Gradient Ascent Post-training Enhances Language Model Generalization

Dongkeun Yoon[1*] **Joel Jang**[1*] **Sungdong Kim**[1,2] **Minjoon Seo**[1]

[1]KAIST [2] NAVER AI Lab

mattyoon99@gmail.com, {joeljang,minjoon}@kaist.ac.kr

sungdong.kim@navercorp.com

Abstract

In this work, we empirically show that updating pretrained LMs (350M, 1.3B, 2.7B) with just a few steps of **G**radient **A**scent **P**ost-training (GAP) on random, unlabeled text corpora enhances its zero-shot generalization capabilities across diverse NLP tasks. Specifically, we show that GAP can allow LMs to become comparable to 2-3x times larger LMs across 12 different NLP tasks. We also show that applying GAP on out-of-distribution corpora leads to the most reliable performance improvements. Our findings indicate that GAP can be a promising method for improving the generalization capability of LMs without any task-specific fine-tuning [1].

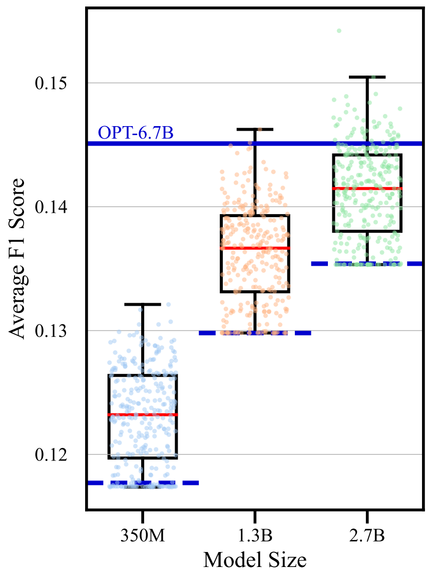


Figure 1: Average validation F1-score measured on four dialogue datasets. A single dot represents a single GAP run, each with random text samples (total of 300 runs per LM size). The dashed horizontal lines indicate performance of OPT LMs (baseline) of the same size. For reference we also show the performance of 6.7B-OPT baseline with a solid line.

1 Introduction

Recently, Language Models (LMs) pretrained on a vast amount of text corpora have shown to be capable of performing diverse downstream NLP tasks in a zero-shot manner (Brown et al., 2020; Rae et al., 2021; Chowdhery et al., 2022; Zhang et al., 2022) or through in-context learning (Brown et al., 2020; Min et al., 2022) without any gradient updates. This paradigm has been preferred over task-specific fine-tuning (Devlin et al., 2019), which requires considerable amount of labeled data for the given target task.

Motivated by the positive effect of gradient ascent during fine-tuning (Foret et al., 2021), in this work, we explore whether adapting pretrained LMs with **G**radient **A**scent **P**ost-training (GAP) on random, unlabeled text corpora can bring any benefits in terms of enhancing its generalization capabilities of performing diverse downstream NLP tasks in a zero-shot or few-shot manner *without* the need for task-specific training data.

Specifically, we apply just a few steps of gradient ascent to OPT LMs (Zhang et al., 2022) using randomly sampled text sequences from 3 different corpora from the Pile (Gao et al., 2021) with varying degree of familiarity between the LM and the corpus. Experimental results show that this simple approach achieves performance gains across 12 downstream NLP tasks: 4 dialogue tasks and 8 classification tasks. We observe that applying GAP with out-of-distribution data, specifically code data that OPT was not explicitly trained on, results in the most reliable performance gain.

Our main contributions can be summarized into two folds:

- We empirically show that GAP is a promising generalization enhancement technique as it is (1) effective, as evidenced by multiple benchmark results; (2) simple & efficient, requiring maximum 15 steps of parameter update; (3)

[*] Equal Contribution

[1]Code and full results for individual GAP runs are available at https://github.com/kaist-lklab/GAP

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 2: Short Papers, pages 851–864
July 9-14, 2023

versatile, as it can be applied easily to any pretrained LMs and does not necessitate task-specific fine-tuning.

- We show analysis of what makes GAP work by splitting the corpora into three groups according to the LMs' degree of familiarity with the data. We observe that performing GAP with the most unfamiliar (out-of-distribution) data results in the most reliable performance gain.

2 Related Works

Task-Specific Gradient Ascent Deep neural network models exhibiting poor generalization due to converging at sharp local minima is a well-known phenomenon in literature (Keskar et al., 2017; Izmailov et al., 2018; Cha et al., 2021; Chen et al., 2022). To address this issue, Foret et al. (2021) introduce Sharpness-Aware Minimization (SAM), an algorithm that performs both gradient ascent as well as gradient descent during task-specific fine-tuning to avoid sharp local minima, improving performance. The effectiveness of SAM has motivated several studies to apply them to LMs and report meaningful improvements in performance.

Bahri et al. (2022) have shown that applying SAM when fine-tuning various scales of T5 LMs (Raffel et al., 2020) on multiple downstream tasks results in a substantial performance gains. Similarly, Kaddour et al. (2022) also explore SAM across computer vision, natural language processing, and graph representation learning tasks, further bolstering its efficiency.

While SAM was proposed as a robust fine-tuning methodology that targets convergence on supervised dataset, we instead explore the benefits gradient ascent can bring *without* task-specific labeled data for generic LMs.

Task-Agnostic Gradient Ascent In a recent study, Jang et al. (2022) investigate the use of gradient ascent for addressing privacy risks in LMs. The main objective of the work is utilizing gradient ascent to *unlearn* specific token sequences; surprisingly, they report unexpected performance gains in some cases. Our work can be seen as a direct extension of this phenomenon where our main objective is to enhance the generalization capabilities instead of forgetting specific data to ensure privacy.

3 Gradient Ascent Post-training (GAP)

In this section, we give a formal definition of GAP. Specifically, given an LM with parameters w and a sequence of tokens $\boldsymbol{x} = (x_1, ..., x_N)$, GAP is defined as:

$$w_{t+1} = w_t + \alpha \nabla f_{w_t}(\boldsymbol{x}) \tag{1}$$

$$f_{w_t}(\boldsymbol{x}) = -\sum_{n=1}^{N} \log(p_{w_t}(x_n | x_{<n})) \tag{2}$$

where t represents the gradient ascent iteration, α denotes the learning rate, $x_{<n}$ indicates the token sequence $(x_1, ..., x_{n-1})$ and $p_{w_t}(x_n|x_{<n})$ represents the likelihood of predicting the next token, x_n, given the previous token sequence as an input to an LM with parameter w_t.

Markedly, GAP solely utilizes gradient ascent and does not actively facilitate convergence, as it updates the model parameters to maximize (1) the language modeling loss function (2). We propose GAP as an unsupervised methodology that can bring significant performance gains even without curated fine-tuning data.

4 Experiments

4.1 Experimental Setup

Baseline Models and Evaluation Datasets We use OPT (350M, 1.3B, 2.7B, 6.7B) LMs (Zhang et al., 2022) as the baseline LMs. We observe the effect GAP has on their generalization capabilities which is measured via evaluation on 12 different downstream NLP tasks; we use Wizard of Wikipedia (Dinan et al., 2019), Empathetic Dialogues (Rashkin et al., 2019), Blended Skill Talk (Smith et al., 2020) and WizInt (Komeili et al., 2022) to evaluate generative capabilities, Hellaswag (Zellers et al., 2019) to assess linguistic reasoning abilities, Winogrande (Sakaguchi et al., 2021) and COPA (Brassard et al., 2022) to measure commonsense reasoning abilities, and ARC-Easy (Clark et al., 2018), ARC-Challenge (Clark et al., 2018), PIQA (Bisk et al., 2020), MathQA (Amini et al., 2019) and PubmedQA (Jin et al., 2019) to measure the scientific reasoning abilities. The exact prompts used for each task are provided in Appendix A.

Random Unlabeled Data We apply GAP on text snippets from three different corpora, which all originate from the Pile (Gao et al., 2021) training set: (1) Training Data Extraction Challenge

(TDEC)[2], (2) Common Crawl (CC) and (3) Github (Git.). We choose these corpora in order to observe the effect of the LMs' degree of familiarity with the data. Training Data Extraction Challenge includes examples from the Pile that are identified to be easy-to-extract from GPT-Neo LMs (Black et al., 2022), mainly due to high levels of duplication. We assume these examples are also relatively easier-to-extract from OPT LMs as they were also pretrained on subset of the Pile, indicating the highest level of familiarity / memorization. We consider OPT LMs to be familiar (in-domain) to Common Crawl, as it was included in their pretraining corpora. As OPT LMs were not explicitly trained on the Github corpora we consider OPT to be unfamiliar (out-of-distribution) with Github. Examples of the random unlabeled data are provided in Appendix D.

Configurations For each of the 3 LM sizes [350M, 1.3B, 2.7B], we sample a total of 300 text samples (each 200 token lengths long) for applying GAP, with 100 samples taken from each of the three corpora. For each run, a single text sample is used, ultimately resulting in 300 runs of GAP per LM size. Therefore, a single epoch of a GAP run comprises of a single gradient ascent step with batch size set to 1. The number of maximum epochs is set to 15 and we report the validation score from the best-performing epoch, as preliminary experiments showed gradient ascent past 15 steps mostly resulted in performance degradation. Due to computational constraints we sample the validation data to a maximum of 320 samples per dataset for all of the 12 evaluation datasets. For further exploration of GAP as a methodology, we use the checkpoints with the best validation scores and evaluate the LMs on the test datasets for the 4 dialogue tasks. We do not separately report the test evaluation results for classification datasets since most of them require direct submission to the task website. For a single run, we use one Nvidia 40GB A100 GPU. Further details regarding the experimental configurations (e.g. optimizer, learning rate, etc.) are provided in Appendix B.

4.2 Dialogue Tasks

Main Results As shown in Figure 1 in Section 1, GAP substantially enhances the average validation performance on the 4 dialogue tasks, with median F1-score of 1.3B LMs outperforming the 2.7B LM baseline, and some 1.3B LMs even able to match the performance of the 6.7B LM baseline [3]. We report the average test F1 score as well as MAUVE (Pillutla et al., 2021), diversity (Su et al., 2022), and generation length of our best validation checkpoints for each model size (excluding outliers) in comparison to the baseline LMs in Table 1 [4].

Model	F1	MAUVE	Diversity	Length
350M	11.4	44.3	74.0	11.8
+ *GAP*	**12.5**	**67.2**	**87.3**	14.4
1.3B	13.5	48.2	82.8	11.4
+ *GAP*	**14.0**	**69.5**	**86.7**	13.8
2.7B	13.8	51.3	86.9	11.3
+ *GAP*	**14.7**	**73.0**	**93.1**	14.5
6.7B	14.5	51.1	88.3	11.9

Table 1: Average test scores on dialogue datasets. We evaluate OPT baselines and our best-performing checkpoints excluding outliers. Individual results are provided in Appendix C.

Comparison	Metric	Win	Loss	Tie
Ours vs. Baseline	C	**43%**†	17%	40%
	F	**36%**†	15%	49%
	I	**40%**†	17%	43%
Ours vs. Human	C	**41%**	37%	22%
	F	**33%**	30%	37%
	I	23%	**50%**†	27%

Table 2: Human evaluation results from dialogue generation task, WizInt (Komeili et al., 2022). The C, F, and I indicate coherence, fluency, and informativeness, respectively. † indicates the significance with p-value lower than 0.1 by bootstrap test between pairs.

Results show a substantial improvement in all of the metrics, F1 Score, MAUVE, and generation length, with our 1.3B and 2.7B LM checkpoints even outperforming the larger LM baselines. This result is significant considering that no task-specific dataset is used. Examples of text generation for the dialogue tasks are provided in Appendix E.

Human Evaluation We also evaluate and compare the qualitative quality of generated responses of the baseline LMs and the LMs adapted with GAP

[2]https://github.com/google-research/lm-extraction-benchmark

[3]Detailed numerical data for the median values is available in C.

[4]Explanation of how MAUVE and diversity is measured is provided in Appendix B.

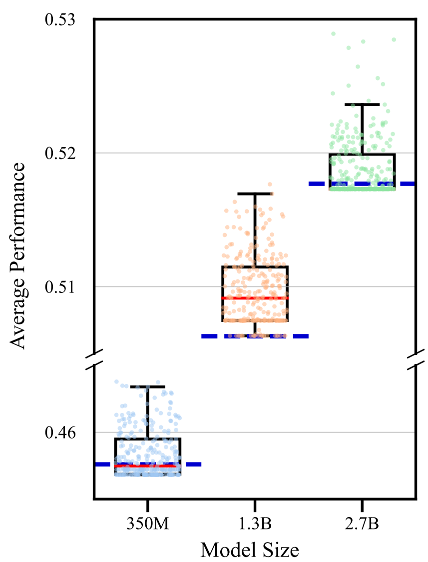


Figure 2: Average validation accuracy measured on 8 classification tasks. A single dot represents a single GAP run, total of 300 runs per LM size. The dashed horizontal lines indicate performance of baseline LMs with same size.

side-by-side. For this, we sample 100 contexts from the WizInt (Komeili et al., 2022) dataset and generate the corresponding responses with the 2.7B LM baseline and 2.7B LM + GAP denoted as *Ours*. Then, we compare the generated response pairs from the LMs from the perspective of three metrics: coherence, fluency, and informativeness (Su et al., 2022). We ask human evaluators to select the better response from each pair with respect to each metrics [5]. We find our GAP-enhanced LM shows significant strengths in all the metrics compared to its baseline (Table 2). Moreover, our LM shows comparable performance to human upper bounds (gold response) except for informativeness.

4.3 Classification Tasks

The average validation performances of the 8 classification tasks when performing GAP on the OPT LMs are shown in Figure 2. While GAP fails to provide consistent improvements for 350M LMs and 2.7B LMs, mostly resulting in a degradation of performance as shown by the median performance underperforming the baselines, the LMs show considerable performance gains in some cases for the larger LMs. This result suggests that although GAP does not show steady improvement of generalization for the classification tasks unlike the dialogue tasks, it does show some potential for improvement considering that some runs did result in substantial improvements. We leave choosing the right text samples to perform GAP on for a consistent performance enhancement on classification tasks for future work.

[5]Further study details are in Appendix F.

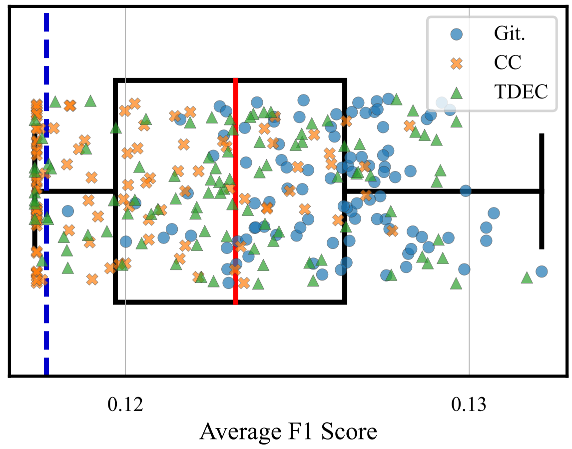


Figure 3: Average validation F1 score improvements for 350M-LMs measured on four dialogue datasets. Each symbol represents a single GAP run, with 100 runs per corpus and a total of 300 runs. The dashed vertical line indicates the performance of 350M-OPT LM.

Model	All	Git.	CC	TDEC
350M	12.3	**12.6**	11.9	12.3
1.3B	13.7	**13.8**	13.6	13.5
2.7B	14.1	**14.3**	14.2	13.9

Table 3: Average validation F1 score measured on four dialogue datasets, split into the origin of the unlabeled data. The values for **Git.**, **CC**, **TDEC** are the median value of the 100 runs for each corpus. The value for **All** is the median value of the 300 total GAP runs.

4.4 Analysis of GAP

Figure 3 shows the average performance of the 300 GAP runs for the 350M LMs (zoomed-in version of Figure 1). To observe the effect of LMs' familiarity to the unlabeled data, we plot the dots with different symbols with respect to the corpus. Interestingly, samples from the unfamiliar corpus (Github) results in significant improvements, mostly achieving higher scores than the median score. Consistent findings are also evident in Table 3, with Github achieving the highest median F1 scores across all model sizes. This suggests that future applications of GAP can be applied more efficiently by mostly using unfamiliar (out-of-distribution) text. Additional figures for other LM sizes are available in Appendix C.

5 Conclusion

In this work, we introduce GAP, a novel method of improving the generalization capability of LMs without any task-specifc data by sampling random text and performing gradient ascent for a few steps. We show that our approach is (1) simple to use, (2) effective in making more robust LMs, and (3) has much room for improvements for future work when scaling the number of GAP runs (e.g. >300) and choosing specific text samples (e.g. out-of-distribution text) to perform GAP on. Thus, we urge the community to consider GAP when prompting off-the-shelf pretrained LMs for performing diverse downstream NLP tasks.

Limitations

While we show that applying GAP can result in a significant improvement in the generalization capability of LMs, especially for dialogue tasks, we are only able to show 300 GAP runs for each LM size in this work. We leave scaling the number of GAP runs, as well as selecting *specific* text samples to perform GAP on for future work. Furthermore, a separate validation set of the tasks at interest are needed in order to choose the best checkpoint when performing GAP. Future work may look for other task-agonostic cues such as language modeling loss to determine the best checkpoint to use for inference.

Acknowledgements

This work was partly supported by KAIST-NAVER Hypercreative AI Center (80%) and Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2022-0-00113, Developing a Sustainable Collaborative Multi-modal Lifelong Learning Framework, 20%).

References

Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. MathQA: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2357–2367, Minneapolis, Minnesota. Association for Computational Linguistics.

Dara Bahri, Hossein Mobahi, and Yi Tay. 2022. Sharpness-aware minimization improves language model generalization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7360–7371.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Sidney Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, Usvsn Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. 2022. GPT-NeoX-20B: An open-source autoregressive language model. In *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*, pages 95–136, virtual+Dublin. Association for Computational Linguistics.

Ana Brassard, Benjamin Heinzerling, Pride Kavumba, and Kentaro Inui. 2022. COPA-SSE: Semi-structured explanations for commonsense reasoning. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3994–4000, Marseille, France. European Language Resources Association.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. 2021. Swad: Domain generalization by seeking flat minima. In *Advances in Neural Information Processing Systems*, volume 34, pages 22405–22418. Curran Associates, Inc.

Xiangning Chen, Cho-Jui Hsieh, and Boqing Gong. 2022. When vision transformers outperform resnets without pre-training or strong data augmentations. In *International Conference on Learning Representations*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *ArXiv*, abs/1803.05457.

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.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents. In *International Conference on Learning Representations*.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. 2021. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2021. The pile: An 800gb dataset of diverse text for language modeling.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. 2018. Averaging weights leads to wider optima and better generalization.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 2022. Knowledge unlearning for mitigating privacy risks in language models.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. 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.

Jean Kaddour, Linqing Liu, Ricardo Silva, and Matt J. Kusner. 2022. When do flat minima optimizers work?

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. 2017. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization.

Mojtaba Komeili, Kurt Shuster, and Jason Weston. 2022. Internet-augmented dialogue generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8460–8478, Dublin, Ireland. Association for Computational Linguistics.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. Mauve: Measuring the gap between neural text and human text using divergence frontiers. In *NeurIPS*.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. 2021. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*.

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

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381, Florence, Italy. Association for Computational Linguistics.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2021. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*.

Eric Michael Smith, Mary Williamson, Kurt Shuster, Jason Weston, and Y-Lan Boureau. 2020. Can you put it all together: Evaluating conversational agents' ability to blend skills. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2021–2030, Online. Association for Computational Linguistics.

Yixuan Su, Tian Lan, Yan Wang, Dani Yogatama, Lingpeng Kong, and Nigel Collier. 2022. A contrastive framework for neural text generation. In *Advances in Neural Information Processing Systems*.

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

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.

A Task Prompts

Table 4 shows the prompts we use for each of the 12 benchmark dataset to enable zero-shot/few-shot learning. For dialogue tasks (Wizard of Wikipedia, Blended Skill Talks, Empathetic Dialogues, WizInt), we use the prompts used by Zhang et al. (2022).

B Details of Experimental Configurations

In this section, we give further details of our main experimental setting of performing GAP. We use Adam optimizer (Kingma and Ba, 2014) with a constant learning rate of 5e-5 with no weight decay and no dropout.

For the dialogue tasks, we adopt the settings of Zhang et al. (2022) and prompt the LM with alternating "User 1:" and "User 2:" lines of dialogue (examples shown in Appendix A). To generate tokens, we employ greedy decoding method and set a maximum generation length of 32 tokens. For the classification tasks, we use a *verbalizer* method by selecting the output option with higher log-likelihood following Brown et al. (2020); Sanh et al. (2021). We use unigram F1 score as our main metric for the dialogue generation tasks and accuracy for the classification tasks.

For the diverse metrics used for evaluation on the test sets of the 4 dialogue tasks, MAUVE (Pillutla et al., 2021) compares the text representation of the LM generated-response to human-written text, higher values indicate greater similarity to human-written text. Diversity metric (Su et al., 2022) measures token-level repetition, with higher values indicating greater diversity and less repetition in the generated text.

C Full Results

Tables 5 and 6 show the median validation score of all 300 GAP runs. For classification tasks, the median values do not show significant improvements. However for dialogue tasks, GAP shows considerable improvements across all tasks.

Tables 7, 8, 9 and 10 show the individual test performance for each dialogue dataset. The four dialogue datasets are: Blended Skill Talks (**BST**), Empathetic Dialogues (**ED**), Wizard of Wikipedia (**WoW**) and **WizInt**. Our models demonstrate superior performance compared to their same sized baselines on every metrics in all four task.

Figures 4 and 5 represent the familiarity analysis results for 1.3B and 2.7B sized models, respectively. For both 1.3B and 2.7B models, data sampled from the out-of-domain corpora (Github) results in reliable performance gains. For the bigger sized models, in-domain corpora (CC) also results in competitive performance gains, suggesting larger sized morels are more robust to GAP data selection.

D Examples of Random Data

Table 11 shows examples of the random data we use to apply GAP to OPT LMs. Specifically, they are the best performing data for each model size.

E Examples of Dialogue Generation Outputs

Table 12 shows some examples of text generated by baseline models and our models trained with GAP. Notice that our models generate diverse and interesting text while also maintaining coherence to the given dialogue history.

F Details of Human Evaluation

We conduct the human evaluation on Amazon Mechanical Turk (AMT). An example of the interface shown to the workers is shown in Figure 6. Specifically, we recruit three different annotators for each comparison pair with a compensation of 1$ per instance. We include brief instructions on the evaluation including descriptions of three metrics. Then, we ask the workers to compare each generated (or ground-truth for human baseline) response pair with the given dialogue context. We evaluate 200 samples in total, including 100 for the OPT baseline and 100 for the human upper bounds. The Fleiss kappa among the workers is calculated as 0.36, which indicates moderate-level agreements. We also test the significance between the comparing systems via a bootstrap test with 100,000 samplings.

Table 4: Full list of the prompts used for the 12 evaluation datasets.

Dataset	Prompt
PIQA	{goal} [option]
ARC-Easy/Challenge	{question} [option]
COPA	{premise} [option]
HellaSwag	{input} [option]
Winogrande	{sentence} [option]
MathQA	{problem} [option]
PubmedQA	Question: {problem} \nAnswer: [option]
Wizard of Wikipedia, **Blended Skill Talks**, **Empathetic Dialogues**, **WizInt**	User 1: {turn}\nUser 2: {turn}\nUser 1: {turn}\n ... User 2:

Model	Avg.	BST	ED	WoW	WizInt
350M	11.77	11.88	10.17	12.05	13.00
+ *GAP*	**12.31**	**12.45**	**10.64**	**12.37**	**13.78**
1.3B	12.98	14.04	12.35	11.68	13.85
+ *GAP*	**13.60**	**14.45**	**12.58**	**12.37**	**15.02**
2.7B	13.54	13.18	12.42	12.86	**15.69**
+ *GAP*	**14.09**	**13.90**	**13.03**	**13.76**	15.65
6.7B	14.51	14.93	13.71	14.24	15.18

Table 5: **Validation F1-score** of OPT baselines and median **validation F1-score** of all GAP runs, measured on four dialogue datasets: Blended Skill Talks (**BST**), Empathetic Dialogues (**ED**), Wizard of Wikipedia (**WoW**) and **WizInt**.

Model	Avg.	ARC-Chall.	ARC-Easy	Hella-swag	MathQA	PIQA	Pubmed-QA	COPA	Wino-grande
350M	45.76	11.64	45.63	35.94	**21.88**	67.50	**54.37**	69.00	53.13
+ *GAP*	**45.84**	**19.32**	45.63	**36.88**	21.25	67.50	53.75	69.00	**53.44**
1.3B	50.63	24.07	56.25	39.38	22.81	69.38	58.44	76.00	58.75
+ *GAP*	**50.91**	**24.75**	56.25	**40.00**	**23.13**	**70.00**	58.44	76.00	58.75
2.7B	**51.77**	26.78	57.50	41.87	21.25	**72.50**	58.44	78.00	57.81
+ *GAP*	51.73	26.78	57.50	41.87	21.25	72.19	58.44	78.00	57.81
6.7B	54.39	32.20	61.87	45.63	21.25	75.94	58.44	77.00	62.81

Table 6: **Validation accuracy** of OPT baselines and median **validation accuracy** of all GAP runs, measured on classification datasets.

Model	BST	ED	WoW	WizInt
350M	11.18	10.43	13.24	10.92
+ *GAP*	**12.68**	**11.38**	**13.89**	**12.13**
1.3B	14.26	12.51	14.38	13.01
+ *GAP*	**14.83**	**12.74**	**15.18**	**13.37**
2.7B	14.00	13.09	14.40	13.58
+ *GAP*	**15.12**	**13.71**	**15.40**	**14.45**
6.7B	15.04	13.79	15.19	13.92

Table 7: Test **F1-score** of our best performing GAP models and OPT baselines on each dialogue datasets.

Model	BST	ED	WoW	WizInt
350M	48.73	31.01	53.58	43.91
+ *GAP*	**74.87**	**62.29**	**82.37**	**82.55**
1.3B	52.6	53.0	40.8	46.2
+ *GAP*	**74.7**	**54.5**	**76.4**	**72.44**
2.7B	59.8	49.4	55.4	40.6
+ *GAP*	**82.2**	**51.3**	**86.7**	**71.5**
6.7B	55.7	43.4	56.3	48.8

Table 8: Test **MAUVE** of our best performing GAP models and OPT baselines on each dialogue datasets.

Model	BST	ED	WoW	WizInt
350M	69.29	85.01	62.64	79.34
+ *GAP*	**83.22**	**91.79**	**82.96**	**91.09**
1.3B	82.62	84.43	81.07	83.23
+ *GAP*	**86.78**	**88.99**	**84.33**	**86.64**
2.7B	85.36	91.09	82.04	89.26
+ *GAP*	**93.99**	**96.22**	**89.73**	**92.38**
6.7B	86.95	92.29	81.28	92.67

Table 9: Test **diversity** of our best performing GAP models and OPT baselines on each dialogue datasets.

Model	BST	ED	WoW	WizInt
350M	10.91	10.65	13.4	12.23
+ *GAP*	**13.23**	**13.26**	**15.86**	**15.35**
1.3B	10.69	11.18	11.95	11.72
+ *GAP*	**12.89**	**12.49**	**15.05**	**14.8**
2.7B	10.4	10.72	12.39	11.58
+ *GAP*	**13.09**	**13.98**	**15.83**	**15.21**
6.7B	11.25	10.89	13.36	12.22

Table 10: Test **generation length** of our best performing GAP models and OPT baselines on each dialogue datasets.

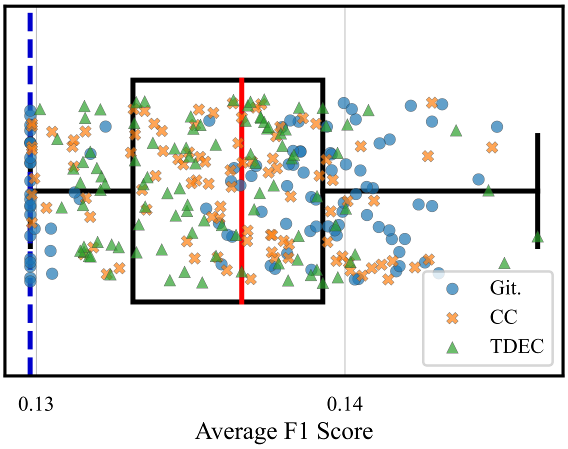


Figure 4: Analysis of average validation F1 score improvements for 1.3B-LMs measured on four dialogue datasets.

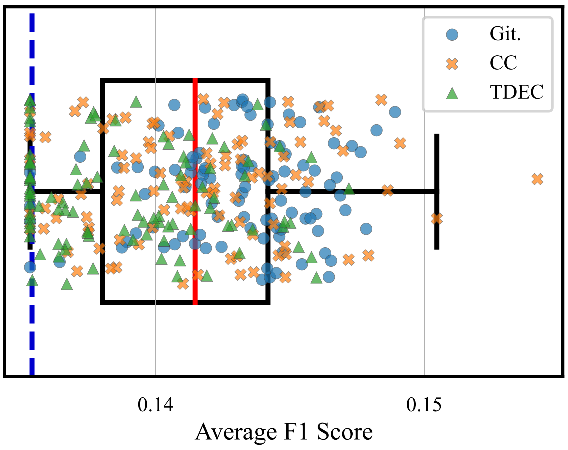


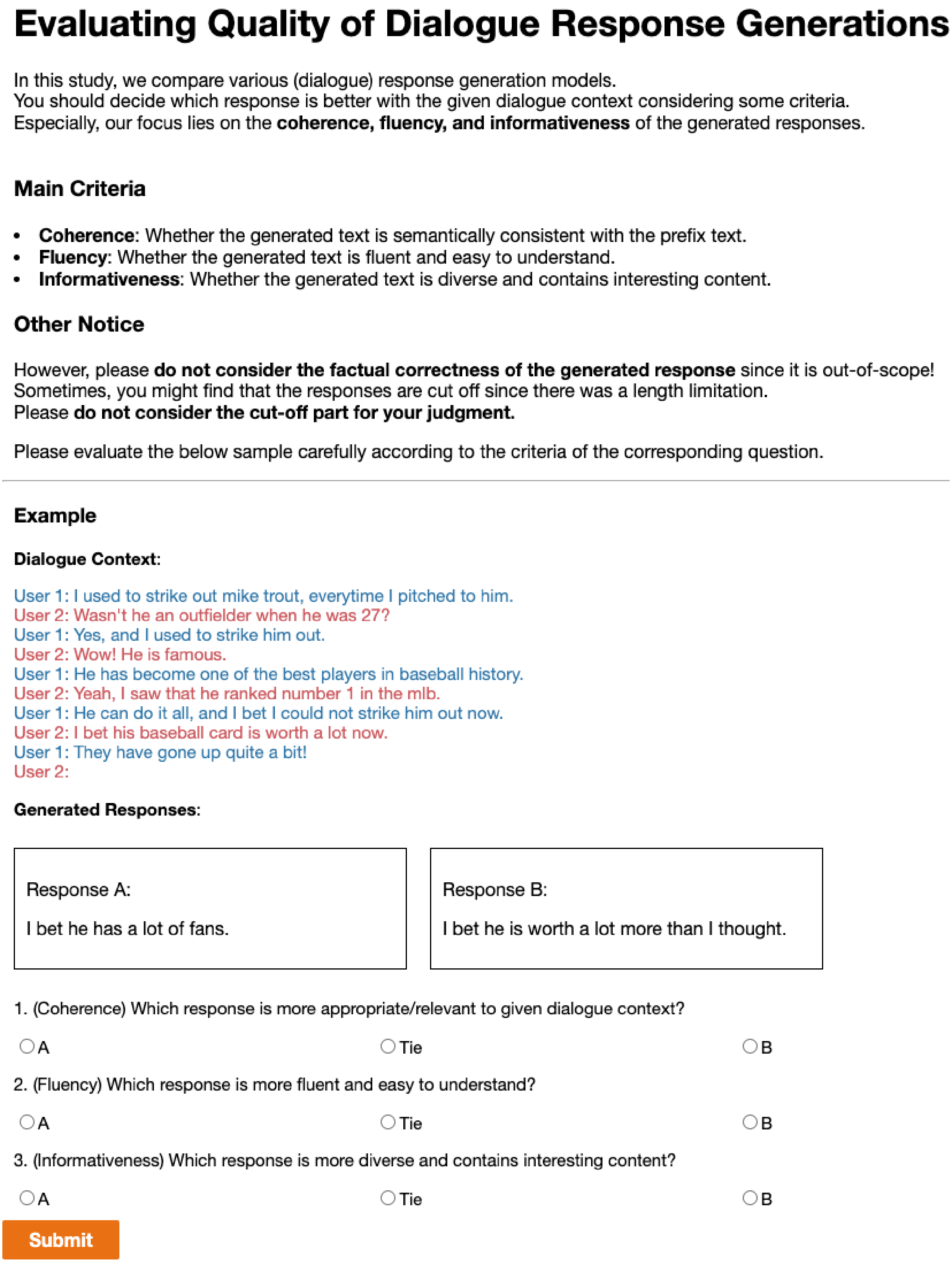
Figure 5: Analysis of average validation F1 score improvements for 2.7B-LMs measured on four dialogue datasets.

Model	Text
350M + *GAP*	"metadata": ,\n "source": [\n "Canary rollouts are used to release new models safely to only a small subset of users such as 5%. They are useful if you want to test in live production without affecting the entire user base. Since the majority of traffic goes to the existing model, the cluster size of the canary model can be relatively small since it's only receiving 5% traffic."\n]\n },\n {\n "cell_type": "markdown",\n "metadata": {},\n "source": [\n "Instead of ‘deploy()‘, we can create an ‘Endpoint Configuration‘ with multiple variants for canary rollouts and A/B testing."\n]\n },\n {\n "cell_type": "code",\n "execution_count": null,\n
1.3B + *GAP*	\n\tld d, 08\n\tld a, 10\n\tld b, 11\nlfill_vram:\n\tld(hl++), a\n\tadd a, b\n\tdec d\n\tjrnz lfill_vram\n\tld a, 90\n\tldff(45), a\n\tld a, 40\n\tldff(41), a\n\txor a, a\n\tldff(0f), a\n\tld a, 02\n\tldff(ff), a\n\tei\n\thalt\n\n.text@7000\nlprint4:\n\tld b, 90\n\tcall lwaitly_b\n\txor a, a\n\tldff(40), a\n\tld bc, 7a00\n\tld hl, 8000\n\tld d, 00\nlprint_copytiles:\n\tld a, (bc)\n\tinc bc
2.7B + *GAP*	crafts of Jharkhand. The people of the state who belong to the different ethnic groups in the state are mainly engaged in this form of craft.\n\nThe Jharkhand bamboo crafts that the tribal people of Jharkhand are engaged in show a great deal of intricate and fascinating hand work, which is quite unique to the state of India. The major articles that are made out of bamboo in Jharkhand include baskets and accessories used for fishing and hunting.\n\nThe bamboo crafts in Jharkhand that the ethnic people of the state of Jharkhand make are mostly carved out of the bamboos available locally. The variety of bamboo produced by the bamboo grooves of the state is not very thick. However, these bamboos are suitable for the different kinds of bamboo crafts at Jharkhand, since they are not weak and yet can be twisted and turned to a great extent.\n\nMetal Works of Jharkhand\n\nMetal

Table 11: Example of the best performing random data for each model size.

Model	Generation
	User 1: I know absolutely nothing about archery, but would be interested in your views on it. User 2: It is really very fun. It can be considered a sport or a skill where a bow is used to propel arrows. User 1: Hmm. Do you go to classes? It sound like such fun - I've seen it in movies but not really thought much about it. User 2: I do not. It used to be mainly for hunting and fighting, but not anymore. User 1: Perhaps I should give it a go, not for hunting but for sport.My husband does bow hunt, if that's a similar sort of thing. He seems to be quite good at it. User 2: It is an increasingly popular competitive sport and recreational activity. User 1: Interesting. Do many countries participate? It would be fun to see the ways countries differentiate. User 2:
1.3B	There are many countries that participate.
+ *GAP*	There are many countries that participate in the sport. Some countries have national competitions, while others have regional competitions.
	User 1: My friend bought a big house and I'm happy for her but at the same time I'm a little bit jealous. User 2: Big houses are a pain in the neck to keep clean. User 1: True, but there is so much space for her kid to run and play at and I live in a tiny apartment with my son. User 2:
2.7B	I'm sure she'll be happy with her new house.
+ *GAP*	I know what you mean. My house is so small that I can't even fit my son's toys in the living room.
	User 1: I am an accountant. What is your profession. User 2: pacioli established accounting in 1494 if I remember correctly ha. I work in healthcare. User 1: What is your role in healthcare. I have been an accountant for 5 years. User 2: I have an administrative role at a non-profit hospital. User 1: That is interesting. What other things will you like to tell me about your profession. User 2: I work in obtaining funding for the hospital. What type of accounting do you do User 1: I do general accounting. User 2: Lee had major impacts in the field of cost accounting. User 1: That is interesting to know. Who is lee. User 2:
2.7B	Lee was a pioneer in cost accounting.
+ *GAP*	Lee was a famous American accountant. He was the founder of the American Institute of Certified Public Accountants (AICPA).

Table 12: Examples of texts generated by baseline OPT models and our GAP applied models, given dialogue histories as prompts.



Evaluating Quality of Dialogue Response Generations

In this study, we compare various (dialogue) response generation models.
You should decide which response is better with the given dialogue context considering some criteria.
Especially, our focus lies on the **coherence, fluency, and informativeness** of the generated responses.

Main Criteria

- **Coherence**: Whether the generated text is semantically consistent with the prefix text.
- **Fluency**: Whether the generated text is fluent and easy to understand.
- **Informativeness**: Whether the generated text is diverse and contains interesting content.

Other Notice

However, please **do not consider the factual correctness of the generated response** since it is out-of-scope!
Sometimes, you might find that the responses are cut off since there was a length limitation.
Please **do not consider the cut-off part for your judgment.**

Please evaluate the below sample carefully according to the criteria of the corresponding question.

Example

Dialogue Context:

User 1: I used to strike out mike trout, everytime I pitched to him.
User 2: Wasn't he an outfielder when he was 27?
User 1: Yes, and I used to strike him out.
User 2: Wow! He is famous.
User 1: He has become one of the best players in baseball history.
User 2: Yeah, I saw that he ranked number 1 in the mlb.
User 1: He can do it all, and I bet I could not strike him out now.
User 2: I bet his baseball card is worth a lot now.
User 1: They have gone up quite a bit!
User 2:

Generated Responses:

Response A: I bet he has a lot of fans.	Response B: I bet he is worth a lot more than I thought.

1. (Coherence) Which response is more appropriate/relevant to given dialogue context?

○ A ○ Tie ○ B

2. (Fluency) Which response is more fluent and easy to understand?

○ A ○ Tie ○ B

3. (Informativeness) Which response is more diverse and contains interesting content?

○ A ○ Tie ○ B

Submit

Figure 6: An example of the Mturk interface used for the human evaluation of the dialogue response generation quality.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 6

☑ A2. Did you discuss any potential risks of your work?
Section 6

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Section 1

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

B ☒ Did you use or create scientific artifacts?

Left blank.

☐ B1. Did you cite the creators of artifacts you used?
No response.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

☐ 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)?
No response.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

☐ 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.
No response.

C ☑ Did you run computational experiments?

Section 4 and 6

☑ 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 and 6

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 and 6

☑ 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?
Section 4 and 6

☑ 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 4 and 6

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?
Section 4 and 6

☑ 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.?
Section 6

☑ 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)?
Section 4 and 6

☑ 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?
Section 4 and 6

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☒ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
We weren't able to obtain the information because Amazon Mechanical Turk does not provide the information.