A Grounded Preference Model for LLM Alignment

Tahira Naseem[*,†] **Guangxuan Xu**[†]
Sarathkrishna Swaminathan **Asaf Yehudai** **Subhajit Chaudhury**
Radu Florian **Ramón Fernandez Astudillo** **Asim Munawar**
[*]tnaseem@us.ibm.com
IBM Research

Abstract

Despite LLMs' recent advancements, they still suffer from factual inconsistency and hallucination. An often-opted remedy is retrieval-augmented generation – however, there is no guarantee that the model will strictly adhere to retrieved grounding. Fundamentally, LLMs need to be aligned to be more faithful to grounding, which will require high-quality preference annotations. This paper investigates whether we can create high-quality grounded preference data for model alignment without using annotations from humans or large proprietary models. We experimented with existing entailment data and proposed approaches to generate synthetic grounded preference data, with which we train a Grounded Preference Model(GPM). We demonstrate through Proximal Policy Optimization(PPO) training of Mistral-7B-Instruct that our GPM model can successfully align powerful LLMs to generate much better grounded responses as judged by GPT4. Moreover, we show that our GPM is also a great faithfulness classifier, achieving SoTA in dialogue sub-tasks of the TRUE faithfulness Benchmark. We release GPM under the Apache 2.0 license [1].

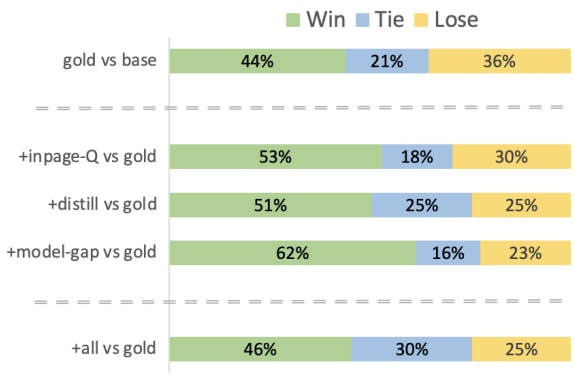

Figure 1: GPM ablations results comparing Mistral and its aligned versions with GPT-4 as a judge. The first bar plot compares gold-GPM(trained only on entailment data) aligned Mistral with the original Mistral. Subsequent plots use GPM-gold as a baseline; we find GPM trained with model-gap synthetics plus entailment gold outperform gold-GPM alignment by large margin.

1 Introduction

Large Language Models (LLMs) have seen rapid advancements, yet they continue to suffer hallucinations in both open-domain and grounded generations (Goodrich et al., 2019; Kryscinski et al., 2019). This undermines the usability of LLMs for high-stake applications. To address these challenges, we explore model alignment with Reinforcement Learning (RL) to emulate human preferences in model outputs (Ouyang et al., 2022b; Bai et al., 2022; Touvron et al., 2023). More specifically, we propose Grounded Preference Model (GPM) – a model trained to assess the overall quality of *grounded* responses, and leverage it to align LLMs toward more faithful generation.

Training a preference model traditionally demands substantial human annotation, which is expensive and labor-intensive. Moreover, there are very few publicly available preference datasets for content-grounded dialogues. Our proposed method simplifies this process by utilizing existing entailment datasets combined with synthetically generated preference data for model alignment. Specifically, we explore ways to reliably generate/curate preference pairs for queries from existing content-grounded datasets.

We demonstrate the effectiveness of GPM for model alignment on a leading LLM, Mistral-7B-Instruct-v0.1[2] (Jiang et al., 2023). The correctness and helpfulness of Mistral responses, as judged by GPT4, improves after PPO training for all variants of GPM. Moreover, we evaluate GPM as a faithfulness metric on TRUE benchmark. Our model

[1] https://huggingface.co/ibm/grounded-preference-model
† equal contribution

[2] all mentions of Mistral refer to Mistral-7B-Instruct-v0.1

Findings of the Association for Computational Linguistics ACL 2024, pages 151–162
August 11-16, 2024

performs comparably to the SoTA model on average, with clear gains in the dialogue sub-task – establishing a new SoTA for the task.

2 Grounded Preference Model

2.1 Preference Data Creation

Grounded Preference Model(GPM) is trained to prefer a faithful and high quality response over a hallucinated and incoherent response. We can formalize the preference dataset as pairs (1 winning, 1 losing) of triplets, each comprising of three elements: Document, Conversation, and Response represented by D, Q, and R respectively. The Document serves as the grounding knowledge. The Conversation can be a single-turn or multi-turn dialogue pertaining to the document and ending in a user query. The Response is the generated output. Each preference example is a triplet pair, $e_{win} = (D, Q, R), e_{lose} = (D, Q, R)$.

In the following, we describe various ways to create grounded preference data:

Gold Entailment Data (Gold) Several human-created entailment datasets have examples comprising a premise and a hypothesis along with an entailment label. We can re-purpose entailment data to be grounded preference data by the following process: for each pair of entailment instance that shares either the premise or the hypothesis, but have different entailment labels, we combine them to create one grounded preference instance; the example with a positive entailment label is preferred over the other.

We apply this process to six entailment datasets: FEVER (Thorne et al., 2018), HoVer (Jiang et al., 2020), MNLI (Bowman et al., 2015), SNLI (Williams et al., 2018), SciTail (Khot et al., 2018), and VitaminC (Schuster et al., 2021). The statistics of each dataset is given in Table 5 in appendix A.

Inpage Query Swap (inpage-Q) This method works on datasets where multiple query and response pairs correspond to the same document. Examples of such datasets include Multi-Doc2Dial and Wish-QA (Feng et al., 2021; Yehudai et al., 2024). Given a document and multiple gold conversations/questions, $\{Q_w^i\}_{i=1}^n$ along with their responses, $\{R_w^i\}_{i=1}^n$, we create negatives by swapping Q_w^j with Q_w^k for $k \in [n]; k \neq j$. This will result in a triplets $e_l = (D^i, Q^k, R^i)$ where both response and query are related to the document but the response does not address the question. We refer to this type of synthetic preference data as "inpage-Q" dataset.

Let the Best LLM Win (model-gap) In this method, we hypothesized that the ranking of various LLMs should translate into a ranking over their generated output. Given a pair of LLMs where one is superior (i.e. Falcon-180B and Falcon-40B), we generate responses via each of them for the same D_w and Q_w. The response from the higher-ranked model is treated as a positive response while the other one, as negative. This approach is similar to Kim et al. (2023), however, we explore its efficacy in content grounded setting. We apply this method using the following LLMs listed in the order of their ranking: Falson-180b, Falcon-40b, flan-t5-xxl, flan-t5-xl and flan-t5-large (Penedo et al., 2023; Wei et al., 2022). The source datasets for this method come from SQuAD-v2, CoQA, Multi-Doc2Dial, QUAC and FloDial (Raghu et al., 2021). We refer to this type of synthetic preference data as "model-gap" dataset.

Faith Score Distillation (distill) In this method, for a gold faithful triplet $e_w = (D_w, Q_w, R_w)$, we generate multiple responses for query Q_w and document D_w at high sampling temperature (T=1.2), encouraging hallucinative responses. To ensure these generated responses can be treated as negatives, we evaluate their faithfulness to the document using an ensemble of faithfulness metrics. Responses that score below a threshold are used as negatives. Since this method distills knowledge from faithfulness metrics to create synthetic data, we refer to it as "distill" dataset. Flan-t5-xxl and Flan-t5-xl are used to generate responses, while faithfulness metrics ANLI, FactCC, and SummaC are used for filtering responses. The source datasets are SQuAD-v2, CoQA, Multi-Doc2Dial, QUAC and FloDial (Raghu et al., 2021).

2.2 Preference Model Objective

The preference modeling objective is defined via the Bradley-Terry (Bradley and Terry, 1952; Rafailov et al., 2023) model of pairwise comparisons,

$$p(x \succ y) = \frac{\exp(r_x)}{\exp(r_x) + \exp(r_y)}$$

where $\succ$ indicates preference relation and r_i is the score (or reward) for i. When used in the context of LLMs, the elements of the comparison are

model-generated responses, and the scores are assigned with respect to a context (typically an instruction or a question). Following the Bradley-Terry model, the objective would be,

$$\mathcal{L}(r_\theta, \mathcal{D}) = -\mathbb{E}_{(e_w,e_l)\sim\mathcal{D}}[\log \sigma(r_\theta(e_w) - r_\theta(e_l)]$$

We implement this objective using an encoder-only transformer model for r_θ. In particular, we use the DeBERTa large model[3] and employ token-type embeddings to distinguish D, Q from R. A reward modeling head is added on top of the [CLS] token's output embedding in the form of a $d \times 1$ linear layer, where d is the dimension of the final hidden layer.

2.3 Preference Model Training

We train the GPM on 1.8 million gold and 0.7 million synthetically generated samples. For each synthetic data type, the ratio between gold and synthetic during training is 10:1 respectively. We train for 100k steps with a batch size of 20 and a learning rate of 1e-5[4]. We run one experiment for each setting and use the last checkpoint.

3 GPM for LLM Alignment

We use the standard RLHF procedure (Ouyang et al., 2022a) for model alignment that optimizes:

$$\mathbb{E}_{(x,y)\sim D_{\pi_\phi^{RL}}}\left[r_\theta(x,y) - \beta \log \frac{\pi_\phi^{RL}(y|x)}{\pi^{BASE}(y|x)}\right]$$

where r_θ denotes the reward score, π_ϕ^{RL} represents the RL policy and π^{BASE} is the initial (instruct) model, serving as a baseline policy. β moderates the Kullback-Leibler divergence to prevent excessive deviation of π_ϕ^{RL} from π^{BASE}. We optimize the above objective using Proximal Policy Optimization(PPO) (Schulman et al., 2017).

3.1 Experimental Setup

We use TRLX Library (Castricato et al., 2023) for PPO training – modified to perform parameter efficient Quantized LoRA(QLoRA) (Dettmers et al., 2023) fine-tuning. This allows us to fit the entire PPO training pipeline on a single 80GB GPU.

[3] microsoft/deberta-v3-large

[4] we did not perform hyperparameter search, and used the biggest batch that could fit in memory.

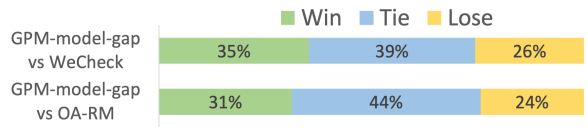

Figure 2: GPM vs. OA-RM and WeCheck for PPO (3.1)

Training Setup We curate 100k data as the distribution $D_{\pi_\phi^{RL}}$ to sample prompts for PPO training. Train data are from the following sources MultiDoc2Dial (Feng et al., 2021), QuAC (Choi et al., 2018), SQuAD_v2 (Rajpurkar et al., 2016), CoQA (Reddy et al., 2018), ASQA (Stelmakh et al., 2022), ELI5 (Fan et al., 2019), DoQA (Campos et al., 2020), FloDial (Raghu et al., 2021) (see Appendix 7 for statistics). We chose Mistral-7B-Instruct-v0.1 to be the policy model, and write tailored system prompts and instructions to allow better learning and exploration (see Appendix 8 for details on the policy model and prompts).

Baselines We choose 2 external models as baselines for alignment reward: 1) WeCheck[5] (Wu et al., 2022) – the highest performing model for faithfulness on TRUE benchmark (Honovich et al., 2022) 2) OA-RM is an OpenAssiatant reward model trained on publicly available helpfulness preference datasets[6]. Similar to GPM both these models depart from the deberta-v3-large. We also compare GPM aligned models against the base LLM.

GPM-variants Five variants of GPM are depicted in Figure 1. *gold* is the GPM trained only on the gold entailment data. *+in-page-Q*, *+distill* and *+model-gap* are GPMs trained on combination of gold entailment data plus the respective synthetic data types. *all* is a GPM trained on all of the synthetic preference data plus the gold entailment data.

GPT4 as a Judge We use GPT4 to evaluate the correctness and helpfulness of response in a grounded setting. The evaluation prompt is adapted from (Zheng et al., 2023) and released in Appendix 9. For each comparison round, we randomly sampled 50 instances from each of the 6 evaluation domains (300 in total). Half of the evaluation data is in-domain for PPO training, while the other half is out of domain with details in Appendix E. The ordering of the outputs given to GPT4 is shuffled at instance-level to prevent ordering bias.

[5] nightdessert/WeCheck

[6] OpenAssistant/reward-model-deberta-v3-large-v2

3.2 Ablation Results for Alignment

The first bar chart of Figure 1 shows that Grounded Preference Model trained with only gold entailment data already produces an better aligned Mistral for grounded generation. Then, we set gold entailment model as the baseline, and compare it with GPMs trained with additional synthetic preference data, to see if synthetic data adds value. It is clear from the middle bar charts that entailment + synthetic preference data makes better preference model for alignment. In particular, GPM with added model-gap data stood out to beat gold entailment model 62% to 23%. However, it's worth noting that all-vs-gold shows more modest improvements over gold, displaying a diminishing returns effect rather than a synergic effect when we combined all sources of synthetic data.

3.3 Comparison with External Models

While our GPMs consistently outperform gold-entailment model, we also compare best version of GPM with the SoTA faithfulness model (WeCheck) and a strong human-preference reward model OA-RM. Despite the strong results by both OA-RM and WeCheck, GPM still wins by a 7-9% margin. Knowing that GPM-model-gap is not trained on any human preference data, this alignment performance is quite astonishing.

3.4 Qualitative Analysis

The two baseline reward models – OA-RM and WeCheck – are trained with different goals and training objectives. OA-RM is trained to judge the helpfulness of non-grounded responses, while WeCheck measures the faithfulness of a text to another text. GPM aims to address both, the faithfulness of a response to the grounding document as well as its adequacy to the user question. GPT4 judgments often found the responses of the LLM aligned using GPM to be more directly answering the questions while being consistent with the document. On the other hand, OA-RM would sometimes allow the LLM to hallucinate (see Table 1), trading faithfulness for helpfulness – and WeCheck preferred responses that had a high overlap with the grounding context even when they did not necessarily answer user query (see Table 2).

limitations of GPM — for instance, currently, GPM emphasizes greatly on faithfulness to the document/passage, and as a result, sometimes does not pay due attention to the adequacy of the response

[Document/Passage] The river as far as Stamford was used by the Romans for navigation, as it formed part of a system including the Car Dyke, which ran along the western edge of the Fens and crossed the river near the modern Folly River. Navigation to Stamford was improved by the canal. Boats used on the canal were small lighters, around wide, capable of carrying from seven to fourteen tons, and normally worked in trains of four vessels. With the arrival of the railways, river trade declined.
[User Conversation] User: where does the river welland start and finish Agent: The river rises in the Hothorpe Hills, at Sibbertoft in Northamptonshire, then flows generally northeast to Market Harborough, Stamford and Spalding, to reach The Wash near Fosdyke User: with reference to the mouth of the river which you mentioned earlier, what is its water temperature?
[Response after PPO with GPM] The water temperature of the river Welland at its mouth near Fosdyke is not provided in the given document.
[Response after PPO with OA-RM] 15 degrees Celsius

Table 1: Hallucinated response after PPO with OA-RM

to the question. However, it is better in this aspect than the purely faithfulness-based classifier such as WeCheck.

4 GPM as Faithfulness Metric

While GPM combines preference and groundedness, it should preserve its ability to discern unfaithful responses and remain a reliable faithfulness metric. We evaluate GPM 's performance on the TRUE benchmark (Honovich et al., 2022) – designed to comprehensively evaluate faithfulness across 11 datasets from 4 tasks. We report performance on dialogue sub-task as well as overall scores. We use ROC-AUC as our evaluation metric Honovich et al. (2022). We compare variants of GPM against two top-performing models: T5-11B fine-tuned on ANLI (Honovich et al., 2022; Nie et al., 2020), and WeCheck (Wu et al., 2023).

4.1 Results on TRUE Benchmark

Table 3 shows that GPM delivers strong results on TRUE benchmark, achieving new SoTA in the Dialogue sub-tasks on which our GPM specializes. We observe the synergic effect(absent from LLM alignment experiment) that GPM + all gives

[Document/Passage] (removed for brevity)
[User Conversation] User: where does this whale species get its name from?
[Response after PPO with GPM] The name "sperm whale" comes from the fact that they are known for producing spermaceti, a waxy substance that was once used for a variety of purposes, including as a lubricant and in the production of soap and candles.
[Response after PPO with WeChek] According to the provided document, the sperm whale is the largest toothed whale, with adult males measuring up to long and weighing up to ... ... but mature males are typically 30% to 50% longer and three times as massive as females. Newborn sperm whales are usually between ...

Table 2: Unhelpful response after PPO with WeCheck

Model	Data	TRUE (ROC AUC) Dialogue	Overall
ANLI	ANLI	77.7	81.5
WeCheck		86.2	**84.8**
GPM	Gold	86.4	83.1
GPM	+ inpage-Q	86.4	83.3
GPM	+ model-gap	86.7	80.7
GPM	+ distill	86.2	83.6
GPM	+ all	**87.2**	84.3

Table 3: Variants of GPM on TRUE benchmark.

the highest TRUE Benchmark scores. We also observe that the GPM + model-gap which gives the best alignment performance do not display similar strength in the faithful benchmark. A plausible interpretation is that the model-gap data contribute more towards preference than faithfulness, thus resulting in greater improvement on preference alignment. Overall, we do believe that a solid faithful benchmark performance is the foundation for GPM 's success in Grounded Preference Alignment.

4.2 Effect of Model Architecture

We also explore transformer architectures other than the bidirectional encoders – in particular, we train auto-regressive and encoder-decoder transformer models with the same training setup as GPM. The underlying model for GPM is deberta-v3-large, which is a 435M parameters model. To keep the model size comparable, we experiment with gpt2-large (774M) (Radford et al., 2019) and t5-large (770M) (Raffel et al., 2020) models. The results in Table 4 show that the encoder only deberta model (GPM) outperforms these models on

Model	Params	TRUE (ROC AUC) Dialogue	Overall
gpt2-large	774M	72.4	68.6
t5-large	770M	81.3	77.4
deberta-v3-large	435M	**87.2**	**84.3**

Table 4: Different transformer architectures trained with the same Gold+all, bottom row correponds to the GPM.

TRUE by a large margin. We conjecture that the bi-directional attention makes this model more context-aware and hence better suited for scoring outputs. However, we note that the pre-training setups of these models are different, which can be a strong contributing factor in their final performance after preference training.

5 Related Work

Various approaches have been proposed to make LLMs more reliable. Prompting-based methods prompt with counterfactual demonstrations (Zhou et al., 2023) or employ chain-of-thought self-verification (Dhuliawala et al., 2023). Prefix tuning (Jones et al., 2023); tunes model's system message on a synthetic task where hallucinations are easy to identify and then transfer them to abstractive summarization tasks. On the other hand, there are RL-based approaches that use automatic metrics to reward faithful generation. Du and Ji (2023) employ SacreBLEU and BertScore as reward signals, while Roit et al. (2023) use log-probabilities from the ANLI-classifier (Nie et al., 2020). ; Unlike previous works that rely on the existing automatic metrics of faithfulnes, our major contribution is a preference model trained specifically for the grounded generation tasks. A significant body of work models faithfulness as a classification task: Nie et al. (2020), Wu et al. (2023) and Gekhman et al. (2023). Our work is set apart in its modeling approach as well as in its ability to leverage relative preferences, obviating the need for hard labels.

6 Conclusions

This paper investigates the recipe for grounded preference alignment. We find that entailment data can be repurposed to train good grounded preference models, which align base policy towards faithful generation. Moreover, by adding synthetic preference data, we are able to train GPM that not only achieves new SoTA in faithful, but also serves as a reward model for LLM alignment. Our approach is simple and has no reliance on proprietary AIs.

7 Limitations

We use GPT4 as a judge for evaluation, which correlates with human preferences, but it can not be seen as a perfect substitute. We test our approach on Mistral-7B-Instruct; testing on a few more models will strengthen the results and further establish the generality of the method.

References

Vaibhav Adlakha, Shehzaad Dhuliawala, Kaheer Suleman, Harm de Vries, and Siva Reddy. 2021. Topiocqa: Open-domain conversational question answering with topic switching. *Transactions of the Association for Computational Linguistics*, 10:468–483.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Ralph Allan Bradley and Milton E. Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3-4):324–345.

Jon Ander Campos, Arantxa Otegi, Aitor Soroa, Jan Deriu, Mark Cieliebak, and Eneko Agirre. 2020. DoQA - accessing domain-specific FAQs via conversational QA. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7302–7314, Online. Association for Computational Linguistics.

Louis Castricato, Alex Havrilla, Shahbuland Matiana, Duy V. Phung, Aman Tiwari, Jonathan Tow, and Maksym Zhuravinsky. 2023. trlX: A scalable framework for RLHF.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. Quac: Question answering in context. In *Conference on Empirical Methods in Natural Language Processing*.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *ArXiv*, abs/2305.14314.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2023. Chain-of-verification reduces hallucination in large language models.

Wanyu Du and Yangfeng Ji. 2023. Blending reward functions via few expert demonstrations for faithful and accurate knowledge-grounded dialogue generation.

Nouha Dziri, Ehsan Kamalloo, Sivan Milton, Osmar Zaiane, Mo Yu, Edoardo M Ponti, and Siva Reddy. 2022. FaithDial: A Faithful Benchmark for Information-Seeking Dialogue. *Transactions of the Association for Computational Linguistics*, 10:1473–1490.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3558–3567, Florence, Italy. Association for Computational Linguistics.

Song Feng, Siva Sankalp Patel, Hui Wan, and Sachindra Joshi. 2021. Multidoc2dial: Modeling dialogues grounded in multiple documents. *ArXiv*, abs/2109.12595.

Zorik Gekhman, Jonathan Herzig, Roee Aharoni, Chen Elkind, and Idan Szpektor. 2023. Trueteacher: Learning factual consistency evaluation with large language models. In *EMNLP*.

Ben Goodrich, Vinay Rao, Peter J. Liu, and Mohammad Saleh. 2019. Assessing the factual accuracy of generated text. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery amp; Data Mining*, KDD '19. ACM.

Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. 2022. TRUE: Re-evaluating factual consistency evaluation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3905–3920, Seattle, United States. Association for Computational Linguistics.

Albert Qiaochu Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, L'elio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *ArXiv*, abs/2310.06825.

Yichen Jiang, Shikha Bordia, Zheng Zhong, Charles Dognin, Maneesh Singh, and Mohit Bansal. 2020. HoVer: A dataset for many-hop fact extraction and

claim verification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3441–3460, Online. Association for Computational Linguistics.

Erik Jones, Hamid Palangi, Clarisse Simões, Varun Chandrasekaran, Subhabrata Mukherjee, Arindam Mitra, Ahmed Awadallah, and Ece Kamar. 2023. Teaching language models to hallucinate less with synthetic tasks.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2018. Scitail: A textual entailment dataset from science question answering. In *AAAI Conference on Artificial Intelligence*.

Sungdong Kim, Sanghwan Bae, Jamin Shin, Soyoung Kang, Donghyun Kwak, Kang Min Yoo, and Minjoon Seo. 2023. Aligning large language models through synthetic feedback. *arXiv preprint arXiv:2305.13735*.

Wojciech Kryscinski, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Neural text summarization: A critical evaluation. 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 540–551, Hong Kong, China. Association for Computational Linguistics.

Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Evaluating the factual consistency of abstractive text summarization. *arXiv preprint arXiv:1910.12840*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. Summac: Re-visiting nli-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. 2022a. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022b. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon llm: Outperforming curated corpora with web data, and web data only.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*.

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.

Dinesh Raghu, Shantanu Agarwal, Sachindra Joshi, and Mausam. 2021. End-to-end learning of flowchart grounded task-oriented dialogs. In *Conference on Empirical Methods in Natural Language Processing*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Conference on Empirical Methods in Natural Language Processing*.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2018. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Paul Roit, Johan Ferret, Lior Shani, Roee Aharoni, Geoffrey Cideron, Robert Dadashi, Matthieu Geist, Sertan Girgin, Leonard Hussenot, Orgad Keller, Nikola Momchev, Sabela Ramos Garea, Piotr Stanczyk, Nino Vieillard, Olivier Bachem, Gal Elidan, Avinatan Hassidim, Olivier Pietquin, and Idan Szpektor. 2023. Factually consistent summarization via reinforcement learning with textual entailment feedback. In

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 6252–6272, Toronto, Canada. Association for Computational Linguistics.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *ArXiv*, abs/1707.06347.

Tal Schuster, Adam Fisch, and Regina Barzilay. 2021. Get your vitamin C! robust fact verification with contrastive evidence. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 624–643, Online. Association for Computational Linguistics.

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. 2022. ASQA: Factoid questions meet long-form answers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8273–8288, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Mirac Suzgun, Nathan Scales, Nathanael Scharli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed Huai hsin Chi, Denny Zhou, and Jason Wei. 2022. Challenging big-bench tasks and whether chain-of-thought can solve them. In *Annual Meeting of the Association for Computational Linguistics*.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. 2018. The fact extraction and VERification (FEVER) shared task. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 1–9, Brussels, Belgium. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin R. Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Daniel M. Bikel, Lukas Blecher, Cristian Cantón Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony S. Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel M. Kloumann, A. V. Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, R. Subramanian, Xia Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zhengxu Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *ArXiv*, abs/2307.09288.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Wenhao Wu, Wei Li, Xinyan Xiao, Jiachen Liu, Sujian Li, and Yajuan Lv. 2022. Wecheck: Strong factual consistency checker via weakly supervised learning. *arXiv preprint arXiv:2212.10057*.

Wenhao Wu, Wei Li, Xinyan Xiao, Jiachen Liu, Sujian Li, and Yajuan Lyu. 2023. WeCheck: Strong factual consistency checker via weakly supervised learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 307–321, Toronto, Canada. Association for Computational Linguistics.

Asaf Yehudai, Boaz Carmeli, Yosi Mass, Ofir Arviv, Nathaniel Mills, Assaf Toledo, Eyal Shnarch, and Leshem Choshen. 2024. Genie: Achieving human parity in content-grounded datasets generation.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*.

Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied Sanosi Saied, Weizhu Chen, and Nan Duan. 2023. Agieval: A human-centric benchmark for evaluating foundation models. *ArXiv*, abs/2304.06364.

Wenxuan Zhou, Sheng Zhang, Hoifung Poon, and Muhao Chen. 2023. Context-faithful prompting for large language models. In *Findings of EMNLP*.

A Statistic of Preferences from Entailment Data

Dataset	Number of Examples
FEVER	522,181
HoVER	9,072
MNLI	274,720
SNLI	51,485
SciTail	708,581
VitaminC	318,302

Table 5: Preferences from Entailment data.

B QLoRA Tuning in Llama2-Based Models

In tuning the LoRA parameters for WizardLM, we chose a subset of modules for the `lora_target_module` parameter. This subset includes:

- `"up_proj"`
- `"q_proj"`
- `"down_proj"`
- `"o_proj"`
- `"v_proj"`
- `"k_proj"`

This selection contrasts with the default set in QLoRA, which typically encompasses all linear layers in the model: `['up_proj', 'q_proj', 'gate_proj', 'down_proj', 'o_proj', 'v_proj', 'k_proj']`. Our tailored approach reduces CUDA memory requirements, and allows longer context lengths during training.

For the Mistral-7B model, we adhered to the standard QLoRA configuration, incorporating all linear layers as the `lora_target_module`.

C Hyper-Parameters for Model Training

To facilitate the replication of our results in Proximal Policy Optimization (PPO) experiments using the TRLX (Castricato et al., 2023) library, we enumerate the key hyperparameters used. Our training utilized the DeepSpeed engine, specifically leveraging its stage-2 configuration.

Training Hyper-Parameters The following table 6 outlines the crucial hyperparameters employed during the training process, including the quantization parameters:

Compute Each training run is performed on single NVIDIA A100 GPU with 80GB memory. It takes 38 hours to complete training for Mistral-7B model for 1 epoch with 100k steps.

D Statistics of Training Dataset

Table 7 shows the PPO training set statistics.

Parameter	Value
gradient_accumulation_steps	1
batch_size	1
max_input_length	900
max_output_length	150
optimizer	adamw
num_rollouts	256
num_layers_unfrozen	2
init_kl_coef	0.05
num_training_steps	100000
chunk_size	8
gradient_checkpointing	true
double_quant	true
quant_type	'nf4'
load_in_4_bits	true
torch_dtype	torch.float32

Table 6: Training Hyper-Parameters and Model Quantization Parameters

Dataset-name	NO. of Samples
MultiDoc2Dial (Feng et al., 2021)	16,723
QuAC (Choi et al., 2018)	11,009
SQuAD_v2 (Rajpurkar et al., 2016)	11,133
CoQA (Reddy et al., 2018)	11,102
ASQA (Stelmakh et al., 2022)	5,568
ELI5 (Fan et al., 2019)	22,216
DoQA (Campos et al., 2020)	5,481
FloDial (Raghu et al., 2021)	16,669
	Sum: 99,901

Table 7: The sampling proportions for PPO training dataset; up-sampling is applied to get desired proportion.

E Details of Evaluation Dataset used in GPT4 as a Judge

Evaluation Datasets We include both in-domain and out-of-domain datasets for evaluation. In-domain datasets include Multi-Doc2Dial, QuAC and SQuAD-v2, which the model has seen during PPO training[7]. Out-of-domain evaluation set includes TopiOCQA (Adlakha et al., 2021), FaithDial (Dziri et al., 2022), and Natural Questions (Kwiatkowski et al., 2019), which were absent from PPO training.

F System Prompts for Different Models

Mistral-7B-Instruct-v0.1: Mistral 7B (Jiang et al., 2023) is an open-source model that outperforms Llama2 13B on LLM benchmarks, including AGI Eval (Zhong et al., 2023) and BBH (Suz-

[7]Note, that in PPO-based RL training, the model never uses gold responses from these datasets.

gun et al., 2022). **Mistral-7B-Instruct-v0.1** is the aligned version of Mistral-7B. Training details are not disclosed for the model.

Note, we use the same system prompt during PPO training and model evaluation. The following table 8 contains the prompts.

G Faithfulness Metrics Results

We use three faithfulness metrics: ANLI (Nie et al., 2020), FactCC (Kryściński et al., 2019) and SummaC (Laban et al., 2022). We also report standard generation evaluation metrics: RougeL, Bert-Recall and Bert-KPrecision.

The evaluation is conducted on the development set of 6 datasets: In-domain datasets include Multi-Doc2Dial, QuAC and SQuAD-v2, which the model has seen during PPO training[8]. Out-of-domain evaluation set includes TopiOCQA (Adlakha et al., 2021), FaithDial (Dziri et al., 2022), and Natural Questions (Kwiatkowski et al., 2019), which were absent from PPO training.

Results see Table 10

[8]Note, that in PPO-based RL training, the model never uses gold responses from these datasets.

Model Name	Prompt Template
Mistral-7B-Instruct	<s>[INST] <<SYS>> You are a helpful, respectful and honest assistant. Always answer as helpfully as possible, while being safe. Your answers should not include any harmful, unethical, racist, sexist, toxic, dangerous, or illegal content. Please ensure that your responses are socially unbiased and positive in nature. If a question does not make any sense, or is not factually coherent, explain why instead of answering something not correct. If you don't know the answer to a question, please don't share false information. <</SYS>> [document]: <DOCUMENT-TEXT> Answer the following questions based on the above document. [/INST] [conversation]: </s>[INST] <USER-QUERY-TEXT>[/INST] <AGENT-RESPONSE-TEXT></s>[INST]<USER-QUERY-TEXT> [/INST]

Table 8: Prompt for both RL-Alignment and Evaluation

Model Name	Prompt Template
GPT4-judge	Please act as an impartial judge and evaluate the quality of the responses provided by the two AI assistants to the user question displayed below. Your evaluation should consider correctness and helpfulness. You will be given a reference document, a user conversation, assistant A's answer, and assistant B's answer. Your job is to evaluate which assistant's answer is better based on the information in the reference document and the user conversation so far. Begin your evaluation by comparing both assistants' answers with the document and the user conversation so far. Identify and correct any mistakes. Avoid any position biases and ensure that the order in which the responses were presented does not influence your decision. Do not allow the length of the responses to influence your evaluation. Do not favor certain names of the assistants. Be as objective as possible. After providing your explanation, output your final verdict by strictly following this format: "[[A]]" if assistant A is better, "[[B]]" if assistant B is better, and "[[C]]" for a tie. [User Document] ... [User Conversation] ... [The Start of Assistant A's Answer] ... [The End of Assistant A's Answer] [The Start of Assistant B's Answer] ... [The End of Assistant B's Answer]

Table 9: Prompt for GPT4 as a judge evaluation.

RM	ANLI	Factcc	Summac	RougeL	B-Rec.	B-KPrec.
base model	0.5	0.22	0.40	0.14	0.10	0.10
Gold	0.57	**0.64**	**0.81**	0.16	**0.27**	**0.75**
+inpage-Q	0.51	0.49	0.37	0.19	0.16	0.18
+distill	0.52	0.26	0.48	0.11	0.19	0.14
+model_gap	0.46	0.51	0.45	0.17	0.21	0.27

Table 10: PPO training of Mistral on variants of GPM with different synthetic data types.