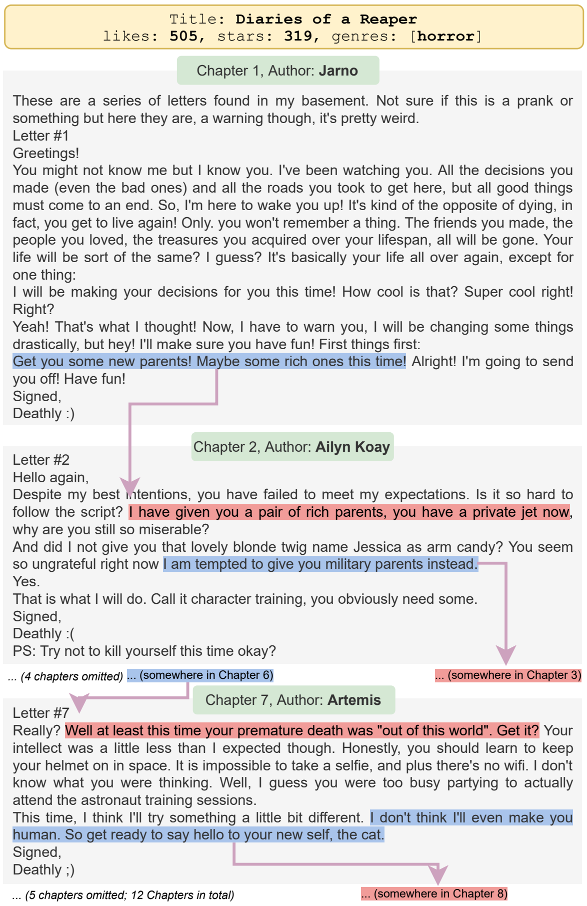

Figure 1: An example story with 12 turns in the StoryWars dataset. In each turn, the author leaves a "floor" for the next author to continue collaboratively.

[1] We make our data, code, and models publicly available at https://github.com/ylndu/storywars

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 3044–3062
July 9-14, 2023 ©2023 Association for Computational Linguistics

Large Language Models (LLMs) (Devlin et al. 2019, Liu et al., 2019, Yang et al. 2019, Raffel et al. 2019, Brown et al. 2020, Zhang et al. 2022, Chowdhery et al. 2022, Touvron et al. 2023) have demonstrated exceptional performance on various understanding and generation benchmarks, indicating their potential in addressing natural language processing (NLP) challenges related to collaborative storytelling. This prompts an intriguing question within the research community: *How could LLMs synergize both their understanding and generation capabilities via multitask learning to address the challenges of collaborative storytelling?*

We present STORYWARS, a dataset of over 40,000 stories gathered from an online collaborative storytelling platform[2]. Figure 1 shows an example story in the STORYWARS dataset. Each story contains rich information including its title, genres given by the initial author, chapters written by different authors, and human ratings including stars and likes. Each chapter was written by exactly one author and the previous author might leave a *collaborative floor* (Coates, 1997) for the next author to continue. Therefore, for a model to generate a continuing chapter, it needs to understand the preceding context, including the title, genres, and the writing styles and intentions of previous authors conveyed in the collaborative floor.

Due to the multitask nature of collaborative storytelling and the rich information of the STORYWARS, we design 12 task types, including both understanding and generation task types, as a multitask benchmark for an initial probe of collaborative storytelling. We follow the task definition from FLAN (Wei et al., 2021), where each task type contains multiple tasks. In the end, our benchmark contains 101 tasks in total, split such that it covers all fully-supervised, few-shot, and zero-shot learning application scenarios. It is important to note that prevailing multitask NLP benchmarks are either focusing on understanding (e.g. Wang et al., 2018, Wang et al., 2019) or generation (e.g. Gehrmann et al., 2021, Khashabi et al., 2021, Liu et al., 2021) alone, or only a subset of the learning scenarios. To our knowledge, we are the first to propose a story benchmark that contains both understanding and generation in all three scenarios.

Large language models have been shown to not only be fully-supervised, few-shot, and zero-shot learners but also multitask ones. Instruction Tuning (Wei et al., 2021, Sanh et al., 2022, Chung et al., 2022) has been the state-of-the-art approach for zero-shot and few-shot scenarios. However, it has not yet been applied in the fully-supervised setting. We evaluated Instruction Tuning on the benchmark and we found that in addition to achieving state-of-the-art results in zero-shot and few-shot scenarios, when combined with single-task fine-tuning, Instruction Tuning can surpass single-task fine-tuning alone, resulting in a consistent performance boost of 1.53 points on average for all tasks.

Our contributions are as follows:

- We introduce a novel collaborative story dataset STORYWARS that comprises 40k stories written by 9.4k different authors, with rich information such as genres and human ratings, to promote research in the field of collaborative storytelling.
- We propose a new benchmark based on STORYWARS that consists of 7 understanding and 5 generation task types, totaling in 101 tasks for testing the fundamental abilities of LLMs to model collaborative stories. The benchmark covers the fully-supervised, few-shot, and zero-shot scenarios.
- We present INSTRUCTSTORY, a instruction-tuned model that demonstrates strong performance on the STORYWARS benchmark in all three learning scenarios. In addition, we show for the first time that we could extend Instruction Tuning with a single-task finetuning stage to achieve superior performance and obtain robust performance boost.

2 Related Work

2.1 Story Datasets

The most popular story datasets that have been widely used by many story generation systems in the past are ROCStories (Mostafazadeh et al., 2016) and WritingPrompts (Fan et al., 2018). ROCStories comprises five-sentence commonsense short stories, and WritingPrompts includes 300k open-domain prompt-story pairs, neither of which are collaboratively written. On the other hand, Storium (Akoury et al., 2020) and roleplayerguild (Louis and Sutton, 2018), are collaborative and written by multiple authors in turns, but in a game setting. The key distinction of our STORYWARS dataset is that the stories are both collaborative and open-domain. For a comparison of these datasets, refer to Table 1.

[2] www.storywars.net Unfortunately, the website has closed down by the time of writing this paper. Some stories could be recovered from https://archive.md/sAOOq

Dataset	# Stories	# Words per story	Genres	Human Ratings	Open-Domain	Multi-Turn Collab.	User-Gen
ROCStories	98,156	88	✘	✘	✔	✘	✘
WritingPrompts	303,358	735	✘	✘	✔	✘	✔
roleplayerguild	1,439	3,079	✘	✘	✘	✔	✔
Storium	5,743	19,278	✘	✘	✘	✔	✔
STORYWARS	40,135	367	✔	✔	✔	✔	✔

Table 1: Comparison of our STORYWARS dataset with previous story datasets.

2.2 Multitask NLP Benchmarks

Existing multitask NLP benchmarks tends to focus on evaluating either understanding (Wang et al., 2018, Wang et al., 2019) or generation (Gehrmann et al., 2021, Khashabi et al., 2021, Liu et al., 2021) capabilities of NLP models. There are task-specific benchmarks that address both, such as those for dialog (Mehri et al., 2020) and code (Lu et al., 2021). For the task of storytelling, the LOT benchmark (Guan et al., 2022) focuses on both aspects but is limited to Chinese and has fewer tasks than our proposed STORYWARS dataset. BIG-bench (Srivastava et al., 2022), which includes 204 tasks for understanding and generation, only tests zero-shot and few-shot abilities without finetuning. STORYWARS provides a benchmark for story understanding and generation with 101 tasks spanning all zero-shot, few-shot, and full-supervised scenarios for various applications.

2.3 Multitask NLP and Instruction Tuning

Current multitask LLMs mainly follow two approaches. The first approach involves finetuning, such as with ExT5 (Aribandi et al., 2022) and Muppet (Aghajanyan et al., 2021), where the model is made more generalized through multitask finetuning and then fine-tuned again on downstream tasks. The second approach focuses solely on zero-shot and few-shot performance, with the goal of bridging the gap between finetuning and these performance levels, as seen in FLAN (Wei et al., 2021), T0(Sanh et al., 2022), FLAN-T5 (Chung et al., 2022), and ZeroPrompt (Xu et al., 2022). These models often utilize Instruction Tuning or similar frameworks. In this paper, we extend Instruction Tuning's capabilities to achieve superior performance in the full-supervised scenario as well.

3 Methodology

3.1 The STORYWARS Dataset

We obtained the STORYWARS dataset from storywars.net, an online collaborative storytelling platform where users can pitch ideas and create stories. However, once an initial chapter is published, the story becomes part of the Story Wars community and can be contributed to by other users. For a continuing chapter to be officially recognized, it must be voted in by other users, resulting in a high quality of stories on the platform.

We scraped and parsed the stories on Story Wars, ending up in obtaining 76k stories. We then used FastText (Bojanowski et al., 2017) language identification to filter for English stories and further cleaned the dataset by removing noisy stories based on GPT-2 perplexity (Radford et al., 2019). We also removed stories that are shorter than 30 words or stories with chapters that are shorter than 10 words. To further ensure the quality of the dataset, we also remove stories that have very low human ratings, such as likes and stars.

In consideration of ethical issues, we employed OpenAI Content Moderation APIs[3] and the Detoxify[4] toxicity classifier to identify and remove potentially harmful content, such as toxicity, obscenity/sexual content, threats, insults, identity hate, and self-harm posts from the dataset. Furthermore, to safeguard user privacy, we replaced all URLs, email addresses, and phone numbers with special tokens <URL>, <EMAIL>, and <PHONE>.

After thorough data cleaning, we obtained a final dataset of 40,135 stories written by 9,494 authors. Due to the fact that the long tail of genres is very noisy, we made the simplifying assumption that each story contains a single dominant genre, if any. Each story in the dataset was structured with sev-

[3]https://beta.openai.com/docs/api-reference/moderations
[4]https://github.com/unitaryai/detoxify

eral key elements, including a title, a genre (which could be empty), the numbers of likes and stars received, the authors and the corresponding chapters.

We denote an arbitrary story in the dataset as $s \in S$, where $S = \{(p, (c_i, a_i)_{i=0}^{t}, g, r_l, r_s)\}$. That is, each story s_i is denoted by a 5-tuple of a title p, chapter-author pairs (c_i, a_i) of t turns, a genre g, a likes rating r_l, and a stars rating r_s.

3.2 The Multitask Benchmark

3.2.1 Story Understanding Tasks

Genre Classification Understanding the genre of a story is essential for collaborative storytelling models to comprehend the context. The genre classification task involves identifying the genre of a story. This task can be formulated as a binary text classification problem, where given a story, the task is to predict whether it belongs to a specific genre g. This can be represented as $g = f(c_1, c_2, ..., c_t)$.

Authorship Attribution Identifying the author of a text is a crucial step in understanding the writing style of an individual. Authorship attribution, traditionally, is the task of determining the author of a given text. In this paper, we formulate the task of authorship attribution as identifying the author of a specific chapter, represented as $a = f(c)$.

Authorship Verification Authorship Verification, in contrast to author attribution, is the task of determining whether two texts have been written by the same author by comparing their writing styles. The task is represented as $y = f(c_i, c_j)$, where y is a binary variable.

Connectivity Inference Understanding the chapter shifts in long-range stories can be a beneficial ability for collaborative storytelling. Following Sun et al. (2022), we also include the connectivity inference task, where the goal is to determine whether two given chapters are consecutive in a story. The task is represented as $y = f(c_n, c_m)$.

Temporal Inference Inspired from the Connectivity Inference task, we also aim to evaluate a model's ability to understand the temporal relationships between chapters in a story. The Temporal Inference task involves determining whether two chapters in the same story are in the correct chronological order. For example, (c_i, c_{i+1}) and (c_i, c_{i+5}) would be considered positive instances, while (c_{i+5}, c_i) would not. The task is represented as $y = f(c_n, c_m)$, where y is a binary variable.

Story Scoring Understanding human ratings of a story is crucial for generating texts that align with human preferences. Many dialog-related applications rely on human labelers to rate texts based on different criteria, e.g. LAMDA (Thoppilan et al., 2022). Since StoryWars contains human ratings in the form of likes and stars, we propose to include a regression task for story scoring as a task type. We follow Raffel et al. (2019) and normalize the story ratings to a range from 0-10, with rounded scores to the nearest increment of 0.1, and convert the float to string. Given a rating score, such as r_l, the task is represented as $r_l = f(c_1, c_2, ..., c_t)$.

Story Segmentation Although stories are already divided into chapters, it is still possible to evaluate models' ability to identify chapter boundaries within a story, where one chapter concludes and another begins, in order to encourage the model to capture discourse-level information. We design the task of story segmentation as $c_1, b_1, c_2, b_2, ..., b_{t-1}, c_t = f(s)$, where b_i is the boundary between two chapters.

3.2.2 Story Generation Tasks

Next Chapter Generation The next chapter generation problem is defined as an generation task that takes previous chapters and genre information as input, and then generates the subsequent chapter. This is represented as $c_{k+1} = f(c_1, c_2, ..., c_k, g)$.

Conditional Story Generation The conditional story generation problem is defined as an generation task that also takes previous chapters and genre information as input, but then generates the entire continuation of the story until the conclusion instead. It further evaluates an NLP model's capability to plan and organize the story. This is represented as $c_{k+1}, c_{k+2}, ..., c_t = f(c_1, c_2, ..., c_k, g)$.

Chapter Infilling In line with Ippolito et al. (2019), the chapter infilling task evaluates an NLP model's ability to generate an intermediate chapter given the context of a preceding and subsequent chapter. This is represented as $c_k = f(c_{k-1}, c_{k+1})$.

Global Infilling Building on the chapter infilling task, the global infilling problem considers more extensive context information, including both preceding and subsequent chapters. This is represented as $c_k = f(c_1, c_2, ..., c_{k-1}, c_{k+1}, ..., c_t)$.

Temporal Ordering Following Lin et al. (2021), we also include a task that unscrambles chapter sequences based on temporal information, except that we simplify the problem by eliminating the requirement for the NLP model to infill masked chapters. This is represented as $c_1, c_2, ..., c_t = f(permute(c_1, c_2, ..., c_t))$.

Task Type	#Tasks	Train	Dev	Test
Fully-supervised				
Genre Classification	27	2,000	250	250
Author Attribution	30	2,000	250	250
Author Verification	1	144,000	20,925	20,925
Connectivity Inference	1	59,402	7,521	6,963
Temporal Inference	1	84,632	9,480	8,928
Story Scoring	2	17,046	1,485	1,484
Story Segmentation	1	17,256	1,500	1,500
Next Chapter Generation	1	40,729	5,845	5,043
Conditional Story Generation	1	23,473	4,345	3,543
Chapter Infilling	1	23,473	4,345	3,543
Global Infilling	1	23,473	4,345	3,543
Temporal Ordering	1	78,554	8,932	8,407
Few-shot				
Genre Classification	10	32	32	200
Zero-shot				
Genre Classification	23	0	0	200

Table 2: Task statistics for the STORYWARS benchmark.

3.2.3 The Benchmark

Benchmark task statistics The 12 task types translate into 101 tasks based on STORYWARS, with 96 understanding tasks and 5 generation tasks. It is worth noting that the majority of the understanding tasks are genre classification tasks (60) and author attribution tasks (30). Out of the 60 genre classification tasks, we split them into 27 fully-supervised, 10 few-shot, and 23 zero-shot datasets, according to the genre frequency so that the split closely aligns with realistic data distribution. For the fully-supervised and few-shot tasks, we divided the data into training, dev, and test sets. For the zero-shot tasks, we used all the data as a test set by sampling. The remaining task types were used for fully-supervised scenarios. It is important to mention that all of the data in the fully-supervised, few-shot, and zero-shot scenarios are disjoint to prevent data leakage. The overall task data statistics can be found in the Table 2.

Evaluation metrics For the genre classification, author attribution, author verification, temporal inference, and connectivity inference tasks, we use F-1 score as the evaluation metric, due to the imbalance nature of the task data. For the story scoring tasks, in line with Raffel et al. (2019) for regression tasks, we use Spearman correlation coefficients as the evaluation metric, because it measures monotonic relationships. For the story segmentation task, we use Boundary Similarity (Fournier, 2013) as the evaluation metric. For the generation tasks, following the suggestions introduced in Chhun et al. (2022), Qin et al. (2019), and Gangal et al. (2021), we use BERTScore (Zhang* et al., 2020) as the evaluation metric, as it has been shown by Chhun et al. (2022) to have better correlation with human evaluation at both the story-level and system-level for story generation systems than other automatic metrics including frequently-used BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004). Also, Gangal et al. (2021) points out that in the narrative reordering problem, similar to our temporal ordering task, BERTScore also correlates quite well with human evaluations. We recognize that there is currently no widely accepted or reliable automatic evaluation metric in the field of story generation, and the use of automatic evaluation in this field is often criticized. However, for the purpose of fast and fair comparison, we chose to follow previous work and use the current best available metric, even though we acknowledge that it may not be perfect.

For evaluating the model performance, we calculate the macro-average of the performance on all tasks within each task type, this allows us to compare models across different task types. The metrics for understanding, generation, and overall performance are determined by the macro-average of the scores across the corresponding task types.

3.3 The INSTRUCTSTORY Framework

The main goal of instruction tuning is to evaluate the performance of unseen tasks in zero-shot and few-shot learning scenarios, and to show that it can improve the gap between zero-shot and fully-supervised learning performances. Additionally, we are interested in how instruction tuning can improve the performance of fully-supervised tasks.

To accomplish our goal, we propose a two-stage training approach called INSTRUCTSTORY. In the first stage, we use instruction tuning as a form of *pre-finetuning* Aghajanyan et al. (2021). During this stage, we use instructions instead of task prefixes proposed in Muppet Aghajanyan et al. (2021) to enhance the model's ability to generalize to new instructions. In the second stage, after instruction tuning with the fully-supervised task mix, we use single-task finetuning to continually train the model for each fully-supervised task. We use T5-large-lm-adapt (770m) as the base model for instruction tuning INSTRUCTSTORY and all of the training tasks are from the STORYWARS fully-supervised training split. Figure 2 illustrates the overall INSTRUCTSTORY framework. The instructions we used are included in Appendix A.1.

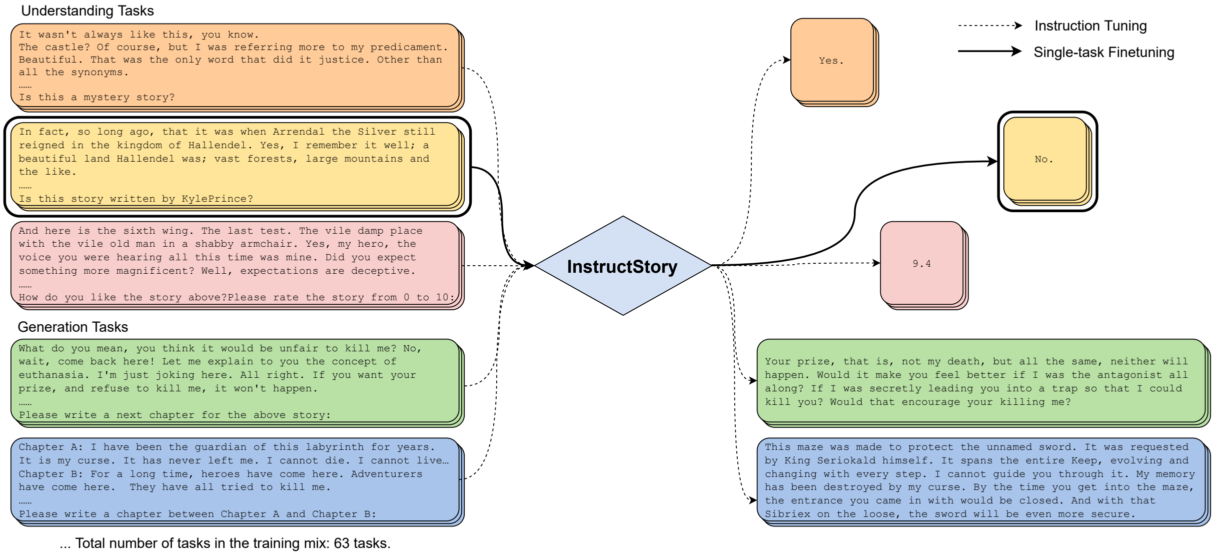

Figure 2: INSTRUCTSTORY undergoes a two-stage training process. In stage 1 (- - →), we instruction tune the model on 63 story tasks to improve generalization to unseen zero-shot and few-shot tasks. In stage 2 (→), we perform single-task finetuning on each fully-supervised task to optimize performance on specific tasks.

4 Experimental Results

4.1 Baselines

We include several strong baseline models with a comparable number of parameters. For understanding tasks, we include **BERT-large** (345m), **RoBERTa-large** (354m), and **DeBERTa-v2-xlarge** (900m) as baselines. For generation tasks, we include **GPT2-medium** (345m), **GPT2-large** (774m), and **OPT-350m** as baselines. These models all have comparable or near-comparable numbers of parameters. To demonstrate the effectiveness of our method, we also include **T5-large-lm-adapt** (770m) as a baseline model in the overall comparison. In addition, for the few-shot and zero-shot scenarios, we include the state-of-the-art instruction tuning model **FLAN-T5-large** (Chung et al., 2022) as a comparison baseline.

4.2 Experimental Setup

To train INSTRUCTSTORY, we use instruction tuning on T5-large-lm-adapt for 5 epochs using the fully-supervised task mix. We use the Adam optimizer with a learning rate of 5e-5 and a batch size of 64. At each gradient step, examples are randomly sampled from all tasks. The maximum input and target sequence lengths are set to 1024, and any longer inputs or targets will be truncated.

For the fully-supervised learning scenario, both INSTRUCTSTORY and all the baselines are finetuned on a single task for 10 epochs for each task. The best performing checkpoint for each task is chosen based on the performance on its dev set. Note that BERT-large, RoBERTa-Large, and DeBERTa-v2-xlarge all have a maximum sequence length of 512, while GPT2-medium and GPT2-Large have a maximum sequence length of 1024 and OPT-350m has a maximum sequence length of 2048. We truncate the data instances based on the respective max sequence lengths of the models.

For the few-shot learning scenario, we finetune all the models and use early stopping based on the dev set performance. Also, we are unable to use in-context learning demonstrations like in Chung et al. (2022), as the story lengths are often too long to fit within the max input sequence length.

For the zero-shot scenarios, we only compare INSTRUCTSTORY with T5 and FLAN-T5, as the other baseline models have poor zero-shot performance.

More information about training specifics and hyperparamters can be seen in Appendix A.2.

4.3 Main Results

Fully-supervised Results The fully-supervised results are presented in Table 3. We show that INSTRUCTSTORY can achieve a 1.53 point increase in the overall average score compared to the single-task finetuned T5 baseline. Additionally, for understanding tasks, INSTRUCTSTORY outperforms T5 by 2.06 points. When compared to other strong understanding baselines including BERT, RoBERTa, and DeBERTa, INSTRUCTSTORY also achieves

Task Type	Task	BERT	RoBERTa	DeBERTa	T5	InstructStory
Genre Classification†	animals	82.69	86.02	82.24	82.88	**86.79**
	fantasy	43.70	47.37	48.75	47.95	**50.98**
	horror	45.67	55.64	**60.15**	52.05	53.33
	war	59.77	68.97	76.00	70.59	**78.26**
	poetry	78.90	**85.71**	79.65	81.97	84.96
	drama	42.67	45.30	46.43	44.21	47.40
	mystery	43.58	51.47	48.53	47.48	**51.97**
	fanfiction	55.28	62.26	**67.27**	63.41	66.07
	dystopia	43.48	57.14	61.16	52.23	**63.55**
	sci-fi	65.42	61.07	**67.24**	62.69	66.67
	AVG	51.86	61.15	**62.20**	60.15	<u>61.88</u>
Author Attribution†	aspiringwriter	66.67	**69.57**	62.02	60.40	67.18
	sagittarius	50.94	54.74	58.02	48.52	**64.81**
	Hope!	61.82	**81.13**	62.30	56.21	68.22
	Shasta	52.17	55.56	58.49	37.04	**59.38**
	Scorpio :)	61.82	**81.13**	62.30	56.21	68.22
	Zed	67.27	72.94	**81.82**	73.27	78.85
	Nathan.N	82.61	84.78	86.00	86.32	**87.23**
	Ellipsis	78.85	**83.67**	59.38	67.89	78.00
	Luke V.	72.09	69.77	69.23	63.24	**73.79**
	Amelia Rose	50.00	**70.10**	68.57	53.62	68.97
	AVG	64.52	**72.31**	69.08	62.03	<u>70.79</u>
Author Verification	author_verification	23.19	23.41	23.17	22.94	<u>**23.57**</u>
Temporal Inference	temporal_inference	72.90	77.74	**80.18**	78.51	<u>79.04</u>
Connectivity Inference	connectivity_inference	65.03	62.97	67.61	67.20	<u>**68.72**</u>
Story Scoring	likes_scoring	53.54	**75.74**	60.81	67.35	<u>68.82</u>
	stars_scoring	55.34	**66.60**	56.02	63.15	<u>63.26</u>
Story Segmentation	story_segmentation	31.38	47.28	41.09	46.87	<u>**47.33**</u>
Understanding AVG		51.90	59.43	57.39	57.56	<u>**59.62**</u>
Task Type	Task	**GPT2-l**	**GPT2-m**	**OPT-350m**	**T5**	**InstructStory**
Next Chapter Generation	next_chapter	81.35	80.90	**83.25**	82.17	<u>82.43</u>
Conditional Story Generation	conditional	79.40	79.33	**82.39**	81.10	<u>81.24</u>
Chapter Infilling	chapter_infilling	80.93	80.67	**82.89**	82.34	<u>82.51</u>
Global Infilling	global_infilling	81.49	81.30	**83.70**	82.22	<u>82.44</u>
Temporal Ordering	temporal_ordering	76.49	76.33	92.77	90.08	<u>**93.14**</u>
Generation AVG		79.93	79.71	**85.00**	83.58	<u>84.35</u>
Understanding and Generation Overall AVG		-	-	-	68.40	<u>**69.93**</u>

Table 3: Fully-supervised results of INSTRUCTSTORY and other baselines. **Bold** numbers indicate the best score across all models, and <u>underlined</u> numbers indicate cases where INSTRUCTSTORY outperforms the T5 baseline. Due to space limits, only 10 random tasks from the task type are shown. Full results can be found in the Appendix A.3.

the best results. For generation tasks, INSTRUCTSTORY outperforms T5 by 0.77 points. It also achieves favorable performance when compared to other strong generation baselines such as GPT2-medium and GPT2-large, although performing a little bit worse than OPT-350m. We hypothesize that the difference in performance between OPT-350m and INSTRUCTSTORY is due to the base model, specifically the size of the pretraining corpus (35B tokens vs 180B tokens).(Zhang et al., 2022)

Few-shot Results The few-shot results are shown in Table 4. For the few-shot scenario, INSTRUCTSTORY achieves the highest score of 61.44, followed by FLAN-T5 which achieved the second highest score of 59.45, outperforming all the T5, BERT, RoBERTa, and DeBERTa baselines. This demonstrates that even when instruction-tuned on a different dataset distribution, FLAN-T5 can still achieve competitive results when further fine-tuned for few-shot tasks.

task	BERT	RoBERTa	DeBERTa	T5	FLAN-T5	InstructStory
wordgames	59.65	**80.90**	77.27	62.40	71.05	73.68
rebellion	38.38	45.87	33.33	43.24	**50.00**	**50.00**
mythology	47.27	59.79	61.54	62.07	66.67	**67.33**
future	30.00	40.00	50.90	36.23	44.86	**54.70**
friendship	38.82	46.96	44.62	49.23	53.33	**55.36**
fairytale	45.93	60.32	65.52	74.07	72.09	**79.59**
dreams	47.48	64.15	58.62	**78.16**	71.26	76.74
crime	48.54	**66.67**	36.04	65.42	62.22	65.26
change	44.00	**50.36**	32.91	33.90	47.89	39.19
action	38.30	40.25	36.47	41.13	**55.10**	52.54
AVG	43.84	55.53	49.72	54.59	59.45	**61.44**

Table 4: Few-shot benchmark results. INSTRUCTSTORY outperforms all other baselines.

task†	T5	FLAN-T5	InstructStory
reality	32.56	39.56	39.47
lies	30.22	46.34	70.33
vampire	19.12	63.33	58.82
surreal	31.41	33.86	46.25
suspense	31.82	42.77	43.68
supernatural	39.34	48.28	45.33
family	14.88	51.16	60.00
revenge	35.00	58.06	57.14
crazy	30.00	42.31	43.08
world	30.63	34.92	50.75
AVG	32.09	47.79	**60.00**

Table 5: Zero-shot benchmark results. INSTRUCTSTORY out performs T5 and even FLAN-T5. †: Due to space limits, we only show 10 random tasks. Full results can be found in Appendix A.3.

Zero-shot Results We can see the zero-shot results in Table 5. In the zero-shot scenario, we compare INSTRUCTSTORY with T5 and FLAN-T5, and we can see that INSTRUCTSTORY has a significant improvement in zero-shot performance, a 28.08 increase from T5 and a 12.21 increase from FLAN-T5. This is expected because our instruction tuning training task mix has a similar, though unseen, data distribution to the zero-shot test sets.

4.4 Discussions

INSTRUCTSTORY brings a robust improvement in performance. By comparing T5 and INSTRUCTSTORY in Table 3, we see that INSTRUCTSTORY scores higher than T5 in every task type. The performance gain is consistent across all task types. Even on the task level, INSTRUCTSTORY achieves better results than T5 in 24 out of 27 genre classification tasks and 23 out of 30 authorship attribution tasks. This indicates that in fully-supervised scenario, one can confidently use the power of instruction tuning to improve performance.

	IS	IS_U	IS_G	T5
Fully-sup AVG	61.88	61.27	60.45	60.15
Few-shot AVG	61.44	59.83	54.95	54.59
Zero-shot AVG	60.00	58.41	32.31	32.09

Table 6: INSTRUCTSTORY vs its variants IS_U and IS_G.

Ablation: Instruction tuning with both understanding and generation tasks is more effective than instruction tuning with only understanding tasks or only generation tasks. Table 6 illustrates this by comparing the fully-supervised, few-shot, and zero-shot genre classification scores of INSTRUCTSTORY, its variants IS_U, and IS_G, where IS_U and IS_G are instruction tuned with understanding tasks mix and generation tasks mix, separately. From the table, we can see that IS > IS_U > IS_G > T5 across all zero-shot, few-shot, and fully-supervised learning scenarios, which indicates that instruction tuning with a mix of understanding and generation tasks is better than instruction tuning with only one of them.

5 Conclusion

We introduced a novel dataset STORYWARS and a multitask benchmark for collaborative story understanding and generation. Our proposed INSTRUCTSTORY model, which leverages instruction tuning as multitask pre-finetuning, outperformed both its single-task finetuning baseline and other strong models on the STORYWARS benchmark and established strong performance in all zero-shot, few-shot, and fully-supervised learning scenarios. We hope that our newly proposed STORYWARS dataset will serve as a catalyst for research in the field of collaborative storytelling and inspire further advancements in this area.

6 Limiations

Our proposed InstructStory method utilizes both single-task finetuning and instruction tuning to achieve good results. However, when finetuned on a new task, the model may suffer from the problem of catastrophic forgetting and lose its multitasking generalization abilities. Recent research by Scialom et al. (2022) has investigated this issue in instruction-tuned models and proposed a technique called Rehearsal to mitigate it. However, this work primarily focuses on zero-shot scenarios and does not address fully-supervised learning. It would be of interest to explore whether it is possible to finetune on a single task while preserving the model's multitasking abilities and generalization capabilities. We leave this question as an area for future research.

Additionally, it is important to note that our approach of single-task finetuning for each downstream task results in multiple models being required to be served simultaneously, which can lead to increased computational costs. In practice, this is a trade-off that must be carefully considered, as it requires balancing performance requirements with the resources available. It can be an important factor to consider when implementing this approach in real-world settings.

In the end, a proper and thorough evaluation of collaborative story generation remains an on-going research. While automatic evaluation metrics such as BERTScore has the best human correlations at story-level and system-level per Chhun et al. (2022), it may not be comprehensive enough in evaluating the highly creative output of collaborative story generation. There is a need for more nuanced and sophisticated metrics that can capture the complexity and diversity of collaborative stories. Therefore, the development and validation of appropriate evaluation methods is crucial for progress in this field.

7 Ethical Considerations

In Section 3.1, we have discussed our procedures to identify and remove potential harmful content and user privacy information. However, it is important to also consider the broader ethical implications of using AI in collaborative storytelling. These include issues such as ensuring fair and unbiased representation, protecting data privacy, and preventing the use of AI-generated content for harmful purposes. For example, AI-generated stories or characters may perpetuate stereotypes or reinforce societal biases if they are trained on biased data. Therefore, it is crucial to consider and address these ethical issues in order to create inclusive and responsible AI-generated stories that do not harm individuals or groups.

References

Armen Aghajanyan, Anchit Gupta, Akshat Shrivastava, Xilun Chen, Luke Zettlemoyer, and Sonal Gupta. 2021. Muppet: Massive multi-task representations with pre-finetuning. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 5799–5811, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nader Akoury, Shufan Wang, Josh Whiting, Stephen Hood, Nanyun Peng, and Mohit Iyyer. 2020. STORIUM: A Dataset and Evaluation Platform for Machine-in-the-Loop Story Generation. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 6470–6484, Online. Association for Computational Linguistics.

Vamsi Aribandi, Yi Tay, Tal Schuster, Jinfeng Rao, Huaixiu Steven Zheng, Sanket Vaibhav Mehta, Honglei Zhuang, Vinh Q. Tran, Dara Bahri, Jianmo Ni, Jai Gupta, Kai Hui, Sebastian Ruder, and Donald Metzler. 2022. Ext5: Towards extreme multi-task scaling for transfer learning. In International Conference on Learning Representations.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5:135–146.

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

Cyril Chhun, Pierre Colombo, Fabian M. Suchanek, and Chloé Clavel. 2022. Of human criteria and automatic metrics: A benchmark of the evaluation of story generation. In Proceedings of the 29th International Conference on Computational Linguistics, pages 5794–5836, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

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

Hyung Won Chung, Le Hou, S. Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Wei Yu, Vincent Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed Huai hsin Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. ArXiv, abs/2210.11416.

Jennifer Coates. 1997. The construction of a collaborative floor in women's friendly talk.

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

Angela Fan, Mike Lewis, and Yann Dauphin. 2019. Strategies for structuring story generation. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 2650–2660, Florence, Italy. Association for Computational Linguistics.

Chris Fournier. 2013. Evaluating text segmentation using boundary edit distance. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1702–1712, Sofia, Bulgaria. Association for Computational Linguistics.

Varun Gangal, Steven Y. Feng, Eduard H. Hovy, and Teruko Mitamura. 2021. NAREOR: the narrative reordering problem. CoRR, abs/2104.06669.

Sebastian Gehrmann, Tosin Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Anuoluwapo Aremu, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna-Adriana Clinciu, Dipanjan Das, Kaustubh Dhole, Wanyu Du, Esin Durmus, Ondřej Dušek, Chris Chinenye Emezue, Varun Gangal, Cristina Garbacea, Tatsunori Hashimoto, Yufang Hou, Yacine Jernite, Harsh Jhamtani, Yangfeng Ji, Shailza Jolly, Mihir Kale, Dhruv Kumar, Faisal Ladhak, Aman Madaan, Mounica Maddela, Khyati Mahajan, Saad Mahamood, Bodhisattwa Prasad Majumder, Pedro Henrique Martins, Angelina McMillan-Major, Simon Mille, Emiel van Miltenburg, Moin Nadeem, Shashi Narayan, Vitaly Nikolaev, Andre Niyongabo Rubungo, Salomey Osei, Ankur Parikh, Laura Perez-Beltrachini, Niranjan Ramesh Rao, Vikas Raunak, Juan Diego Rodriguez, Sashank Santhanam, João Sedoc, Thibault Sellam, Samira Shaikh, Anastasia Shimorina, Marco Antonio Sobrevilla Cabezudo, Hendrik Strobelt, Nishant Subramani, Wei Xu, Diyi Yang, Akhila Yerukola, and Jiawei Zhou. 2021. The GEM benchmark: Natural language generation, its evaluation and metrics. In Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021), pages 96–120, Online. Association for Computational Linguistics.

Seraphina Goldfarb-Tarrant, Tuhin Chakrabarty, Ralph Weischedel, and Nanyun Peng. 2020. Content planning for neural story generation with aristotelian rescoring. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4319–4338, Online. Association for Computational Linguistics.

Jian Guan, Zhuoer Feng, Yamei Chen, Ruilin He, Xiaoxi Mao, Changjie Fan, and Minlie Huang. 2022. LOT: A story-centric benchmark for evaluating Chinese long text understanding and generation. Transactions of the Association for Computational Linguistics, 10:434–451.

Daphne Ippolito, David Grangier, Chris Callison-Burch, and Douglas Eck. 2019. Unsupervised hierarchical story infilling. In Proceedings of the First Workshop on Narrative Understanding, pages 37–43, Minneapolis, Minnesota. Association for Computational Linguistics.

Daniel Khashabi, Gabriel Stanovsky, Jonathan Bragg, Nicholas Lourie, Jungo Kasai, Yejin Choi, Noah A. Smith, and Daniel S. Weld. 2021. GENIE: A leaderboard for human-in-the-loop evaluation of text generation. CoRR, abs/2101.06561.

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.

Shih-Ting Lin, Nathanael Chambers, and Greg Durrett. 2021. Conditional generation of temporally-ordered event sequences. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 7142–7157, Online. Association for Computational Linguistics.

Dayiheng Liu, Yu Yan, Yeyun Gong, Weizhen Qi, Hang Zhang, Jian Jiao, Weizhu Chen, Jie Fu, Linjun Shou, Ming Gong, Pengcheng Wang, Jiusheng Chen, Daxin Jiang, Jiancheng Lv, Ruofei Zhang, Winnie Wu, Ming Zhou, and Nan Duan. 2021. GLGE: A new general language generation evaluation benchmark. In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 408–420, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. CoRR, abs/1907.11692.

Annie Louis and Charles Sutton. 2018. Deep dungeons and dragons: Learning character-action interactions from role-playing game transcripts. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), pages 708–713, New Orleans, Louisiana. Association for Computational Linguistics.

Shuai Lu, Daya Guo, Shuo Ren, Junjie Huang, Alexey Svyatkovskiy, Ambrosio Blanco, Colin B. Clement, Dawn Drain, Daxin Jiang, Duyu Tang, Ge Li, Lidong Zhou, Linjun Shou, Long Zhou, Michele Tufano, Ming Gong, Ming Zhou, Nan Duan, Neel Sundaresan, Shao Kun Deng, Shengyu Fu, and Shujie Liu. 2021. Codexglue: A machine learning benchmark dataset for code understanding and generation. CoRR, abs/2102.04664.

Lara J. Martin, Prithviraj Ammanabrolu, William Hancock, Shruti Singh, Brent Harrison, and Mark O. Riedl. 2017. Event representations for automated story generation with deep neural nets. CoRR, abs/1706.01331.

S. Mehri, M. Eric, and D. Hakkani-Tur. 2020. Dialoglue: A natural language understanding benchmark for task-oriented dialogue. ArXiv, abs/2009.13570.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 839–849, San Diego, California. Association for Computational Linguistics.

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.

Nanyun Peng, Marjan Ghazvininejad, Jonathan May, and Kevin Knight. 2018. Towards controllable story generation. In NAACL Workshop.

Lianhui Qin, Antoine Bosselut, Ari Holtzman, Chandra Bhagavatula, Elizabeth Clark, and Yejin Choi. 2019. Counterfactual story reasoning and generation. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 5043–5053, Hong Kong, China. 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. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. CoRR, abs/1910.10683.

Hannah Rashkin, Asli Celikyilmaz, Yejin Choi, and Jianfeng Gao. 2020. PlotMachines: Outline-conditioned generation with dynamic plot state tracking. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4274–4295, Online. 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.

Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. 2022. Fine-tuned language models are continual learners.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ameet Annasaheb Rahane, Anantharaman S. Iyer, Anders Andreassen, Andrea Santilli, Andreas Stuhlmuller, Andrew M. Dai, Andrew D. La, Andrew Kyle 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 Karakacs, Bridget R. Roberts, Bao Sheng Loe, Barret Zoph, Bartlomiej Bojanowski, Batuhan Ozyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Stephen Howald, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, C'esar Ferri Ram'irez, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Chris Waites, Christian Voigt, Christopher D. Manning, Christopher Potts, Cindy Tatiana Ramirez, Clara Rivera, Clemencia Siro, Colin Raffel, Courtney Ashcraft, Cristina Garbacea, Damien Sileo, Daniel H Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Gonz'alez, Danny Hernandez, Danqi Chen, Daphne Ippolito, Dar Gilboa, David Dohan, D. 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, Ekaterina Shutova, Ekin Dogus Cubuk, Elad Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth P. Donoway, Ellie Pavlick, Emanuele Rodolà, Emma FC Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang, Ethan A. Chi, Ethan Dyer, Ethan J. Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fan Xia, Fatemeh Siar, Fernando Mart'inez-Plumed, Francesca Happ'e, François Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Wang, Gonzalo Jaimovitch-L'opez, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Han Sol Kim, Hannah Rashkin, Hanna Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Shevlin, Hinrich Schütze, Hiromu Yakura, Hongming Zhang, Hubert Wong, Ian Aik-Soon Ng, Isaac Noble, Jaap Jumelet, Jack Geissinger, John Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, J. Brooker Simon, James Koppel, James Zheng, James Zou, Jan Koco'n, Jana Thompson, Jared Kaplan, Jarema Radom, Jascha Narain Sohl-Dickstein, Jason Phang, Jason Wei, Jason Yosinski, Jekaterina Novikova, Jelle Bosscher, Jenni Marsh, Jeremy Kim, Jeroen Taal, Jesse Engel, Jesujoba Oluwadara Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Jane W Waweru, John Burden, John Miller, John U. Balis, Jonathan Berant, Jorg Frohberg, Jos Rozen, José Hernández-Orallo, Joseph Boudeman, Joseph Jones, Joshua B. Tenenbaum, Joshua S. Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakrishnan, Katerina Ignatyeva, Katja Markert, Kaustubh D. Dhole, Kevin Gimpel, Kevin Ochieng' 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, Luca Lam, Lucy Noble, Ludwig Schmidt, Luheng He, Luis Oliveros Col'on, Luke Metz, Lutfi Kerem cSenel, Maarten Bosma, Maarten Sap, Maartje ter Hoeve, Madotto Andrea, Maheen Saleem Farooqi, Manaal Faruqui, Mantas Mazeika, Marco Baturan, Marco Marelli, Marco Maru, M Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew Leavitt, Matthias Hagen, M'aty'as Schubert, Medina Baitemirova, Melissa Arnaud, Melvin Andrew McElrath, Michael A. Yee, Michael Cohen, Mi Gu, Michael I. Ivanitskiy, Michael Starritt, Michael Strube, Michal Swkedrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Monica Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, T MukundVarma, Nanyun Peng, Nathan Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas S. Roberts, Nicholas Doiron, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish Keskar, Niveditha 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, Pi-Bei Hwang, P. Milkowski, Piyush S. Patil, Pouya Pezeshkpour, Priti Oli, Qiaozhu Mei, QING LYU, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer Gabriel, Rahel Habacker, Ram'on Risco Delgado, 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, Ruslan Salakhutdinov, Ryan Chi, Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib J. Singh, Saif M. Mohammad, Sajant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Sam Bowman, Samuel S. Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A. Rous, Sarik Ghazarian, Sayan Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, Shadi S. Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, Soo hwan Lee, Spencer Bradley Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano Ermon, Stella Rose Biderman, Stephanie C. Lin, S. Prasad, Steven T. Piantadosi, Stuart M. Shieber, Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq A. Ali, Tatsuo Hashimoto, Te-Lin Wu, Theo Desbordes, Theodore Rothschild, Thomas Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, T. N. Kornev, Timothy Telleen-Lawton, Titus Tunduny, Tobias Gerstenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler O'Brien 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, W Vossen, Xiang Ren, Xiaoyu F Tong, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yang Song, Yasaman Bahri, Ye Ji Choi, Yichi Yang, Yiding Hao, Yifu Chen, Yonatan Belinkov, Yu Hou, Yu Hou, Yuntao Bai, Zachary Seid, Zhao Xinran, Zhuoye Zhao, Zi Fu Wang, Zijie J. Wang, Zirui Wang, Ziyi Wu, Sahib Singh, and Uri Shaham. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. ArXiv, abs/2206.04615.

Simeng Sun, Katherine Thai, and Mohit Iyyer. 2022. ChapterBreak: A challenge dataset for long-range language models. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 3704–3714, Seattle, United States. Association for Computational Linguistics.

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

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aur'elien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. ArXiv, abs/2302.13971.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc.

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

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2021. Finetuned language models are zero-shot learners. CoRR, abs/2109.01652.

Hanwei Xu, Yujun Chen, Yulun Du, Nan Shao, Yanggang Wang, Haiyu Li, and Zhilin Yang. 2022. Zeroprompt: Scaling prompt-based pretraining to 1, 000 tasks improves zero-shot generalization. CoRR, abs/2201.06910.

Jingjing Xu, Xuancheng Ren, Yi Zhang, Qi Zeng, Xiaoyan Cai, and Xu Sun. 2018. A skeleton-based model for promoting coherence among sentences in narrative story generation. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 4306–4315, Brussels, Belgium. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc.

Lili Yao, Nanyun Peng, Weischedel Ralph, Kevin Knight, Dongyan Zhao, and Rui Yan. 2019. Plan-and-write: Towards better automatic storytelling. In The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19).

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.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In International Conference on Learning Representations.

A Appendix

A.1 Instruction Template examples

Please refer to Table 7 for the instruction template examples.

A.2 Hypterparameters

Please refer to Table 8 for the hyperparameters.

name	value
batch size	64
learning rate	5e-5
training steps	50000
warmup steps	2000

Table 8: Hypterparameters for INSTRUCTSTORY

A.3 Full results tables

Please refer to Table 9, Table 10, Table 11, and Table 12 for all full results.

task type	input format	output format
genre classification	{story} Is this a {genre} story?	Yes or No
authorship attribution	{story} Is this story written by {author}?	Yes or No
authorship verification	Chapter A: {chapter_a} Chapter B: {chapter_b} Are the two story chapters above written by the same author?	Yes or No
connectivity inference	Chapter A: {chapter_a} Chapter B: {chapter_b} Can Chapter B be the next chapter of Chapter A?	Yes or No
temporal inference	Chapter A: {chapter_a} Chapter B: {chapter_b} Does Chapter A happen before Chapter B?	Yes or No
story scoring	{story} How do you like the story above? Please rate the story from 0 to 10:	0.0 - 10.0
story segmentation	{story} Please segment the story into chapters:	{c_1} ||| {c_2} ||| {c_3} ...
next chapter generation	{$\text{story}_{0:\text{i}}$} Please write a next chapter for the above story:	{chapter_i}
conditional story generation	{$\text{story}_{0:\text{i}}$} Please finish the whole story:	{$\text{story}_{\text{i}:}$}
chapter infilling	Chapter A: {chapter_a} Chapter B: {chapter_b} Please write a chapter between Chapter A and Chapter B:	{chapter_i}
global infilling	Previous chapters: {story_prev} Next chapters: {story_next} Based on the context of previous and next chapters, please fill in a chapter in between:	{chapter_i}
temporal ordering	{$\text{story}_\text{permute}$} Please rewrite the story in correct temporal order:	{$\text{story}_\text{correct}$}

Table 7: Instruction template examples.

task	**BERT**	**RoBERTa**	**DeBERTa**	**T5**	**InstructStory**
war	59.77	68.97	76.0	70.59	78.26
life	35.41	40.0	37.5	51.75	46.48
fanfiction	55.28	62.26	67.27	63.41	66.07
poetry	78.9	85.71	79.65	81.97	84.96
music	69.14	83.87	85.42	83.17	86.6
fantasy	43.7	47.37	48.75	47.95	50.98
humor	60.61	54.12	62.22	61.95	56.07
lgbt	48.08	60.24	63.83	59.81	55.77
school	36.14	63.24	65.22	51.22	51.76
game	58.62	77.55	77.42	68.24	69.57
sad	48.35	56.93	53.97	53.44	55.17
nature	39.51	51.43	48.08	51.85	47.17
magic	60.61	63.74	61.9	59.42	61.76
adventure	40.43	55.24	46.38	44.32	45.64
sci-fi	65.42	61.07	67.24	62.69	66.67
romance	54.84	59.68	60.29	56.52	62.12
hero	32.26	56.14	61.9	70.97	71.84
euphoric	28.26	40.35	44.83	44.59	43.1
space	72.73	74.23	78.72	80.0	78.9
survival	29.73	58.59	59.32	53.06	52.38
mystery	43.58	51.47	48.53	47.48	51.97
drama	42.67	45.3	46.43	44.21	47.4
royalty	72.73	74.0	68.18	74.75	75.47
dystopia	43.48	57.14	61.16	52.23	63.55
death	51.57	60.87	66.67	53.59	60.94
horror	45.67	55.64	60.15	52.05	53.33
animals	82.69	86.02	82.24	82.88	86.79
intellikat	76.47	80.43	72.41	72.0	80.0
Hope!	61.82	81.13	62.3	56.21	68.22
ArtemisNine	46.58	68.42	58.14	65.98	69.09
Mockingjay	50.98	64.52	57.97	31.58	55.63
Rosetta	70.83	78.72	73.79	69.81	78.0
ember	46.6	68.09	59.26	55.71	55.12
CheshireinWonderland	47.31	55.42	63.04	40.7	58.41
Ellipsis	78.85	83.67	59.38	67.89	78.0
Scorpio :)	58.82	73.08	61.54	53.42	64.83
DANDAN THE DANDAN	63.27	70.73	76.6	65.22	71.11
Luke V.	72.09	69.77	69.23	63.24	73.79
Windlion	87.13	90.38	93.07	88.89	92.16
Kitin	86.87	83.72	78.18	80.0	74.42
Tricia L	43.84	70.09	61.29	45.59	64.71
Nathan.N	82.61	84.78	86.0	86.32	87.23
Zed	67.27	72.94	81.82	73.27	78.85
CAPSLOCK	77.59	74.38	80.81	67.96	80.37
R	65.26	88.89	85.71	78.26	88.89
go!den-in-the-mist	78.85	84.96	78.9	66.17	72.73
Libra (inactive)	54.14	62.3	57.89	54.55	57.66
Silverfroststorm	75.79	67.83	55.7	51.5	63.16
Shasta	52.17	55.56	58.49	37.04	59.38
SaintSayaka	71.43	75.21	77.06	61.87	75.23
Amelia Rose	50.0	70.1	68.57	53.62	68.97
sagittarius	50.94	54.74	58.02	48.52	64.81
Phantim	66.67	81.55	78.1	70.59	76.79
Ara Argentum Aurum!	50.94	49.28	56.41	63.46	67.33
aspiringwriter	66.67	69.57	62.02	60.4	67.18
camel	71.15	73.12	77.06	64.41	66.67
darcy	62.65	65.98	63.64	66.67	64.86
author_verification	23.19	23.41	23.17	22.94	23.57
temporal_inference	72.90	77.74	80.18	78.51	79.04
connectivity_inference	65.03	62.97	67.61	67.20	68.72
likes_scoring	53.54	75.74	60.81	67.35	68.82
stars_scoring	55.34	66.60	56.02	63.15	63.26
story_segmentation	31.38	47.28	41.09	46.87	47.33

Table 9: Fully-supervised understanding results of INSTRUCTSTORY and other baselines.

Task	GPT2-l	GPT2-m	OPT-350m	T5	InstructStory
next_chapter	81.35	80.90	83.25	82.17	82.43
conditional	79.40	79.33	82.39	81.10	81.24
chapter_infilling	80.93	80.67	82.89	82.34	82.51
global_infilling	81.49	81.30	83.70	82.22	82.44
temporal_ordering	76.49	76.33	92.77	90.08	93.14

Table 10: Fully-supervised generation results of INSTRUCTSTORY and other baselines.

task	BERT	RoBERTa	DeBERTa	T5	FLAN-T5	InstructStory
wordgames	59.65	80.90	77.27	62.40	71.05	73.68
rebellion	38.38	45.87	33.33	43.24	50.00	50.00
mythology	47.27	59.79	61.54	62.07	66.67	67.33
future	30.00	40.00	50.90	36.23	44.86	54.70
friendship	38.82	46.96	44.62	49.23	53.33	55.36
fairytale	45.93	60.32	65.52	74.07	72.09	79.59
dreams	47.48	64.15	58.62	78.16	71.26	76.74
crime	48.54	66.67	36.04	65.42	62.22	65.26
change	44.00	50.36	32.91	33.90	47.89	39.19
action	38.30	40.25	36.47	41.13	55.10	52.54

Table 11: Few-shot results of INSTRUCTSTORY and other baselines.

task	T5	FLAN-T5	InstructStory
disease	30.36	62.3	67.69
harrypotter	29.63	84.21	85.71
dragons	30.22	70.42	95.0
art	34.53	54.84	87.36
memories	32.65	40.0	70.18
suspense	31.82	42.77	43.68
supernatural	39.34	48.28	45.33
angel	34.48	55.17	82.61
revenge	35.0	58.06	57.14
surreal	31.41	33.86	46.25
history	38.6	54.12	60.34
choices	40.51	28.7	50.0
vampire	19.12	63.33	58.82
lies	30.22	46.34	70.33
crazy	30.0	42.31	43.08
secret	36.19	39.49	44.59
pirates	35.97	41.51	65.63
world	30.63	34.92	50.75
hope	36.99	38.6	57.14
reality	32.56	39.56	39.47
family	14.88	51.16	60.0
emotions	34.67	34.67	60.18
strange	28.19	34.55	38.64

Table 12: Zero-shot results of INSTRUCTSTORY and other baselines.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitation is section 6 after conclusion

☑ A2. Did you discuss any potential risks of your work?
under ethical considerations in section 7

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Left blank.

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

B ☑ Did you use or create scientific artifacts?

Section 3 dataset

☐ B1. Did you cite the creators of artifacts you used?
Not applicable. Left blank.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

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

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Section 3.1

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Left blank.

C ☑ Did you run computational experiments?

section 4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 4 specifies the number of parameters of models.

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

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
section 4

☐ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Not applicable. Left blank.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
section 3.2.3

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

Left blank.

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

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

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

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

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