Following Length Constraints in Instructions

Weizhe Yuan[1,2] **Ilia Kulikov**[1] **Ping Yu**[1] **Kyunghyun Cho**[2]
Sainbayar Sukhbaatar[1] **Jason Weston**[1,2] **Jing Xu**[1]

[1]Meta FAIR [2]New York University

Abstract

Aligned instruction following models can better fulfill user requests than their unaligned counterparts. However, it has been shown that there is a length bias in evaluation of such models, and that training algorithms tend to exploit this bias by learning longer responses. In this work we show how to train models that can be controlled at inference time with instructions containing desired length constraints. Such models are superior in length instructed evaluations, outperforming standard instruction following models such as GPT4, Llama 3 and Mixtral.

1 Introduction

Instruction following has emerged as one of the most important topics in AI (Ouyang et al., 2022; Touvron et al., 2023). In current evaluations both humans and models tend to have a "length bias" whereby they prefer longer responses (Dubois et al., 2024b). Correspondingly, training methods that follow these preferences tend to produce longer responses (Singhal et al., 2023). Some benchmarks incorporate length penalties into their scoring to counteract this bias (Dubois et al., 2024a), but this does not fix the problem at its source.

We argue that the expected length of responses is ill-defined in many queries, and this ambiguity makes evaluation difficult, which in turn affects training algorithms that use these evaluation signals. Typical requests can be ambiguous in terms of the desired response length, e.g. without context the answer to *'Give me information about Coco Gauff"* could be a few sentences, or a multi-page document. To resolve this we propose that evaluation should include *further disambiguating instructions* that prescribe the length of the desired response.

We show that many existing state-of-the-art instruction following models fail to follow such maximum word length instructions adequately. To measure this we construct and evaluate models on length instructed versions of AlpacaEval 2 (Dubois et al., 2024b) and MT-Bench (Zheng et al., 2023) by augmenting existing prompts with length instructions. We find that, for example, GPT4-Turbo violates length constraints almost 50% of the time, highlighting a significant flaw in these models when it comes to steering their output length.

We hence develop a method for improving models at length instruction following. Our approach, Length-Instruction Fine-Tuning (LIFT), involves augmenting an instruction following dataset by inserting length instructions in the original prompts. We define length instructions so that the augmented preference pairs reflect both length constraints and response quality. We then finetune models on this length instruction augmented dataset via Direct Preference Optimization (DPO) (Rafailov et al., 2023). We train both Llama 2 and Llama 3 models using LIFT-DPO and evaluate them on standard and our length instructed benchmarks. We find that our method leads to fewer length constraint violations and improved overall win rates compared to existing instruction following models.

2 Related Work

Length Bias in Model Alignment Reinforcement learning (RL) has been consistently observed to encourage models to produce longer responses (Zhao et al., 2024). This effect seen in training parallels that on the evaluation, whereby longer responses are more preferred(Dubois et al., 2024b), even though not necessarily better (Park et al., 2024; Achiam et al., 2023; Casper et al., 2023).

Length-aware Training Existing approaches e.g. balancing preferences (Singhal et al., 2023), disentangling length from quality (Shen et al., 2023; Chen et al., 2024), adding a length regularizer to the loss (Park et al., 2024), all assume optimum lengths in training. In contrast, our work assumes desired length depends on additional context.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 24232–24243

	AlpacaEval-LI			MT-Bench-LI		
Standard models	Vlt(%)	Win(%)	Words	Vlt(%)	Win(%)	Words
GPT4 Omni (gpt-4o-2024-05-13)	39.0	35.7	180	39.2	30.2	177
GPT4 Turbo (gpt4_1106_preview)	46.1	29.9	182	45.0	28.1	174
GPT4 Turbo (gpt-4-turbo-2024-04-09)	49.3	29.2	187	44.2	27.5	179
Claude 3 Opus (02/29)	37.0	32.9	183	37.9	33.1	174
Mistral Large (24/02)	17.6	28.8	158	20.8	27.7	158
Llama3-70B-Instruct	10.2	38.5	154	20.3	28.5	151
Llama3-8B-Instruct	7.0	22.5	145	20.0	20.0	140

Table 1: **Length Instruction-Following results of SOTA models on the AlpacaEval-LI + MT-Bench-LI benchmarks.** Many SOTA LLMs have large violation rates (Vlt(%)) as they fail to follow length instructions.

3 AlpacaEval-LI & MT-Bench-LI: New Length-Instructed Benchmarks

3.1 Augmenting General Instructions

To test whether models can follow the given length instruction, we augment existing instructions with maximum length limits, with the template "***Answer the following instruction using <MAX_LEN> words or less.***`\n\n`***<ORIGINAL_INSTRUCTION>***". We set *<MAX_LEN>* to the minimum generation length among GPT-4 Turbo(11/06) , Claude 3 Opus(02/29) and Mistral Large(24/02) on the original prompts. This constraint varies for each individual prompt, and is not trivially satisfied.

To establish a strong baseline, we employ the same minimum of three models approach. This ensures the baseline generations always meet the length constraint while maintaining high quality.

3.2 Metrics

Length Instruction Following We use violation rates (Vlt%) to measure the percentage of responses that violate the length constraint.

Response Quality We report pairwise winrates comparing model and baseline generations on length-following instructions, referred to as the *Length-Instructed (LI) Winrate*. We treat the length limit as a hard constraint. Since the baseline always satisfies the length constraint, if the model response being tested exceeds the limit it automatically loses. If the model response satisfies the length limit, we use the standard pairwise comparison.

3.3 AlpacaEval-LI & MT-Bench-LI

Following Section 3.1, we augment AlpacaEval (Dubois et al., 2024b) with length instructions to create *AlpacaEval-Length-Instructed (LI)*. We exclude three out of the 805 Alpaca test instructions with explicit length constraint. For MT-Bench (Zheng et al., 2023), we sample three length limits for each prompt. For simplicity we only consider first turns, giving 240 MT-Bench-LI prompts.

4 Length-Instruction Fine-Tuning (LIFT)

To improve models' length-instruction following ability, we propose the following LIFT method.

Given a pairwise preference dataset $\mathcal{D} = (x, y_i^w, y_i^l)$, let $\texttt{len}(y)$ be the number of words in response y. First, we filter out any triple with difference between $\texttt{len}(y_i^w)$ and $\texttt{len}(y_i^l)$ less than T ($T = 10$ in our experiments). We then prepends an explicit length instruction to the input x_i using the template to form x_i' and construct an augmented dataset $\mathcal{D}' = (x_i', y_i^{w'}, y_i^{l'})$ as follows:

If $\texttt{len}(y_i^w) > \texttt{len}(y_i^l)$: i.e. the winning response is longer, we construct two samples in $\mathcal{D}'$ by, (1) adding a length instruction to x_i that both responses satisfy (we use $\texttt{len}(y_i^w) + T$) and the winning response and losing response remain the same, and (2) adding a length constraint uniformly sampled from the interval $\left[\texttt{len}(y_i^l), \texttt{len}(y_i^w)\right]$, and y_i^w becomes the losing one due to the violation of length constraint, and y_i^l becomes the winning one.

If $\texttt{len}(y_i^w) < \texttt{len}(y_i^l)$: We construct two samples in $\mathcal{D}'$ by, (1)setting length constraint to $\texttt{len}(y_i^l) + T$ that both responses satisfy, (2) sampling a length constraint from the interval $\left[\texttt{len}(y_i^w), \texttt{len}(y_i^l)\right]$. Here, the winning and losing responses remain the same as original.

Our LIFT method can be used to augment instructions with other types of disambiguated constraints, see Appendix L for results on applying LIFT with lower bound length limits. LIFT-augmentation help models prioritize the length constraints over the original preferences when necessary. We train models on both D and D' to handle prompts with and without length instructions.

	AlpacaEval-LI			MT-Bench-LI		
	Vlt(%)	Win(%)	Words	Vlt(%)	Win(%)	Words
Llama2-70B-Base + DPO	65.8	4.6	216	60.8	5.0	199
Llama2-70B-Base + R-DPO (Park et al., 2024) ($\alpha = 0.1$)	45.0	7.7	178	39.4	8.5	161
Llama2-70B-Base + LIFT-DPO	**7.1**	**13.6**	151	**10.0**	**11.0**	146
Llama2-70B-Chat	28.2	11.3	162	38.3	11.9	168
Llama2-70B-Chat + DPO	15.1	10.4	135	24.2	10.8	147
Llama2-70B-Chat + LIFT-DPO	**2.7**	**14.2**	140	**6.7**	**12.5**	135

Table 2: **Llama 2 Length Instruction-Following results on the AlpacaEval-LI + MT-Bench-LI benchmarks.** LIFT-DPO yields improved winrates (Win(%)) and lower length instruction following violation rates (Vlt(%)).

	AlpacaEval-LI			MT-Bench-LI		
	Vlt(%)	Win(%)	Words	Vlt(%)	Win(%)	Words
Llama3-8B-Base + DPO	58.1	5.0	202	50.8	7.7	191
Llama3-8B-Base + LIFT-DPO	**6.1**	**11.1**	153	**13.8**	**12.9**	152
Llama3-8B-Instruct	7.0	22.5	145	20.0	20.0	140
Llama3-8B-Instruct + DPO	7.1	25.1	143	21.3	20.0	142
Llama3-8B-Instruct + LIFT-DPO	**3.1**	**25.6**	161	**10.8**	**26.3**	157

Table 3: **Llama 3 Length Instruction-Following results on the AlpacaEval-LI + MT-Bench-LI benchmarks.** LIFT-DPO yields improved winrates (Win(%)) and lower length instruction following violation rates (Vlt(%)).

5 Experimental Setup

We empirically investigate the effectiveness of our LIFT training strategy on: Llama2-70B-Base and Llama2-70B-Chat models (Touvron et al., 2023) and Llama3-8B-Base and Llama3-8B-Instruct.

Standard Training Data Following Li et al. (2024) we use 3,200 first-turn human-authored English examples from OpenAssistant (OA) dataset (Köpf et al., 2023) as $\mathcal{D}$, that are high-quality based on their human annotated rank (choosing only the highest rank 0 as chosen and rank 1 as loser). We do supervised finetuning (SFT) on the chosen responses of $\mathcal{D}$, then fine-tune the SFT model using the DPO loss on $\mathcal{D}$ as our *Standard DPO* baseline. In addition, we also compare against the Length Regularized DPO (R-DPO) (Park et al., 2024). See DPO training details in Appendix F.

Length-Instructed Fine-Tuning (LIFT) Data We apply our LIFT method to create dataset $\mathcal{D}'$ from $\mathcal{D}$, which yields 5,954 preference pairs with length instructions. We train on $\mathcal{D} \cup \mathcal{D}'$ with the DPO loss, which we call *LIFT-DPO*.

6 Experimental Results

We report performances of SOTA models in Table 1, and our LIFT-DPO results in Table 5 and Table 3. We observe several key findings.

SOTA LLMs fail to follow length instructions In Table 1, SOTA models such as the GPT-4 series exhibit significant challenges in adhering to length instructions. Specifically, the GPT-4 Turbo (0409) shows a high violation rate of 49.3% on our AlpacaEval-LI and 44.2% on MT-Bench-LI.

LIFT-DPO models perform well on AlpacaEval-LI and MT-Bench-LI In Table 5, LIFT-DPO dramatically reduces violation rates compared to the (standard) DPO, from 65.8% to 7.1% on AlpacaEval-LI with the Llama-2-70B-Base model, while improving win rate from 4.6% to 13.6%. For the Llama-2-70B-Chat model, our LIFT-DPO reduces violation from 15.1% to 2.7%, and enhances win rate from 10.4% to 14.2%. On MT-Bench-LI, LIFT-DPO also reduced violation rate while boosting the win rates for both Llama2 and 3 models. While the R-DPO baseline improves over standard DPO on both benchmarks, it still shows significantly higher violation rates compared to LIFT-DPO, which negatively affects R-DPO's win rates. In Appendix M, we show that SFT is not enough for teaching models to follow length instructions.

LIFT-DPO models show no performance degradation when length instructions are not applied On the standard AlpacaEval benchmark, detailed in Table 6, the win rates improved from 12.6% using DPO to 12.9% with LIFT-DPO for the Llama-2-70B-Chat model. The LC winrate increased from 13.9% to 15.7% for Llama-3-8B-Base models, and from 26.3% to 26.5% for the Llama-3-8B-Instruct models with LIFT-DPO. Similar results are ob-

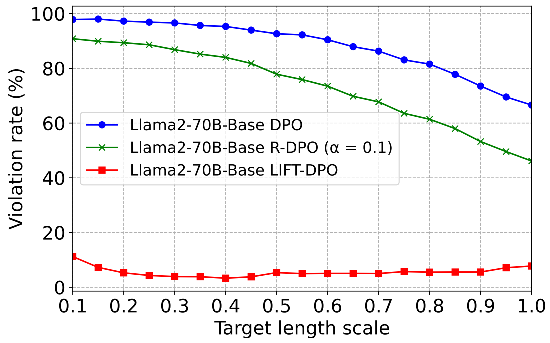

Figure 1: The violation rate of DPO or R-DPO Llama2-70B models on AlpacaEval-LI increases as the target length shortens. However, LIFT-DPO consistently maintains a low violation rate independent of length scale.

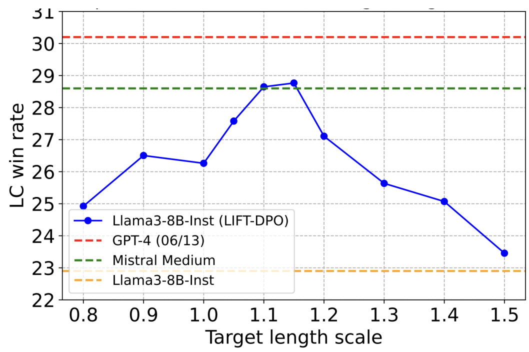

Figure 2: AlpacaEval 2 LC winrate vs target length scale. Our LIFT-DPO Llama-3-8B-Instruct model can be controlled to produce different length responses, which affects overall LC winrate.

served on standard MT-Bench in Appendix Table 8.

LIFT-DPO can follow out-of-distribution length instructions better than existing methods To increase the difficulty of AlpacaEval-LI, we reduce the length limit using a scaling factor from 0.9 to 0.1, and assessed the violation rates of standard DPO, R-DPO and LIFT-DPO in Figure 1. The standard DPO model exhibits increasingly higher violation rates escalating from below 50% to almost 100% when the scale factor is 0.1, indicating significant difficulties in adhering to stringent length constraints. The R-DPO model displays similar trends, suggesting that while it can reduce the generation length, it lacks the capability to precisely steer it. In contrast, our LIFT-DPO model consistently maintains a low violation rate (below 10%) across all tested length scales. Similar trends on MT-Bench-LI are observed in Appendix Figure 8. Furthermore, we assess the LIFT-DPO models on math tasks to validate their ability to follow length in out-of-domain settings (see Appendix N).

LIFT-DPO can follow various natural length prompts better than existing methods We evaluated our LIFT-DPO models on IFEval (Zhou et al., 2023), which includes instructions with verifiable constraints such as "write a 400+ words essay". Our aim was to assess the robustness of our models on natural length prompt templates and length limits beyond what we used in our training process. IFEval consists of 541 "verifiable instructions", of which 44 contain upper bound length constraints. We found that LIFT-DPO achieved significantly lower violation rates on max word constraints and max sentence constraints compared to normal DPO. Detailed results are provided in Appendix K.

Robustness of Length Controlled AlpacaEval Previous research has acknowledged the presence of length bias, and introduced measures to mitigate it, notably through Length-Controlled (LC) AlpacaEval. Nevertheless, we find that the LC winrate can still be manipulated. By scaling the length constraints as we did in AlpacaEval-LI and measuring the AlpacaEval LC winrate, we observe significant fluctuations in the LC winrate from 23% up to 29%, as shown in Figure 2. In contrast, we argue that expected length is ill-defined in many queries (see motivation in Section 1), and that length instruction evaluation helps remove this ambiguity, and hence also any potential gameability.

7 Conclusion

To address the length bias in general instruction following, we propose length instructions, which assess models' abilities to generate responses within given length limits. We introduce two Length-Instructed (LI) benchmarks, MT-Bench-LI and AlpacaEval-LI, and show that SOTA models surprisingly fail to follow length instructions on these benchmarks. We hence propose Length-Instruction Fine-Tuning (LIFT), a method that augments existing general instruction-following examples with varying length limits. LIFT-DPO models show significant improvement in controlling output length while maintaining high response quality. Our length instruction following approach provides a way to compare models without length bias, as it does not suffer from the gameability of simply increasing model response length, as that leads to a violation. In addition, augmenting general instructions with length limits allows for more controllability for users in real-world use cases.

8 Limitations

In this paper, the length limit is set in terms of the number of words. Our results indicate that training on word limits can also transfer to improved adherence to length limits in number of sentence, but more generally length limit can be set in other measure, such as number of characters. The promising results with LIFT-DPO, which follows various length prompt templates despite being trained on a fixed one, suggest a potential avenue for further generalization. Length instructions could be phrased more flexibly, allowing users to specify limits in their own words, such as "Keep the response under 100 words.". We also did not address other kinds of length instructions such as "write 100-300 words". Given promising results showing how LIFT-DPO can improve models' abilities on both lower and upper length limits, future work could adapt our LIFT method to other more complex length instructions. While this paper attempts to address length bias in model evaluations through length instructions, this bias may also arise from a natural human preference for longer and more detailed responses. Future research could further explore human desired response lengths across different instructions. Such studies could further enhance the alignment of models with human expectations. Another possible cause of longer responses could be related to the increased computation allowance that comes with more tokens, which can benefit from future analysis.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. 2023. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*.

Lichang Chen, Chen Zhu, Davit Soselia, Jiuhai Chen, Tianyi Zhou, Tom Goldstein, Heng Huang, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Odin: Disentangled reward mitigates hacking in rlhf. *arXiv preprint arXiv:2402.07319*.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. 2024a. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*.

Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S Liang, and Tatsunori B Hashimoto. 2024b. Alpacafarm: A simulation framework for methods that learn from human feedback. *Advances in Neural Information Processing Systems*, 36.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi-Rui Tam, Keith Stevens, Abdullah Barhoum, Nguyen Minh Duc, Oliver Stanley, Richárd Nagyfi, et al. 2023. OpenAssistant conversations–democratizing large language model alignment. *arXiv preprint arXiv:2304.07327*.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Luke Zettlemoyer, Omer Levy, Jason Weston, and Mike Lewis. 2024. Self-alignment with instruction backtranslation. In *The Twelfth International Conference on Learning Representations*.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Alpacaeval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Ryan Park, Rafael Rafailov, Stefano Ermon, and Chelsea Finn. 2024. Disentangling length from quality in direct preference optimization. *arXiv preprint arXiv:2403.19159*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Wei Shen, Rui Zheng, Wenyu Zhan, Jun Zhao, Shihan Dou, Tao Gui, Qi Zhang, and Xuanjing Huang. 2023. Loose lips sink ships: Mitigating length bias in reinforcement learning from human feedback. *arXiv preprint arXiv:2310.05199*.

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. 2023. A long way to go: Investigating length correlations in rlhf. *arXiv preprint arXiv:2310.03716*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Answer the following instruction using <MAX_LEN> words or less.

<ORIGINAL_INSTRUCTION>

Figure 3: **Length Instruction Following.** We define the above prompt template in order to require models to produce responses within a maximum response length.

Hao Zhao, Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. 2024. Long is more for alignment: A simple but tough-to-beat baseline for instruction fine-tuning. *arXiv preprint arXiv:2402.04833*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

A More Details on LI Evaluation and Training.

Standard AlpacaEval 2 compares model against baseline GPT-4 Preview (1106). In AlpacaEval-LI, the baseline is built from GPT4-1106, Claude3-Opus and Mistral Large as described in Section 3.1. Their respective winrates in the standard AlpacaEval 2 are 50%, 40.5% and 32.7%. This indicates that the resulting baseline is of high quality while consistently meeting the length constraint. For AlpacaEval-LI, We exclude three out of the 805 Alpaca test instructions which already have an explicit length constraint in the original prompt.

Figure 4 shows the ratio of generation lengths over target instruction lengths as target lengths vary. GPT4-0409 generations exceed the target length limits almost 50% of the time (red dots), especially when target lengths are over 200 words. Claude3-Opus has a similar trend according to the scatter plot. We also include results for Mistral Large and LLAMA3-70b-Instruct in Appendix D.

B Examples of LIFT-DPO model responses on length instructions

C Word Count Function We Use

```python
from nltk.tokenize import word_tokenize
import string

def count_words(text) -> int:
    # Count the number of words
    # while excluding punctuations
    return len([word for word in word_tokenize(text) if word not in string.punctuation])
```

D Additional Results on SOTA models' length following measurements

We plot the generation lengths over target instruction lengths on AlpacaEval-LI for Mistral Large and LLAMA3-70b-Instruct in Figure 6. The scatter plots reveal that both models occasionally fail to meet the length constraints.

E Training and test length distribution

The original dataset $\mathcal{D}$ consists of 223 pairs where the two responses have less than $T = 10$ words difference, 1,083 pairs where chosen responses are shorter than loser responses, and 1894 pairs where chosen responses are longer. As a result, $\mathcal{D}'$ contains 1,083 pairs where the original winning response loses due to violations of length limits.

Figure 7 illustrates the distribution of length constraints in our LIFT-DPO training data alongside those in AlpacaEval-LI and MT-Bench-LI. We observed that the majority of our training data features length constraints ranging from 50 to 300, a range that is consistent with that of AlpacaEval-LI. Additionally, we have depicted the distribution of length constraints in AlpacaEval-LI scaled by a factor of 0.1 in Figure 7. Nearly all scaled length constraints fall below 50, constituting only a small fraction of the length constraints present in our training dataset.

F DPO training details

Our DPO training sweeps over a range of learning rates $5e^{-7}$ to $5e^{-6}$ with a cosine learning rate schedule, a batch size of 16, and a dropout rate of 0.1. Specifically for DPO training, we set $\beta = 0.1$. For R-DPO, we set $\alpha \in [0.01, 0.1]$. We had to reverse the sign of the regularization term in Eq. 9 of Park et al. (2024). All Llama2 models are trained for up to 2,000 steps and Llama3 models for up to 20 epochs, and we perform checkpoint selection for early stopping

We perform checkpoint selection by saving a checkpoint every 200 steps and at the end of each

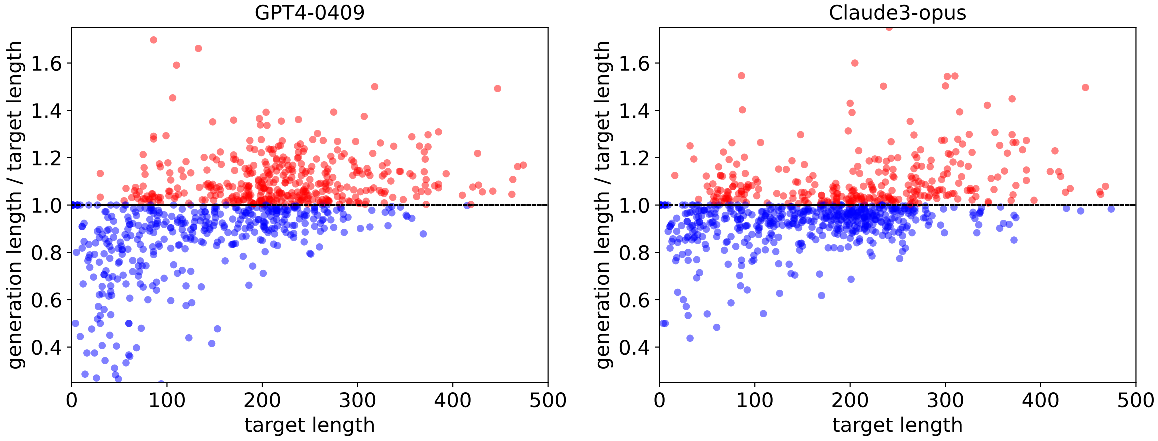

Figure 4: **SOTA Models fail to follow length instructions**. Length instruction following of GPT4-0409 and Claude3-Opus on 802 AlpacaEval-Length-Instructed (LI) examples. The target length is plotted on the x-axis and the ratio of the actual generated length to the target length on the y-axis. Red dots represent violations where the generated length exceeds the target limit, while blue dots satisfy the limit.

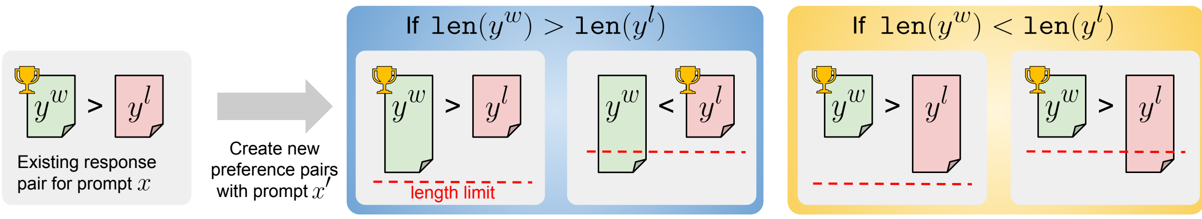

Figure 5: Length-Instruction Fine-Tuning (LIFT) method for augmenting preference pairs (x, y^w, y^l) in general instruction-following tasks with length instructions. The original prompt x is prepended with an extra instruction specifying a target response length, yielding a new prompt x'. The target length is chosen in multiple ways, creating extra preference pairs. In some cases the original winning response y^w becomes the loser if it exceeds the limit.

Instruction	<MAX_LEN>	**Generation**	**Length**
Answer the following instruction using <MAX_LEN> words or less. I'm trying to teach myself to have nicer handwriting. Can you help?	20	Yes, I'd be happy to help you improve your handwriting skills!	12
	40	I'd be happy to help. To improve your handwriting, practice regularly, focus on posture and grip, and try using guidelines or a light box to help you stay within lines.	31
	80	I'd be happy to help you improve your handwriting. Here are a few tips to get you started: * Choose the right tools: Use a pen with good grip and smooth ink flow. Experiment with different types of paper to find one that works well for you. * Practice regularly: Set aside time each day to practice writing. * Focus on posture and grip: Keep your hand steady and your fingers relaxed.	70
I'm trying to teach myself to have nicer handwriting. Can you help?	N/A	Improving your handwriting takes practice, patience, and some guidance. I'd be happy to help you with that. Here are some tips to get you started: 1. **Choose the right tools**: Invest in a few good quality pens or pencils that feel comfortable in your hand. Experiment with different types of pens (e.g., fine tip, medium tip, calligraphy) to find what works best for you. 2. **Practice basic strokes**: Focus on mastering basic strokes such as:	456

Table 4: **Length-Instructed example generations.** We show examples from our Length-Instruction Fine-Tuned (LIFT) Llama-3-8B-Instruct model with different length instruction limits for the same question. The last row is a response generation using the original input without length instructions (partial generation due to limited space). Many state-of-the-art LLMs are unable to follow such length instructions, see Figure 4.

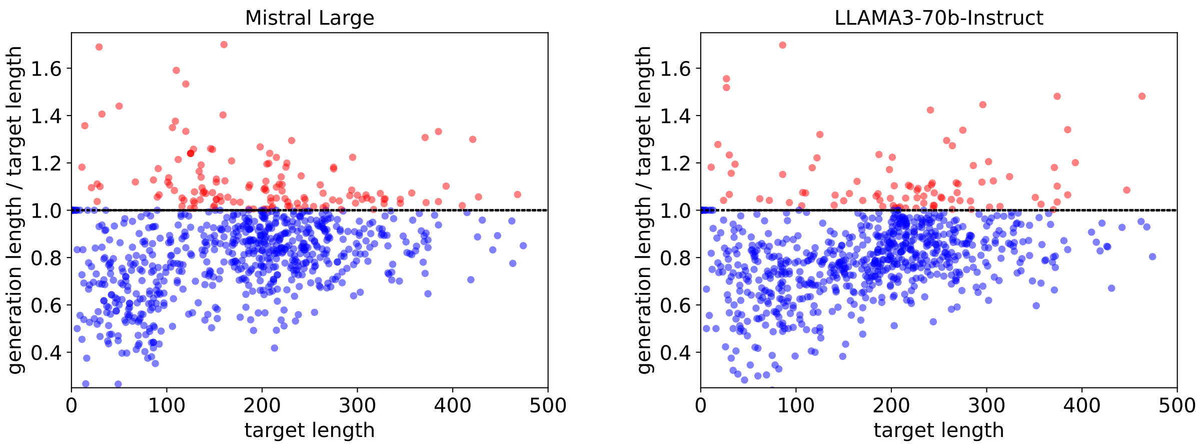

Figure 6: The length instruction following ability of Mistral Large and LLAMA3-70b-Instruct on 802 AlpacaEval Length-Instructed (LI) examples. The scatter plots display each sample from the AlpacaEval LI dataset, with the target length plotted on the x-axis and the ratio of the actual generated length to the target length on the y-axis. Red dots represent violations where the generated length exceeds the target limit, while blue dots satisfy the limit.

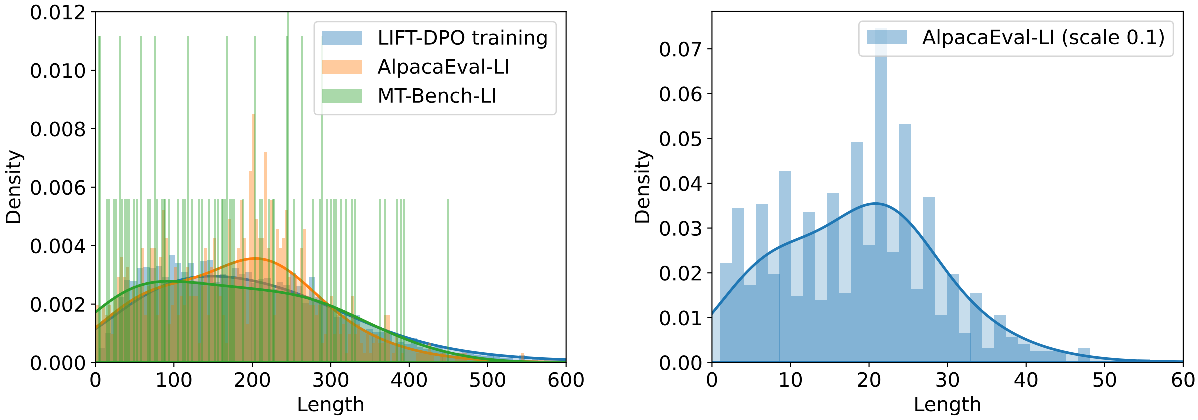

Figure 7: The distribution of length constraints across LIFT-DPO training data, AlpacaEval-LI, and MT-Bench-LI. Additionally, we also include a plot of the AlpacaEval-LI length constraints scaled by a factor of 0.1.

epoch. We then evaluate these checkpoints using GPT-4-Turbo on a set of 253 validation examples, which are derived from various sources as outlined by Li et al. (2024). The LI (Length-Instructed) validation set is augmented from the same validation set but includes length limits, using the minimum length from three strong LLMs in Section 3.1.

For the standard instruction-following validation set, each new model checkpoint is evaluated by comparing its generations pairwise with those from the previous checkpoint, utilizing the AlpacaEval evaluation prompt format (Li et al., 2023). For length-instructed tasks, evaluations are conducted pairwise against a baseline from one of the three LLMs, specifically the one whose generation length matches the length limit specified in the prompt. The win rate of a model checkpoint is calculated as the average of the win rates on both the instruction-following validation set and the LI validation set. We implement early stopping if we observe a decrease in this average win rate.

G More Results

SOTA LLMs fail to follow length instructions As demonstrated in Table 1, SOTA models such as the GPT-4 series exhibit significant challenges in adhering to length instructions. Specifically, the GPT-4 Turbo (0409) shows a high violation rate of 49.3% on our AlpacaEval-LI and 44.2% on MT-Bench-LI. The Llama-3 instruct model series displays considerably lower violation rates. For instance, the Llama3-8B-Instruct model achieves a violation rate of 7.0% on AlpacaEval-LI and 20.0% on MT-Bench-LI, but nevertheless has a lower winrate due to being a less powerful model.

In the standard MT-Bench evaluation, models employ different temperatures (including 0) for different categories during inference time. To expand the size of MT-Bench-LI via sampling, we standardized the temperature setting to 0.7 across all categories for pairwise baseline models as well as models being tested. However, for the standard MT-Bench evaluation reported in Table 8, we switch back to the original setup using different temperatures for different categories and assessing performance on 80 unique questions.

H Decoding Parameters

During inference time, except for the standard MT-Bench evaluations, we apply consistent hyperparameter settings for the Llama models. For the Llama2 models, we set the temperature to 0.7, with a maximum token limit of 2048. For the Llama3 models, the temperature is adjusted to 0.6, maintaining the same top-p of 0.9, but with an increased maximum token limit of 4096. We consistently set top-p to 0.9 for AlpacaEval 2 and AlpacaEval-LI and top-p to 1.0 for MT-Bench and MT-Bench-LI.

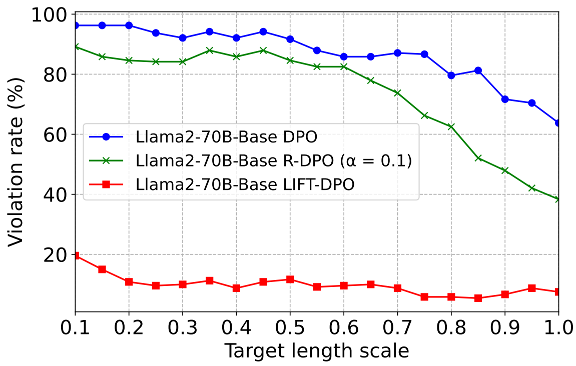

Figure 8: The violation rates of DPO, R-DPO, LIFT-DPO trained models based on Llama2-70B models on MT-Bench-LI as the target length shortens (via target length scale).

I Additional Length Instruction Following Results

In our MT-Bench-LI evaluations, we progressively reduced the length instructions by applying scaling factors to the existing values, ranging from 0.9 down to 0.1. We assessed the performance of various models based on the Llama-2-70B-Base, including standard DPO, R-DPO, and LIFT-DPO, and plotted their violation rates as shown in Figure 8). The results indicate that our LIFT-DPO trained model significantly outperforms both DPO and R-DPO in adhering to length constraints. Specifically, the LIFT-DPO model maintains a violation rate below 20% across all scaling factors, whereas both DPO and R-DPO models exhibit violation rates exceeding 80% when the scaling factor is reduced to less than 0.6. Additionally, we analyzed the performance of models based on Llama-3-8B-Instruct on AlpacaEval-LI under gradually reduced length limits. The observed trend is similar to that of MT-Bench-LI, as depicted in Figure 9.

J AlpacaEval Results & MT-Bench Results

The results of the LIFT-DPO models on standard AlpacaEval and MT-Bench are detailed in Table 6 and Table 8, respectively. Our analysis reveals

	AlpacaEval-LI			MT-Bench-LI		
	Vlt(%)	Win(%)	Words	Vlt(%)	Win(%)	Words
Llama2-70B-Base + DPO	65.8	4.6	216	60.8	5.0	199
Llama2-70B-Base + R-DPO (Park et al., 2024) ($\alpha = 0.01$)	63.8	5.2	217	57.9	2.1	194
Llama2-70B-Base + R-DPO (Park et al., 2024) ($\alpha = 0.1$)	45.0	7.7	178	39.4	8.5	161
Llama2-70B-Base + LIFT-DPO	**7.1**	**13.6**	151	**10.0**	**11.0**	146
Llama2-70B-Chat	28.2	11.3	162	38.3	11.9	168
Llama2-70B-Chat + DPO	15.1	10.4	135	24.2	10.8	147
Llama2-70B-Chat + LIFT-DPO	**2.7**	**14.2**	140	**6.7**	**12.5**	135

Table 5: **Llama 2 Length Instruction-Following results on the AlpacaEval-LI + MT-Bench-LI benchmarks.** LIFT-DPO yields improved winrates (Win(%)) and lower length instruction following violation rates (Vlt(%)).

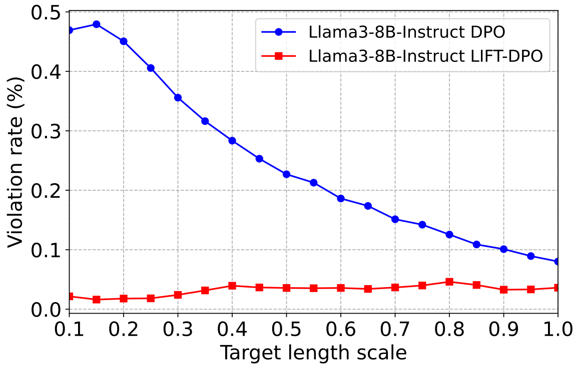

Figure 9: The violation rates of DPO, LIFT-DPO trained models based on Llama3-8B-Instruct on AlpacaEval-LI as the target length shortens (via target length scale).

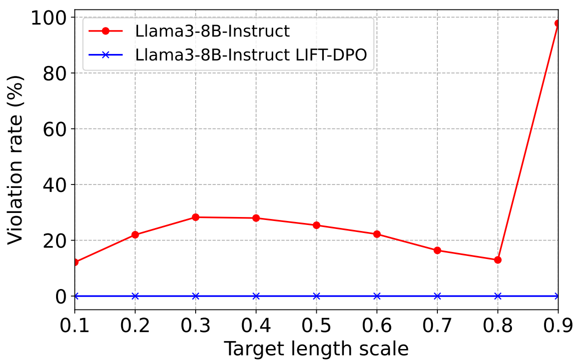

Figure 10: The violation rates of LIFT-DPO trained models based on Llama3-8B-Instruct on GSM8k test set as the target length shortens (via target length scale).

that the LIFT-DPO models exhibit no performance degradation when compared to the standard DPO models on these benchmarks.

K IFEval Results

While existing benchmarks such as IFEval (Zhou et al., 2023) also include instructions with verifiable constraints such as “write in less than 400 words”. We observe that the IFEval benchmark primarily focuses on the accuracy of adhering to verifiable constraints, placing more emphasis on this aspect than on the overall quality of the response. Instead, our Length-Instructed Benchmarks assess models on both the length instruction-following capabilities as well as qualities of generations. IFEval consists of 541 "verifiable instructions", of which 44 contain length constraints with upper bound limits on generation lengths. Among these length constraints, 22 are upper bound limits on word counts and 22 are upper bound limits on sentence length. We tested models’ performance in following these upper bound limit constraints, and the results are shown in Table 9. We found that LIFT-DPO achieved significantly lower violation rates on max word constraints and max sentence constraints compared to normal DPO.

Notably, our LIFT-DPO models, trained on datasets with fixed length instruction templates and word count constraints, not only improve models’ ability to follow natural length prompt templates that differ from what was used during training but also generalize to following max sentence constraints.

L Lower Bound Results

We also evaluate LIFT-DPO method on length constraints with lower bound word counts in this format “Answer the following instruction using at least <MAX_LEN> words. ”

Results of violation rates on lower bound limits and standard AlpacaEval are detailed in Table 10. Our analysis reveals that the LIFT-DPO models significantly lower the violation rates on instructions with lower bound limits and exhibit no performance degradation when compared to the standard DPO models on the AlpacaEval benchmarks. We also tested our model on IFEval where in all 541 instructions, 36 of them have lower bound limits on

Standard models	Vlt(%)	LC-Win(%)	Win(%)	Words
GPT4 Turbo(1106-preview)	91.1	50	50	324
GPT4 Turbo(0409-preview)	77.1	55.0	46.1	277
GPT4 Omni	77.8	57.5	51.3	282
Claude 3 Opus (02/29	57.8	40.5	29.1	219
Mistral Large (24/02)	49.7	32.7	21.4	223
Llama2-70B Chat	84.8	13.9	14.7	296
Llama3-70B Instruct	84.2	34.4	33.2	302
Llama3-8B Instruct	88.6	22.9	22.6	303
Llama2-70B Models				
26Llama2-70B + DPO	60.7	13.1	8.6	211
Llama2-70B + LIFT-DPO	65.7	15.4	9.9	220
Llama2-70B + R-DPO ($\alpha = 0.01$)	57.9	11.3	7.5	204
Llama2-70B + R-DPO ($\alpha = 0.1$)	48.6	13.6	8.0	187
Llama2-70B-Chat + DPO	66.8	23.3	12.6	218
Llama2-70B-Chat + LIFT-DPO	75.9	20.5	12.9	242
Llama3-8B Models				
Llama3-8B + DPO	45.1	13.9	7.8	188
Llama3-8B + LIFT-DPO	33.9	15.7	7.2	158
Llama3-8B-Instruct + DPO	86.5	26.3	25.8	308
Llama3-8B-Instruct + LIFT-DPO	85.1	26.5	22.7	285

Table 6: **Results on the AlpacaEval benchmark**. LIFT-DPO still maintains good performance in the standard (non-length) instruction-following setup.

Model	AlpacaEval-LI Vlt(%)
Llama2-70B + LIFT-SFT	46.7
Llama2-70B + LIFT-DPO	7.1
Llama2-70B-Chat + LIFT-SFT	50.5
Llama2-70B-Chat + LIFT-DPO	2.7

Table 7: **Results of LIFT-SFT on AlpacaEval-LI**. LIFT-SFT along is not enough for teaching models to follow length instructions.

generation word length. Results demonstrate that our LIFT-DPO models also achieve much lower violation rate compared to standard DPO models.

M SFT Results

In our experiments, models initially undergo SFT on preferred responses from our dataset which contain length instructions before the DPO phase. We have evaluated the effectiveness of this phase as well (which we denote as LIFT-SFT); however, our findings indicate that models fine-tuned with SFT alone do not adhere to length instructions as effectively as those further trained with DPO (see Table 7). For instance, the llama2-70B-chat model post-SFT exhibited a violation rate of 50.5% on AlpacaEval-LI, which significantly improved to 2.7% after DPO training (see the Table below for more details). This comparison highlights DPO's crucial role in enhancing the model's ability to precisely follow length constraints.

N LIFT-DPO Models on GSM8k

To see whether our trained models could generalize to out of domain tasks, we have also tested LIFT-DPO models on the GSM8k test set using the exact same setup as in Figure 2. Specifically, we used the gold solution length provided in the GSM8k test set as the target length, and we varied the scale (from 0.1 to 0.9) multiplied to the target length to see how the violation rate goes. Figure 10 are the results comparing the untrained Llama3-8B-Instruct and LIFT-DPO models. It's clear that the LIFT-DPO model maintains 0 violation rate across different scales while the Llama3-8B-Instruct model fails to follow the length constraints sometimes. The results demonstrate that even though our models are trained on the general instruction following domain, the underline length following mechanism learned by the model can generalize to other domains like math as well.

Llama2 Models	Overall Score	Math, Coding & Reasoning	Humanities, Extraction & STEM, Roleplay, Writing	Words
Llama2-70B + DPO	7.45	5.30	8.74	189
Llama2-70B + LIFT-DPO	7.54	4.77	9.21	275
Llama2-70B + R-DPO ($\alpha = 0.01$)	6.65	4.17	8.14	181
Llama2-70B + R-DPO ($\alpha = 0.1$)	6.53	3.53	8.33	163
Llama2-70B-Chat + DPO	7.58	5.03	9.10	218
Llama2-70B-Chat + LIFT-DPO	7.45	4.70	9.10	213
Llama3-8B Models				
Llama3-8B + DPO	7.11	8.44	4.90	158
Llama3-8B + LIFT-DPO	6.99	8.54	4.40	138
Llama3-8B-Instruct + DPO	8.38	6.30	9.62	263
Llama3-8B-Instruct + LIFT-DPO	8.32	6.27	9.55	237

Table 8: **Results on the MT-Bench benchmark**. LIFT-DPO still maintains good performance in the standard (non-length) instruction-following setup.

	Max Sentence	Max Word
Llama3-8B-Base (zero shot)	54.5	72.7
Llama3-8B-Base + DPO	48.5	63.6
Llama3-8B-Base + LIFT-DPO	**22.7**	**16.7**
Llama3-8B-Instruct	13.6	45.5
Llama3-8B-Instruct + DPO	13.6	31.8
Llama3-8B-Instruct + LIFT-DPO	**9.1**	**25.8**

Table 9: **Results of LIFT-DPO on IFEval.** LIFT-DPO yields much lower length instruction following violation rates on max word constraints and max sentence constraints compared to normal DPO.

	AlpacaEval	IFEval	AlpacaEval (Standard)	
	Lower Bound Vlt(%)	Lower Bound Vlt(%)	LC-Win(%)	Words
Llama3-8B-Instruct	18.0	36.1	22.9	304
Llama3-8B-Instruct + DPO	19.8	30.6	25.8	308
Llama3-8B-Instruct + LIFT-DPO	**3.2**	**19.4**	25.8	289

Table 10: **Results of LIFT-DPO on Instructions with Lower Bound Constraints on Generation Lengths:** LIFT-DPO yields much lower violation rates (Vlt(%)) on min word constraints on both AlpacaEval lower bound LI and IFEval lower bound LI, while maintaining similar LC-winrate performance compared to DPO.