Learning to Generate Instructions to Adapt Language Models to New Tasks

Nihal V. Nayak, Yiyang Nan, Avi Trost, and Stephen H. Bach
Department of Computer Science, Brown University
{nnayak2, ynan3, atrost, sbach}@cs.brown.edu

Abstract

We present Bonito, the first open-source model for *conditional task generation*: the problem of converting unannotated corpus into a collection of tasks for instruction tuning. Our goal is to enable efficient task adaptation of instruction tuned language models on users' specialized, private data without relying on proprietary API-access-only models like GPT-4. We create Bonito by remixing existing, general-purpose instruction tuning data into a new training mixture for conditional task generation. Bonito learns to generate new tasks conditioned on the text and desired task type. The generated instructions in the specialized domain can be used to further train language models. We demonstrate that this procedure leads to improved performance on extractive question answering and yes-no question answering: across four datasets, each in a different domain, Bonito improves the F1 score of FLAN T5 Small by an average of 14.5% and FLAN-T5 Base by an average of 4.4%. We also find that Bonito improves FLAN-T5 Large on two out of four datasets but shows a slight negative transfer on the other two datasets. Overall, these results show a promising direction for adapting instruction tuned language models to new tasks without using proprietary models.

1 Introduction

Instruction tuning [29, 38, 50] has become a key tool for getting strong zero-shot performance from large language models. By fine-tuning a language model on a natural language corpus of many *tasks*—each consisting of an input *instruction* and desired *response*)—the model generally improves in its ability to respond to unseen instructions. However, this generalization is still limited by the qualities of the instruction-tuning corpus. Existing corpora like the Public Pool of Prompts (P3) [3], Natural Instructions [29, 47], and Dolly-v2 [9] are focused on text from the Web, classic natural language datasets, and other tasks that generally do not require specialized domain knowledge, such as social media and e-commerce. In this work, we study how to better adapt instruction tuned models to tasks in specialized domains.

Task adaptation of instruction tuned models to specialized domains is important for bringing the benefits of large language models to a wider range of users. Recent evaluations—including evaluations of proprietary models—show that they often significantly underperform specialized models [21, 40, 55], particularly in specialized domains requiring subject matter expertise. Efforts to make domain-specialized language models repeat the time-consuming and labor-intensive creation of training tasks [11, 41, 53]. We aim to automate the generation of domain-specific training tasks to create specialized language models.

Recently, several works have generated tasks by prompting proprietary API-access-only models such as ChatGPT or GPT-4 to adapt language models [24, 44, 49]. In particular, Köksal et al. [24] conditions GPT-3.5 on the unlabeled domain corpus to automatically generate domain-specific tasks [24]. This prompting strategy takes advantage of the vast amounts of unannotated domain

Workshop on Instruction Tuning and Instruction Following at NeurIPS 2023.

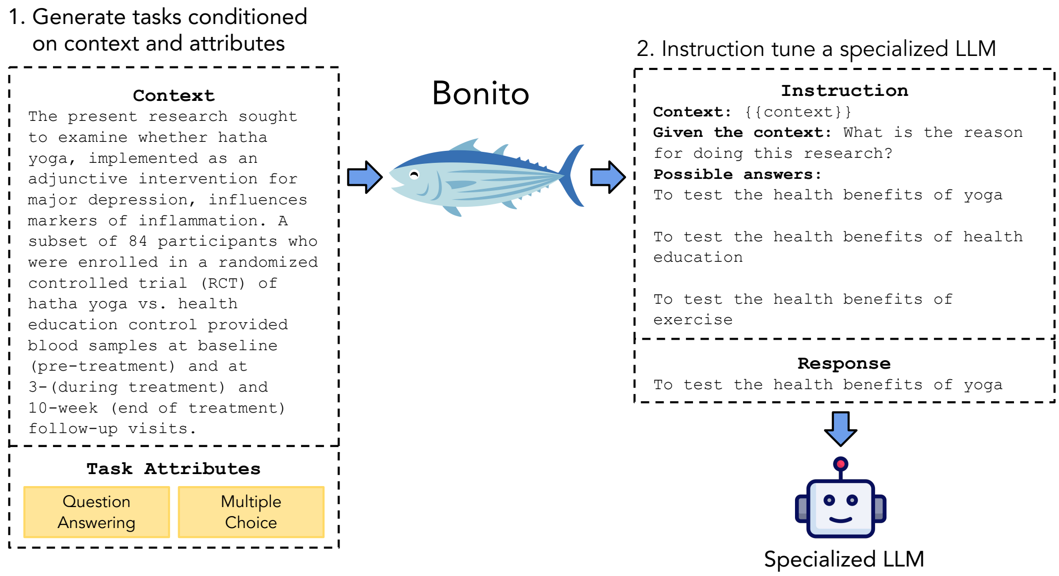

Figure 1: The workflow of Bonito for conditional task generation. Bonito takes the unannotated text or context as input, along with the specified task attributes to generate tasks. For each context, it generates an instruction that references that text and a target response. The output is then used to (further) instruction tune a language model that can be applied to the user's specific task.

corpus available to generate tasks. But, this route is expensive and not usable for proprietary or private research data. New approaches are needed to give users a more effective and accessible way to adapt language models to their own data.

In this paper, we propose to train an open-source model that can generate tasks for instruction tuning that are conditioned on a user's unannotated corpus (Figure 1). We call this problem *conditional task generation*. Our key idea is that we can make a new training dataset using existing datasets for instruction tuning. Datasets like P3 [3] and the FLAN collection [27] exist as templates that convert semi-structured examples of natural language tasks into a fully prompted format, in which both the input and the desired response are text strings. We start by selecting a subset of the templates in P3 that create tasks from *contexts*, which are pieces of text that are required for responding to the instruction. For example, a context could be a paragraph that should be summarized or that contains the answer to a question. We also annotate these templates with task attributes, i.e., the type of task they produce. We then use these templates to create meta-templates for training a new language model. Each meta-template produces training examples in which the input is context and task attributes, and the output is an entire task: the instruction (including the context) and the desired response. In this way, we can easily create abundant, diverse examples of conditional task generation. For example, the dataset we created from P3 contains over 1.5 million such examples. We fine-tune Falcon 7B [2] on this data to create an open-source model for conditional task generation, which we call Bonito.

We demonstrate that Bonito enables efficient task adaptation by generating training data for extractive question answering and yes-no question answering across four specialized domains and adapting an off-the-shelf instruction tuned model. Across three extractive question answering datasets from SQuADShifts [28, 46], Bonito improves FLAN T5 Small by an average of 9.5% and FLAN-T5 Base by an average of 2.3% on F1. However, we find that Bonito improves FLAN-T5 Large by 2.2% on the Reddit dataset and 0.1% on the Amazon dataset but results in negative transfer on the NYT datasets. We modify PubMedQA [20] as a yes-no question answering, which we call PubMedQA-YN, and generate the yes-no question answering tasks on the PubMed abstracts. Bonito on the PubMedQA-YN improves FLAN T5 Small by 29.7% and FLAN T5 Base by 10.5% but shows a small drop in performance on FLAN-T5 Large. Overall, these findings demonstrate the value of conditional task generation to adapt instructed tuned language models to new tasks in diverse and challenging domains.

```
Input
Task Type: question answering
Fine Grained Type: multiple-choice question
answering
Context: Amazing, I think this dog has been on a
leash less than 5 times in his life, but he is
doing fine. Then we get to the vet. Another thing
for Cuddle as he has to get inside a house which he
never does because of Ben's allergies. He has been
doing very fine with to injections and one blood
sample taken.
```

```
Output
{{context}}
What will happen if Cuddle comes inside the house ?
Pick the best answer from the following options:
- Cuddle will jump on Ben.
- None of the above choices.
- Ben will get fleas.
- Ben's allergies will act up.
[pipe]
Ben's allergies will act up.
```

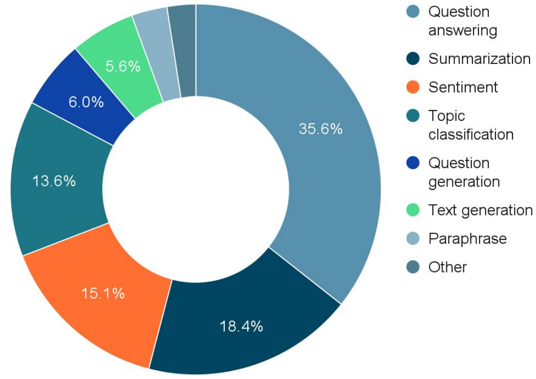

Figure 2: Left: example input-output pair from the attributed task generation mixture. Right: task distribution of the attributed task generation mixture.

2 Bonito: Learning to Generate Tasks

Key Properties We list key properties that we desire in our task generation model: (1) given a corpus containing articles and paragraphs, the model should take the text as input and generate high-quality tasks that require minimal cleaning or post-processing, (2) the model should adhere to the task type like extractive question answering or summarization task, and (3) the model should generate diverse tasks for the exact text with varying styles.

Conditioanl Task Generation with Attributes (CTGA) To create the model satisfying the key properties, we first create the training dataset: conditional task generation with attributes (CTGA). CTGA is a new large-scale dataset for task generation with a total of 1.5M examples with 11 attributes or task types. Figure 2 shows the task type distribution of the dataset.

The dataset is derived from P3 [3] where each input example is associated with a task type (`Task Type:`) as an attribute and optionally a fine-grained task type (`Fine-grained Type:`) followed by the text or context (`Context:`). The output is the attributed task with the prompt or task description and the context (`{context}`) followed by a pipe symbol (`[pipe]`) and the solution to the task. We use the `[pipe]` symbol to separate the input and output for the generated task. Figure 2 shows an example from the dataset.

The dataset is constructed by identifying datasets that require a context to complete the task. For example, SQuAD [35] requires the context to answer the extractive question answering task whereas CommonSenseQA [43] asks a multiple choice question without providing any relevant text. For our work, we consider datasets like SQuAD as it would enable us to convert unlabeled text in a new domain to tasks. We identified a total of 34 datasets to be included in CTGA (see Appendix D).

After selecting relevant datasets with a context, we annotate all the prompts with a task type and optionally a more specific fine-grained task type. Sanh et al. [38] associates each dataset with a task type but we find that a single dataset in PromptSource can have prompts corresponding to multiple task types. For example, in P3, the Social IQa dataset includes prompts for question answering as well as question generation tasks. In total, we annotated 11 task types with 4 additional fine-grained types for a total of 275 prompts. See Appendix D for the list of all the prompts and task types.

For the final CTGA training dataset, we apply the task templates along with the task types to all the relevant datasets. For each example in a dataset, we randomly sample a task template for the dataset and apply the template to create the attributed task example. We limit the total number of examples per dataset to 100,000. The final training dataset can be used to train a suitable model to generate tasks.

Training the Bonito Model We train Bonito with the Falcon-7B model, a decoder-only language model trained on 1 trillion tokens [2]. We include the hyperparameters and design considerations in Appendix B. The same training recipe can be used to train other existing decoder-only language models such as Mistral [1], Pythia [6] and RedPajama [8]. While models such as Llama2 [45] can be trained on CTGA, the license prohibits the use of the output from the LLama2 to enhance any other large language model.

Method	FLAN-T5-Small Reddit	FLAN-T5-Small Amazon	FLAN-T5-Small NYT	FLAN-T5-Base Reddit	FLAN-T5-Base Amazon	FLAN-T5-Base NYT	FLAN-T5-Large Reddit	FLAN-T5-Large Amazon	FLAN-T5-Large NYT
Zero-Shot	43.7	50.4	64.1	66.3	67.9	76.9	71.4	73.3	81.8
Gold	68.1 $_{0.2}$	68.6 $_{0.1}$	75.0 $_{0.0}$	75.5 $_{0.2}$	75.9 $_{0.0}$	81.3 $_{0.1}$	78.9 $_{0.1}$	79.3 $_{0.1}$	83.8 $_{0.1}$
Bonito (Ours)	57.0 $_{0.3}$	60.2 $_{0.0}$	69.6 $_{0.2}$	69.8 $_{0.1}$	70.3 $_{0.3}$	78.2 $_{0.2}$	73.6 $_{0.5}$	73.4 $_{0.3}$	81.1 $_{0.3}$
Δ Zero-Shot	+13.3	+9.8	+5.5	+3.5	+2.4	+1.2	+2.2	+0.1	-0.7

Table 1: Results for extractive question answering on the SQuADShifts benchmark.

3 Experiments

To evaluate the quality of the task generations, we perform extrinsic evaluation by training instruction tuned models on tasks generated by Bonito in specialized domains. This section includes the task details, datasets, models, baselines, and results for the experiments. We include additional information including task generation, training, and evaluation details in Appendix C.

Task Details We perform extrinsic evaluation on extractive question answering and yes-no question answering tasks. In both tasks, we have access to unannotated text from the training split of the target datasets. We generate the tasks with Bonito to get the synthetic training dataset. We then train the model, an instruction tuned model in our experiments, and evaluate the trained model on the test set of the target dataset.

Datasets We experiment with the SQuADShifts benchmark for extractive question answering and the PubMedQA-YN dataset for yes-no question answering. The SQuADShifts benchmark [46], created from the SQuADShifts challenge extractive question answering datasets [28], contains three datasets: Reddit, Amazon, and NYT. PubMedQA [20], created from the PubMed corpus, is a question answering dataset that contains a question and PubMed abstract paired with an answer that is yes, no, or maybe. We remove the maybe answer choice from the test set and call this dataset, PubMedQA-YN. Bonito is used to generate the tasks on the texts from the training splits of these datasets (see Appendix C for more details).

Models In our experiments, we adapt the FLAN models to new tasks [27]. FLAN is an instruction tuned language model trained with a pretrained T5 model on 1836 tasks. We experiment with three models: FLAN-T5 Small (80M), FLAN-T5 Base (250M), and FLAN-T5 Large (780M).

Baselines We consider two key baselines: zero-shot and supervised baseline (Gold). We use FLAN as a zero-shot baseline as they have demonstrated impressive performance on held-out datasets [27]. The supervised baseline (Gold) establishes the upper bound of performance on the task.

Method	Small	Base	Large
Zero-Shot	31.8 $_{0.3}$	57.0 $_{0.2}$	75.3 $_{0.3}$
Gold	65.1 $_{0.4}$	73.8 $_{0.2}$	79.8 $_{0.3}$
Bonito (Ours)	61.5 $_{0.4}$	67.5 $_{0.3}$	74.6 $_{0.3}$
Δ Zero-Shot	+29.7	+10.5	-0.7

Table 2: Results for Yes-no question answering on PubMedQA-YN. Small, Base, and Large are shorthand for the different FLAN models.

Results Table 1 shows that Bonito improves F1 score over the zero-shot performance of FLAN-T5 Small by an average of 9.5% and FLAN-T5 Base by an average of 2.3% on SQuADShifts. We further improve the zero-shot performance of FLAN-T5 Large by 2.2% on the Reddit dataset and 0.1% on the Amazon dataset. However, we find that Bonito with FLAN-T5 Large leads to negative transfer on the NYT dataset. On the PubMedQA-YN dataset, we see that Bonito improves FLAN-T5 Small by an average of 29.7% and FLAN-T5 Base by an average of 10.5% but find the performance drop on the FLAN-T5 Large. These results show that synthetic tasks generated by Bonito significantly benefit smaller rather than larger instruction tuned models. This could be due to the fact that related NYT or PubMed tasks with gold labels are included in the FLAN-T5 training mixture and larger models remember them. Finally, we would like to highlight that Bonito significantly closes the gap between the instruction tuned model and the upper bound without any training data.

4 Conclusion

We present Bonito, a conditional task generation model that can be used to convert unannotated texts into tasks. We show that Bonito generated tasks can be used to further improve instruction tuned models. In the future, we aim to include a broader set of tasks such as multiple-choice question answering and summarization to show the effectiveness of Bonito.

Acknowledgements

We appreciate the comments and advice from Yeganeh Kordi on our draft. Robot icons created by Hilmy Abiyyu A. - Flaticon.[1] We gratefully acknowledge support from Cisco. Disclosure: Stephen Bach is an advisor to Snorkel AI, a company that provides software and services for data-centric artificial intelligence.

References

[1] M. AI. Mistral 7b, 2023. URL `https://mistral.ai/news/announcing-mistral-7b/`.

[2] E. Almazrouei, H. Alobeidli, A. Alshamsi, A. Cappelli, R. Cojocaru, M. Debbah, E. Goffinet, D. Heslow, J. Launay, Q. Malartic, B. Noune, B. Pannier, and G. Penedo. Falcon-40B: an open large language model with state-of-the-art performance. 2023.

[3] S. Bach, V. Sanh, Z. X. Yong, A. Webson, C. Raffel, N. V. Nayak, A. Sharma, T. Kim, M. S. Bari, T. Fevry, Z. Alyafeai, M. Dey, A. Santilli, Z. Sun, S. Ben-david, C. Xu, G. Chhablani, H. Wang, J. Fries, M. Al-shaibani, S. Sharma, U. Thakker, K. Almubarak, X. Tang, D. Radev, M. T.-j. Jiang, and A. Rush. PromptSource: An integrated development environment and repository for natural language prompts. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 93–104, Dublin, Ireland, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-demo.9. URL `https://aclanthology.org/2022.acl-demo.9`.

[4] Y. Bai, A. Jones, K. Ndousse, A. Askell, A. Chen, N. DasSarma, D. Drain, S. Fort, D. Ganguli, T. Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *ArXiv preprint*, abs/2204.05862, 2022. URL `https://arxiv.org/abs/2204.05862`.

[5] M. Bartolo, T. Thrush, R. Jia, S. Riedel, P. Stenetorp, and D. Kiela. Improving question answering model robustness with synthetic adversarial data generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8830–8848, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.696. URL `https://aclanthology.org/2021.emnlp-main.696`.

[6] S. Biderman, H. Schoelkopf, Q. Anthony, H. Bradley, K. O'Brien, E. Hallahan, M. A. Khan, S. Purohit, U. S. Prashanth, E. Raff, A. Skowron, L. Sutawika, and O. van der Wal. Pythia: A suite for analyzing large language models across training and scaling, 2023.

[7] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, E. Li, X. Wang, M. Dehghani, S. Brahma, et al. Scaling instruction-finetuned language models. *ArXiv preprint*, abs/2210.11416, 2022. URL `https://arxiv.org/abs/2210.11416`.

[8] T. Computer. Redpajama: An open source recipe to reproduce llama training dataset, 2023. URL `https://github.com/togethercomputer/RedPajama-Data`.

[9] M. Conover, M. Hayes, A. Mathur, J. Xie, J. Wan, S. Shah, A. Ghodsi, P. Wendell, M. Zaharia, and R. Xin. Free dolly: Introducing the world's first truly open instruction-tuned llm. 2023. URL `https://www.databricks.com/blog/2023/04/12/dolly-first-open-commercially-viable-instruction-tuned-llm`.

[10] J. Cui, Z. Li, Y. Yan, B. Chen, and L. Yuan. Chatlaw: Open-source legal large language model with integrated external knowledge bases. *ArXiv preprint*, abs/2306.16092, 2023. URL `https://arxiv.org/abs/2306.16092`.

[11] C. Deng, T. Zhang, Z. He, Q. Chen, Y. Shi, L. Zhou, L. Fu, W. Zhang, X. Wang, C. Zhou, Z. Lin, and J. He. Learning a foundation language model for geoscience knowledge understanding and utilization. *ArXiv preprint*, abs/2306.05064, 2023. URL `https://arxiv.org/abs/2306.05064`.

[1] https://www.flaticon.com/free-icons/robot

[12] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *ArXiv preprint*, abs/2305.14314, 2023. URL `https://arxiv.org/abs/2305.14314`.

[13] A. Glaese, N. McAleese, M. Trebacz, J. Aslanides, V. Firoiu, T. Ewalds, M. Rauh, L. Weidinger, M. Chadwick, P. Thacker, et al. Improving alignment of dialogue agents via targeted human judgements. *ArXiv preprint*, abs/2209.14375, 2022. URL `https://arxiv.org/abs/2209.14375`.

[14] A. Gudibande, E. Wallace, C. Snell, X. Geng, H. Liu, P. Abbeel, S. Levine, and D. Song. The false promise of imitating proprietary llms. *ArXiv preprint*, abs/2305.15717, 2023. URL `https://arxiv.org/abs/2305.15717`.

[15] P. Gupta, C. Jiao, Y.-T. Yeh, S. Mehri, M. Eskenazi, and J. Bigham. InstructDial: Improving zero and few-shot generalization in dialogue through instruction tuning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 505–525, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics. URL `https://aclanthology.org/2022.emnlp-main.33`.

[16] S. Gururangan, A. Marasović, S. Swayamdipta, K. Lo, I. Beltagy, D. Downey, and N. A. Smith. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.740. URL `https://aclanthology.org/2020.acl-main.740`.

[17] J. He, J. Gu, J. Shen, and M. Ranzato. Revisiting self-training for neural sequence generation. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL `https://openreview.net/forum?id=SJgdnAVKDH`.

[18] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *ArXiv preprint*, abs/1503.02531, 2015. URL `https://arxiv.org/abs/1503.02531`.

[19] O. Honovich, T. Scialom, O. Levy, and T. Schick. Unnatural instructions: Tuning language models with (almost) no human labor. In *Meeting of the Association for Computational Linguistics (ACL)*, 2023.

[20] Q. Jin, B. Dhingra, Z. Liu, W. Cohen, and X. Lu. Pubmedqa: A dataset for biomedical research question answering. 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 2567–2577, 2019.

[21] J. Kocoń, I. Cichecki, O. Kaszyca, M. Kochanek, D. Szydło, J. Baran, J. Bielaniewicz, M. Gruza, A. Janz, K. Kanclerz, A. Kocoń, B. Koptyra, W. Mieleszczenko-Kowszewicz, P. Milkowski, M. Oleksy, M. Piasecki, L. Radliński, K. Wojtasik, S. Woźniak, and P. Kazienko. ChatGPT: Jack of all trades, master of none. *Information Fusion*, page 101861, 2023.

[22] A. Köpf, Y. Kilcher, D. von Rütte, S. Anagnostidis, Z.-R. Tam, K. Stevens, A. Barhoum, N. M. Duc, O. Stanley, R. Nagyfi, et al. Openassistant conversations–democratizing large language model alignment. *ArXiv preprint*, abs/2304.07327, 2023. URL `https://arxiv.org/abs/2304.07327`.

[23] W. Kwon, Z. Li, S. Zhuang, Y. Sheng, L. Zheng, C. H. Yu, J. E. Gonzalez, H. Zhang, and I. Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

[24] A. Köksal, T. Schick, A. Korhonen, and H. Schütze. Longform: Optimizing instruction tuning for long text generation with corpus extraction, 2023.

[25] P. Lewis, Y. Wu, L. Liu, P. Minervini, H. Küttler, A. Piktus, P. Stenetorp, and S. Riedel. PAQ: 65 million probably-asked questions and what you can do with them. *Transactions of the Association for Computational Linguistics*, 9:1098–1115, 2021. doi: 10.1162/tacl_a_00415. URL `https://aclanthology.org/2021.tacl-1.65`.

[26] X. Li, P. Yu, C. Zhou, T. Schick, L. Zettlemoyer, O. Levy, J. Weston, and M. Lewis. Self-alignment with instruction backtranslation. *ArXiv preprint*, abs/2308.06259, 2023. URL `https://arxiv.org/abs/2308.06259`.

[27] S. Longpre, L. Hou, T. Vu, A. Webson, H. W. Chung, Y. Tay, D. Zhou, Q. V. Le, B. Zoph, J. Wei, et al. The flan collection: Designing data and methods for effective instruction tuning. *ArXiv preprint*, abs/2301.13688, 2023. URL `https://arxiv.org/abs/2301.13688`.

[28] J. Miller, K. Krauth, B. Recht, and L. Schmidt. The effect of natural distribution shift on question answering models. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 6905–6916. PMLR, 2020. URL `http://proceedings.mlr.press/v119/miller20a.html`.

[29] S. Mishra, D. Khashabi, C. Baral, and H. Hajishirzi. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3470–3487, Dublin, Ireland, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.244. URL `https://aclanthology.org/2022.acl-long.244`.

[30] R. Mitkov and L. A. Ha. Computer-aided generation of multiple-choice tests. In *Proceedings of the HLT-NAACL 03 Workshop on Building Educational Applications Using Natural Language Processing*, pages 17–22, 2003. URL `https://aclanthology.org/W03-0203`.

[31] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

[32] L. Pan, Y. Xie, Y. Feng, T.-S. Chua, and M.-Y. Kan. Semantic graphs for generating deep questions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1463–1475, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.135. URL `https://aclanthology.org/2020.acl-main.135`.

[33] M. Parmar, S. Mishra, M. Purohit, M. Luo, M. Mohammad, and C. Baral. In-BoXBART: Get instructions into biomedical multi-task learning. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 112–128, Seattle, United States, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-naacl.10. URL `https://aclanthology.org/2022.findings-naacl.10`.

[34] B. Peng, C. Li, P. He, M. Galley, and J. Gao. Instruction tuning with gpt-4. *ArXiv preprint*, abs/2304.03277, 2023. URL `https://arxiv.org/abs/2304.03277`.

[35] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1264. URL `https://aclanthology.org/D16-1264`.

[36] J. Rasley, S. Rajbhandari, O. Ruwase, and Y. He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In R. Gupta, Y. Liu, J. Tang, and B. A. Prakash, editors, *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 3505–3506. ACM, 2020. URL `https://dl.acm.org/doi/10.1145/3394486.3406703`.

[37] V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv preprint*, abs/1910.01108, 2019. URL `https://arxiv.org/abs/1910.01108`.

[38] V. Sanh, A. Webson, C. Raffel, S. H. Bach, L. Sutawika, Z. Alyafeai, A. Chaffin, A. Stiegler, A. Raja, M. Dey, M. S. Bari, C. Xu, U. Thakker, S. S. Sharma, E. Szczechla, T. Kim, G. Chhablani, N. V. Nayak, D. Datta, J. Chang, M. T. Jiang, H. Wang, M. Manica, S. Shen, Z. X. Yong, H. Pandey, R. Bawden, T. Wang, T. Neeraj, J. Rozen, A. Sharma, A. Santilli, T. Févry, J. A. Fries, R. Teehan, T. L. Scao, S. Biderman, L. Gao, T. Wolf, and A. M. Rush. Multitask prompted training enables zero-shot task generalization. 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=9Vrb9D0WI4`.

[39] N. Shazeer and M. Stern. Adafactor: Adaptive learning rates with sublinear memory cost. In J. G. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 4603–4611. PMLR, 2018. URL `http://proceedings.mlr.press/v80/shazeer18a.html`.

[40] X. Shen, Z. Chen, M. Backes, and Y. Zhang. In ChatGPT we trust? Measuring and characterizing the reliability of chatgpt. *ArXiv preprint*, abs/2304.08979, 2023. URL `https://arxiv.org/abs/2304.08979`.

[41] K. Singhal, S. Azizi, T. Tu, S. S. Mahdavi, J. Wei, H. W. Chung, N. Scales, A. Tanwani, H. Cole-Lewis, S. Pfohl, et al. Large language models encode clinical knowledge. *Nature*, pages 1–9, 2023.

[42] K. Singhal, T. Tu, J. Gottweis, R. Sayres, E. Wulczyn, L. Hou, K. Clark, S. Pfohl, H. Cole-Lewis, D. Neal, et al. Towards expert-level medical question answering with large language models. *ArXiv preprint*, abs/2305.09617, 2023. URL `https://arxiv.org/abs/2305.09617`.

[43] A. Talmor, J. Herzig, N. Lourie, and J. Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. 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 4149–4158, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1421. URL `https://aclanthology.org/N19-1421`.

[44] R. Taori, I. Gulrajani, T. Zhang, Y. Dubois, X. Li, C. Guestrin, P. Liang, and T. B. Hashimoto. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`, 2023.

[45] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *ArXiv preprint*, abs/2307.09288, 2023. URL `https://arxiv.org/abs/2307.09288`.

[46] A. Ushio, F. Alva-Manchego, and J. Camacho-Collados. An empirical comparison of lm-based question and answer generation methods. *ArXiv preprint*, abs/2305.17002, 2023. URL `https://arxiv.org/abs/2305.17002`.

[47] Y. Wang, S. Mishra, P. Alipoormolabashi, Y. Kordi, A. Mirzaei, A. Arunkumar, A. Ashok, A. S. Dhanasekaran, A. Naik, D. Stap, et al. Super-natural instructions: Generalization via declarative instructions on 1600+ tasks. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022.

[48] Y. Wang, H. Ivison, P. Dasigi, J. Hessel, T. Khot, K. R. Chandu, D. Wadden, K. MacMillan, N. A. Smith, I. Beltagy, and H. Hajishirzi. How far can camels go? exploring the state of instruction tuning on open resources. *ArXiv preprint*, abs/2306.04751, 2023. URL `https://arxiv.org/abs/2306.04751`.

[49] Y. Wang, Y. Kordi, S. Mishra, A. Liu, N. A. Smith, D. Khashabi, and H. Hajishirzi. Self-instruct: Aligning language model with self generated instructions. In *Meeting of the Association for Computational Linguistics (ACL)*, 2023.

[50] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL `https://openreview.net/forum?id=gEZrGCozdqR`.

[51] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, and J. Brew. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv preprint*, abs/1910.03771, 2019. URL `https://arxiv.org/abs/1910.03771`.

[52] S. Wu, O. Irsoy, S. Lu, V. Dabravolski, M. Dredze, S. Gehrmann, P. Kambadur, D. Rosenberg, and G. Mann. BloombergGPT: A large language model for finance. *ArXiv preprint*, abs/2303.17564, 2023. URL `https://arxiv.org/abs/2303.17564`.

[53] L. Yunxiang, L. Zihan, Z. Kai, D. Ruilong, and Z. You. ChatDoctor: A medical chat model fine-tuned on LLaMA model using medical domain knowledge. *ArXiv preprint*, abs/2303.14070, 2023. URL `https://arxiv.org/abs/2303.14070`.

[54] C. Zhou, P. Liu, P. Xu, S. Iyer, J. Sun, Y. Mao, X. Ma, A. Efrat, P. Yu, L. Yu, S. Zhang, G. Ghosh, M. Lewis, L. Zettlemoyer, and O. Levy. Lima: Less is more for alignment. *ArXiv preprint*, abs/2305.11206, 2023. URL `https://arxiv.org/abs/2305.11206`.

[55] C. Ziems, W. Held, O. Shaikh, J. Chen, Z. Zhang, and D. Yang. Can large language models transform computational social science? *ArXiv preprint*, abs/2305.03514, 2023. URL `https://arxiv.org/abs/2305.03514`.

A Related Work

Instruction Tuned Models Instruction tuned language models show a remarkable ability to follow instructions and generalize to new tasks [7, 26, 27, 29, 38, 50, 54]. They are trained on large-scale training mixtures such as P3 [3] and the FLAN collection to follow instructions. In this work, we remix P3 to create task generation templates and train Bonito to generate tasks for new domains. Instruction tuning with human feedback has demonstrated strong results on open-ended generation [4, 13, 31]. More recently, several instruction tuned models [9, 22], often distilled from GPT [34, 44], have been proposed to elicit open-ended chat responses with the need for the expensive reinforcement learning training. However, numerous works show that models, including those trained with human feedback, often underperform on traditional NLP tasks [12, 31, 48]. In this work, we focus on adapting instruction tuned models without human feedback to traditional NLP tasks.

Task Generation Task generation is a fast-growing area of research to adapt large language models to follow instructions [19, 24, 44, 49]. These models condition either GPT or itself on a set of seed task demonstrations and generate new tasks [19, 49]. However, task generation conditioned on the user's unannotated data has greatly been ignored by these works. Bonito, on the other hand, can be used to create tasks with unannotated data to adapt the instruction model in new domains. Concurrent to this work, Li et al. [26] learn a conditional task generation model that uses the context to produce instructions. Our work differs in several key ways. Bonito is trained on 1.5M gold labeled data to generate tasks based on the context whereas they use a significantly smaller training dataset, i.e., 3K examples. Further, we adapt an instruction-tuned model to a new domain rather than the base model. Finally, unlike Li et al. [26], Bonito is trained with an open-source backbone model which allows wider adoption.

Knowledge Distillation Knowledge distillation is a well-studied area[17, 18, 37]. Typically, smaller models learn from the outputs of a larger model. Most recently, API-based models have been used to generate tasks and distilled into smaller models to mimic the abilities of the API-based models [14, 34]. In our work, we use Bonito to generate tasks based on the user's context and distill them into a smaller instruction tuned model for task adaptation in specialized domains.

Question Generation A range of works has been proposed in question generation over the years [25, 30, 32, 46]. These works use heuristics such as templates [30], named entity recognition [5, 25], and semantic graphs [32]. Ushio et al. [46] is the closest to this manuscript. However, they only focus on question generation and extractive question answering. In contrast, Bonito can generate high-quality tasks beyond extractive question answering.

Domain Adaptation Several works have adapted large language models to specialized domains [10, 16, 52, 53]. These models typically train on large-scale in-domain datasets [11, 15, 33, 42] or a few examples from the domain-specific task [41]. In practice, annotating training datasets for new domains is labor-intensive and expensive. In this work, we focus on generating training data for tasks in new and/or underrepresented domains.

B Hyperparameters for Training Bonito

We train Falcon-7B with a parameter-efficient learning method on the attributed task generation mixture. Following recent work on training large language models, we use Q-LoRA to train the model [12]. The model is trained for 100,000 steps with a warmup of 20,000 steps. For the rest of the hyperparameters including the learning rate, Q-LoRA rank, and trainable LoRA modules, we obtain from Dettmers et al. [12].

C Downstream Tasks: Generation, Training, Gold Data, and Evaluation

Generation Bonito is used to generate extractive question answering tasks and yes-no question answering tasks for the SQuADShifts benchmark and PubMedQA. We use nucleus sampling to generate the outputs in the vLLM framework [23]. In all datasets, we generate one task per context but we can have multiple generations per context if desired. After we get all the generations, we parse for input-output pairs by splitting at `[pipe]`. A small percentage of the generations that do not follow the parsable format are discarded. We generate extractive question answering tasks for three datasets – Reddit, Amazon, and NYT – from the SQuADShifts benchmark. We add the prefix `Task Type: question answering Fine-grained Type: extractive question answering; Context:` before the unannotated texts in the training splits to generate the desired task type. We use a top-p value of 0.95 and a temperature of 0.7 with a maximum sequence length of 128. In the same way, for PubMedQA-YN, we use the prefix `Task Type: question answering Fine-grained Type: yes-no question answering; Context:` and generate tasks on the PubMed abstracts. We use a top-p value of 0.95 and a temperature of 0.5 with a maximum sequence length of 128.

Gold Data To get the upper bound performance, we train the models with the gold training data from each of the datasets. For the extractive question answering experiments, the input is the `{context} {question}` and the output is `{answer}`. In the yes-no question answering experiment, we define five templates similar to Bach et al. [3] and randomly apply one prompt throughout the training dataset for a given seed in training. The input is the templated question and abstract and the output is either yes or no.

Training We train all the parameters of the FLAN model as a sequence-to-sequence modeling task. We set the learning rate to $1e-04$, the batch size to 16, the number of steps to 2,500, the dropout to 0.1, and the weight decay to 0.01. The validation set is used for checkpoint selection after every 100 steps. We use AdaFactor as the optimizer [39] to reduce the memory footprint. For the rest of the hyperparameters, we use the defaults from the transformers library [51]. The models are trained in a distributed multi-gpu environment with the DeepSpeed package [36]. All the models are trained on either 24GB NVIDIA GeForce 3090 or 48GB NVIDIA A40 and A6000 cards depending on their availability in the cluster.

Evaluation We evaluate the models following standard evaluation protocols for extractive question answering and yes-no question answering [35, 37]. For the extractive question answering, we use greedy decoding to generate the predictions. We report the macro average F1 that measures the average overlap between the prediction and the ground truth. For the yes-no question answering, we use ranked evaluation [37] and report the average over five prompt templates.

D Conditional Task Generation with Attributes: Datasets, Tasks, and Task Types

Table 3 lists all the datasets, task types, and prompts used in training Bonito. Question answering includes four fine-grained types: yes-no question answering, extractive question answering, multiple-choice question answering, and question answering without choices. The difference between extractive question answering and question answering without choices is that in extractive question answering the target answer is present in the context whereas in question answering without choices, that always is not the case.

Dataset	Task Type	Fine-grained Task Type	Template Name
adversarial_qa/droberta	Question generation	-	generate_question
adversarial_qa/dbert	Question answering	extractive question answering	based_on
adversarial_qa/dbidaf	Question generation	-	generate_question
adversarial_qa/dbert	Question answering	extractive question answering	question_context_answer
adversarial_qa/droberta	Question answering	extractive question answering	answer_the_following_q
adversarial_qa/dbert	Question answering	extractive question answering	tell_what_it_is
adversarial_qa/dbidaf	Question answering	extractive question answering	answer_the_following_q
adversarial_qa/dbert	Question generation	-	generate_question
adversarial_qa/dbert	Question answering	extractive question answering	answer_the_following_q
adversarial_qa/dbidaf	Question answering	extractive question answering	based_on
adversarial_qa/dbidaf	Question answering	extractive question answering	question_context_answer
adversarial_qa/droberta	Question answering	extractive question answering	tell_what_it_is
adversarial_qa/droberta	Question answering	extractive question answering	based_on
adversarial_qa/droberta	Question answering	extractive question answering	question_context_answer
adversarial_qa/dbidaf	Question answering	extractive question answering	tell_what_it_is
ag_news	Topic classification	-	classify_with_choices
ag_news	Topic classification	-	classify_question_first
ag_news	Topic classification	-	recommend
ag_news	Topic classification	-	classify_with_choices_question_first
ag_news	Topic classification	-	which_section_choices
ag_news	Topic classification	-	which_section
ag_news	Topic classification	-	classify
amazon_polarity	Sentiment	-	Is_this_review
amazon_polarity	Sentiment	-	negative_or_positive_tone
amazon_polarity	Sentiment	-	User_recommend_this_product
amazon_polarity	Sentiment	-	flattering_or_not
amazon_polarity	Sentiment	-	Is_this_review_negative
amazon_polarity	Sentiment	-	convey_negative_or_positive_sentiment
amazon_polarity	Sentiment	-	would_you_buy
amazon_polarity	Sentiment	-	user_satisfied
amazon_polarity	Sentiment	-	Is_this_product_review_positive
app_reviews	Question answering	multiple-choice question answering	categorize_rating_using_review
app_reviews	Question answering	question answering without choices	convert_to_rating
app_reviews	Text generation	-	generate_review
app_reviews	Question answering	multiple-choice question answering	convert_to_star_rating
cnn_dailymail/3.0.0	Text generation	-	generate_story
cnn_dailymail/3.0.0	Text generation	-	spice_up_story
cnn_dailymail/3.0.0	Summarization	-	news_card_view
cnn_dailymail/3.0.0	Summarization	-	news_summary
cnn_dailymail/3.0.0	Summarization	-	tldr_summary
cnn_dailymail/3.0.0	Summarization	-	sum_in_brief
cnn_dailymail/3.0.0	Summarization	-	2_or_3_sentences
cnn_dailymail/3.0.0	Summarization	-	news_stock
cnn_dailymail/3.0.0	Summarization	-	write_an_outline
cosmos_qa	Question generation	-	context_answer_to_question
cosmos_qa	Question answering	multiple-choice question answering	no_prompt_text
cosmos_qa	Question answering	multiple-choice question answering	context_description_question_answer_text
cosmos_qa	Question answering	multiple-choice question answering	description_context_question_answer_text
cosmos_qa	Question answering	question answering without choices	context_question_description_text
cosmos_qa	Question answering	question answering without choices	description_context_question_text
cosmos_qa	Question answering	multiple-choice question answering	no_prompt_id
cosmos_qa	Question answering	multiple-choice question answering	context_question_description_answer_text
cosmos_qa	Question answering	multiple-choice question answering	description_context_question_answer_id
cosmos_qa	Question answering	multiple-choice question answering	context_description_question_answer_id
cosmos_qa	Question answering	multiple-choice question answering	context_question_description_answer_id
cosmos_qa	Question answering	question answering without choices	context_description_question_text
dbpedia_14	Topic classification	-	given_list_what_category_does_the_paragraph_belong_to
dbpedia_14	Topic classification	-	pick_one_category_for_the_following_text
dream	Question answering	multiple-choice question answering	baseline
dream	Question answering	multiple-choice question answering	read_the_following_conversation_and_answer_the_question
dream	Text generation	-	answer-to-dialogue
duorc/SelfRC	Question answering	extractive question answering	movie_director
duorc/ParaphraseRC	Question answering	extractive question answering	extract_answer
duorc/ParaphraseRC	Question generation	-	generate_question_by_answer
duorc/ParaphraseRC	Question answering	extractive question answering	answer_question
duorc/SelfRC	Text generation	-	build_story_around_qa
duorc/SelfRC	Summarization	-	title_generation
duorc/SelfRC	Question answering	extractive question answering	extract_answer
duorc/SelfRC	Question answering	extractive question answering	question_answering
duorc/ParaphraseRC	Question answering	extractive question answering	question_answering
duorc/SelfRC	Question answering	extractive question answering	answer_question
duorc/SelfRC	Question generation	-	generate_question
duorc/SelfRC	Question generation	-	generate_question_by_answer
duorc/SelfRC	Question answering	extractive question answering	decide_worth_it
duorc/ParaphraseRC	Question generation	-	generate_question
duorc/ParaphraseRC	Question answering	extractive question answering	movie_director
duorc/ParaphraseRC	Summarization	-	title_generation
duorc/ParaphraseRC	Text generation	-	build_story_around_qa
duorc/ParaphraseRC	Question answering	extractive question answering	decide_worth_it
gigaword	Summarization	-	TLDR
gigaword	Summarization	-	generate_summary_for_this
gigaword	Summarization	-	write_its_sentence
gigaword	Summarization	-	first_sentence_title

Table 3: Task list

Dataset	Task Type	Fine-grained Task Type	Template Name
gigaword	Summarization	-	write_a_title_for_this_sentence
gigaword	Summarization	-	in_a_nutshell
gigaword	Text generation	-	reverse_writing
gigaword	Text generation	-	write_an_article
gigaword	Summarization	-	make_a_title
glue/mrpc	Paraphrase identification	-	replace
glue/mrpc	Paraphrase identification	-	same thing
glue/mrpc	Paraphrase identification	-	equivalent
glue/mrpc	Paraphrase generation	-	generate_paraphrase
glue/mrpc	Paraphrase generation	-	generate_sentence
glue/mrpc	Paraphrase identification	-	want to know
glue/mrpc	Paraphrase identification	-	paraphrase
hellaswag	Sentence completion	-	how_ends
hellaswag	Sentence completion	-	Open-ended completion
hellaswag	Topic classification	-	Topic of the context
hellaswag	Topic classification	-	Topic without the ending answer
hellaswag	Sentence completion	-	Randomized prompts template
hellaswag	Sentence completion	-	Predict ending with hint
hellaswag	Sentence completion	-	Open-ended start
hellaswag	Sentence completion	-	if_begins_how_continues
imdb	Sentiment	-	Movie Expressed Sentiment
imdb	Sentiment	-	Reviewer Sentiment Feeling
imdb	Sentiment	-	Writer Expressed Sentiment
imdb	Sentiment	-	Negation template for positive and negative
imdb	Sentiment	-	Reviewer Expressed Sentiment
imdb	Sentiment	-	Reviewer Enjoyment
imdb	Sentiment	-	Text Expressed Sentiment
imdb	Sentiment	-	Movie Expressed Sentiment 2
imdb	Sentiment	-	Reviewer Enjoyment Yes No
imdb	Sentiment	-	Reviewer Opinion bad good choices
paws/labeled_final	Paraphrase identification	-	Concatenation
paws/labeled_final	Paraphrase identification	-	Rewrite-no-label
paws/labeled_final	Paraphrase identification	-	Meaning
paws/labeled_final	Paraphrase identification	-	Rewrite
paws/labeled_final	Paraphrase identification	-	Meaning-no-label
paws/labeled_final	Paraphrase identification	-	context-question
paws/labeled_final	Paraphrase identification	-	context-question-no-label
paws/labeled_final	Paraphrase identification	-	task_description-no-label
paws/labeled_final	Paraphrase identification	-	PAWS-ANLI GPT3-no-label
paws/labeled_final	Paraphrase identification	-	Concatenation-no-label
paws/labeled_final	Paraphrase generation	-	paraphrase-task
paws/labeled_final	Paraphrase identification	-	PAWS-ANLI GPT3
qasc	Question answering	multiple-choice question answering	qa_with_separated_facts_1
qasc	Question answering	multiple-choice question answering	qa_with_separated_facts_3
qasc	Question answering	multiple-choice question answering	qa_with_separated_facts_2
quail	Question answering	multiple-choice question answering	no_prompt_text
quail	Question answering	multiple-choice question answering	context_question_answer_description_text
quail	Question answering	multiple-choice question answering	no_prompt_id
quail	Question answering	multiple-choice question answering	context_question_description_answer_text
quail	Question answering	question answering without choices	context_description_question_text
quail	Question answering	multiple-choice question answering	context_description_question_answer_text
quail	Question answering	question answering without choices	description_context_question_text
quail	Question answering	multiple-choice question answering	context_question_answer_description_id
quail	Question answering	question answering without choices	context_question_description_text
quail	Question answering	multiple-choice question answering	description_context_question_answer_text
quail	Question answering	multiple-choice question answering	context_question_description_answer_id
quail	Question answering	multiple-choice question answering	description_context_question_answer_id
quail	Question answering	multiple-choice question answering	context_description_question_answer_id
quoref	Question answering	extractive question answering	Given Context Answer Question
quoref	Question answering	extractive question answering	Find Answer
quoref	Question answering	extractive question answering	Answer Friend Question
quoref	Question answering	extractive question answering	Guess Answer
quoref	Question answering	extractive question answering	Found Context Online
quoref	Question answering	extractive question answering	Answer Question Given Context
quoref	Question answering	extractive question answering	What Is The Answer
quoref	Question answering	extractive question answering	Context Contains Answer
quoref	Summarization	-	Guess Title For Context
quoref	Question answering	extractive question answering	Answer Test
race/all	Question answering	multiple-choice question answering	Select the best answer (generate span)
race/all	Question answering	multiple-choice question answering	Select the best answer
race/all	Question answering	multiple-choice question answering	Select the best answer (no instructions)
race/all	Question generation	-	Write a multi-choice question for the following article
race/all	Question answering	yes-no question answering	Is this the right answer
race/all	Question answering	multiple-choice question answering	Taking a test
race/all	Question answering	question answering without choices	Read the article and answer the question (no option)
race/all	Question generation	-	Write a multi-choice question (options given)
ropes	Question answering	extractive question answering	background_situation_middle
ropes	Question answering	extractive question answering	plain_bottom_hint
ropes	Question answering	extractive question answering	background_new_situation_answer
ropes	Question answering	extractive question answering	prompt_mix

Dataset	Task Type	Fine-grained Task Type	Template Name
ropes	Question answering	extractive question answering	plain_background_situation
ropes	Question answering	extractive question answering	read_background_situation
ropes	Question answering	extractive question answering	prompt_bottom_hint_beginning
ropes	Question answering	extractive question answering	given_background_situation
ropes	Question answering	extractive question answering	new_situation_background_answer
ropes	Question answering	extractive question answering	prompt_beginning
rotten_tomatoes	Sentiment	-	Writer Expressed Sentiment
rotten_tomatoes	Sentiment	-	Reviewer Opinion bad good choices
rotten_tomatoes	Sentiment	-	Movie Expressed Sentiment
rotten_tomatoes	Sentiment	-	Reviewer Enjoyment
rotten_tomatoes	Sentiment	-	Movie Expressed Sentiment 2
rotten_tomatoes	Sentiment	-	Reviewer Sentiment Feeling
rotten_tomatoes	Sentiment	-	Reviewer Expressed Sentiment
rotten_tomatoes	Sentiment	-	Text Expressed Sentiment
rotten_tomatoes	Sentiment	-	Reviewer Enjoyment Yes No
samsum	Text generation	-	Write a dialogue that match this summary
samsum	Summarization	-	Summarize:
samsum	Summarization	-	Sum up the following dialogue
samsum	Summarization	-	Summarize this dialogue:
samsum	Summarization	-	Given the above dialogue write a summary
samsum	Summarization	-	Generate a summary for this dialogue
samsum	Summarization	-	To sum up this dialog
social_i_qa	Question answering	multiple-choice question answering	Show choices and generate index
social_i_qa	Question generation	-	Generate the question from the answer
social_i_qa	Question answering	yes-no question answering	Check if a random answer is valid or not
social_i_qa	Question answering	question answering without choices	I was wondering
social_i_qa	Question answering	multiple-choice question answering	Show choices and generate answer
social_i_qa	Question answering	question answering without choices	Generate answer
squad	Question generation	-	jeopardy
squad	Question generation	-	given_context_generate_question
squad	Question answering	extractive question answering	answer_the_question
squad	Question answering	extractive question answering	given_context_answer_question_variation
squad	Question answering	extractive question answering	answer_given_context_and_question
squad	Question answering	extractive question answering	answer_question_given_context
super_glue/wic	Word sense disambiguation	-	question-context-meaning
super_glue/wic	Word sense disambiguation	-	same_sense
super_glue/wic	Word sense disambiguation	-	GPT-3-prompt
super_glue/wic	Word sense disambiguation	-	affirmation_true_or_false
super_glue/wic	Word sense disambiguation	-	grammar_homework
super_glue/wic	Word sense disambiguation	-	question-context
super_glue/wic	Word sense disambiguation	-	similar-sense
super_glue/wic	Word sense disambiguation	-	polysemous
super_glue/wsc.fixed	Coreference resolution	-	by p they mean
super_glue/wic	Word sense disambiguation	-	question-context-meaning-with-label
super_glue/copa	Sentence completion	-	… What could happen next, C1 or C2?
super_glue/record	Question answering	extractive question answering	the placeholder refers to…
super_glue/copa	Sentence completion	-	… why? C1 or C2
super_glue/copa	Sentence completion	-	… which may be caused by
super_glue/record	Question answering	multiple-choice question answering	What could the placeholder be?
super_glue/record	Question answering	multiple-choice question answering	pick_one_option
super_glue/record	Question answering	multiple-choice question answering	trying_to_decide
super_glue/record	Question answering	multiple-choice question answering	choose_between
super_glue/boolq	Question answering	yes-no question answering	yes_no_question
super_glue/copa	Sentence completion	-	C1 or C2? premise, so/because…
super_glue/record	Question answering	extractive question answering	In the question above, the placeholder stands for
super_glue/record	Question answering	extractive question answering	exercise
super_glue/record	Question answering	multiple-choice question answering	Can you figure out…
super_glue/boolq	Question answering	yes-no question answering	I wonder…
super_glue/boolq	Question answering	yes-no question answering	could you tell me…
super_glue/boolq	Question answering	yes-no question answering	exercise
super_glue/boolq	Question answering	yes-no question answering	based on the following passage
super_glue/boolq	Question answering	yes-no question answering	after_reading
super_glue/boolq	Question answering	yes-no question answering	exam
super_glue/wic	Word sense disambiguation	-	GPT-3-prompt-with-label
super_glue/boolq	Question answering	yes-no question answering	GPT-3 Style
super_glue/boolq	Question answering	yes-no question answering	valid_binary
super_glue/copa	Sentence completion	-	i_am_hesitating
super_glue/wsc.fixed	Coreference resolution	-	Who or what is/are
super_glue/wsc.fixed	Coreference resolution	-	replaced with
super_glue/wsc.fixed	Coreference resolution	-	GPT-3 Style
super_glue/wsc.fixed	Coreference resolution	-	in other words
super_glue/wsc.fixed	Coreference resolution	-	does the pronoun refer to
super_glue/wsc.fixed	Coreference resolution	-	I think they mean
super_glue/wsc.fixed	Coreference resolution	-	p is/are r
super_glue/wsc.fixed	Coreference resolution	-	the pronoun refers to
super_glue/copa	Sentence completion	-	choose
super_glue/copa	Sentence completion	-	best_option
super_glue/copa	Sentence completion	-	more likely

Dataset	Task Type	Fine-grained Task Type	Template Name
super_glue/wsc.fixed	Coreference resolution	-	does p stand for
super_glue/boolq	Question answering	yes-no question answering	based on the previous passage
super_glue/record	Question answering	multiple-choice question answering	Which one is the placeholder?
super_glue/record	Question answering	extractive question answering	corrupted
super_glue/copa	Sentence completion	-	exercise
super_glue/copa	Sentence completion	-	cause_effect
super_glue/copa	Sentence completion	-	… As a result, C1 or C2?
super_glue/copa	Sentence completion	-	plausible_alternatives
wiki_hop/original	Question answering	multiple-choice question answering	choose_best_object_interrogative_1
wiki_hop/original	Question answering	question answering without choices	generate_subject
wiki_hop/original	Question answering	multiple-choice question answering	choose_best_object_affirmative_3
wiki_hop/original	Question answering	question answering without choices	generate_subject_and_object
wiki_hop/original	Question answering	multiple-choice question answering	choose_best_object_interrogative_2
wiki_hop/original	Question answering	question answering without choices	generate_object
wiki_hop/original	Question answering	multiple-choice question answering	choose_best_object_affirmative_2
wiki_hop/original	Question answering	question answering without choices	explain_relation
wiki_hop/original	Question answering	multiple-choice question answering	choose_best_object_affirmative_1
xsum	Summarization	-	DOC_given_above_write_one_sentence
xsum	Summarization	-	summarize_DOC
xsum	Summarization	-	college_roommate_asked_DOC_so_I_recap
xsum	Summarization	-	read_below_DOC_write_abstract
xsum	Summarization	-	DOC_write_summary_of_above
xsum	Summarization	-	DOC_how_would_you_rephrase_few_words
xsum	Summarization	-	summarize_this_DOC_summary
xsum	Summarization	-	article_DOC_summary
xsum	Summarization	-	DOC_tldr
xsum	Summarization	-	DOC_boils_down_to_simple_idea_that
yelp_review_full	Sentiment	-	format_score
yelp_review_full	Sentiment	-	based_on_that
yelp_review_full	Sentiment	-	on_a_scale
yelp_review_full	Sentiment	-	so_i_would
yelp_review_full	Sentiment	-	this_place
yelp_review_full	Sentiment	-	format_star
yelp_review_full	Sentiment	-	format_rating