Week 6: InstructGPT Report

BEH Chuen Yang

Abstract

This report takes a closer look at the InstructGPT paper by Ouyang et al. (2022), one of the first papers to describe training language models which are attuned to human preferences, in terms of safety, truthfulness, and helpfulness. We detail insights from the paper, and follow up with a few potential research directions inspired by Ouyang et al. (2022)'s findings.

1 Paper Summary

Ouyang et al. (2022) introduce InstructGPT, a language model (LM) trained not just by using a large corpus of text (in pre-training, as well as Supervised Fine-Tuning (SFT)), but also with high-quality human feedback via Reinforcement Learning from Human Feedback (RLHF) (Christiano et al. (2017); Stiennon et al. (2020)).

The authors find InstructGPT to be more truthful and less toxic based on dataset as well as human evaluations (e.g. using Gehman et al. (2020)), at minimal cost to its performance on public NLP benchmarks. Despite it being <1% of GPT-3-175B's size, InstructGPT was vastly preferred by human evaluators over GPT-3-175B when asked to select between the two models' outputs on a variety of tasks.

1.1 Why Human Feedback?

While most LMs have managed to predict the next token in natural language sequences with much success at this point (Radford et al. (2019); Brown et al. (2020)), the authors of InstructGPT are interested in taking LMs a step further: they want LMs to be better understand users' intent, and act in a way which is most helpful to them (this is termed *alignment*).

With a pure ”predict-next-token” objective, LMs are prone to generating objectionable outputs, described in the paper as ”toxic, biased, or otherwise harmful” (Ouyang et al. (2022)).

While definitive causes have not been established, it is likely that the LMs have simply encountered such content in the training data, and are producing it as they were trained to do. While it may seem irresponsible to include such content in the training data, we note it may be impossible to completely filter out such content via manual methods, given the sheer size of the training data (Karpathy (2025)).

Instead, model alignment emerges as a more practical and scalable alternative. In this regime, an *aligned* LM is still *capable* of spewing out objectionable content, but is *trained to avoid doing so*. Qualitatively speaking, this would mean the LM is better at answering users' queries and/or commands in a way that is helpful, truthful, and safe.

2 The RLHF Pipeline

The original RLHF pipeline used in InstructGPT comes after *Pre-Training and Supervised Fine-Tuning (SFT)* of the LM, and consists of three sub-steps:

2.1 Human-Labelled Data Collection

Aligning LMs doubtlessly requires gold-standard, human feedback on possible responses by the LM to user queries, which can still be quite taxing, even if not prohibitive, to collect.

Indeed, Ouyang et al. (2022) had to engage a team of 40 contractors selected for their ability to detect and respond to sensitive speech (i.e. anything that could elicit strong, negative emotions).

The contractors were given various tasks such as

- Creating instruction-like prompts for the LM to respond to.
- Writing responses to user- and contractor-generated prompts.
- Ranking the LM's responses to user- and contractor-generated prompts, based on their helpfulness, truthfulness, and safety.

2.2 Reward Model Training

Despite the extensive work done by the contractors, it remains vanishingly unlikely that the LM will generate a word-for-word copy of contractors' responses to user queries under typical usage conditions.

This motivates the use of Reward Modelling to predict the quality of the LM's responses *in general*, based on the contractors' feedback on a small set of responses. (See Section 2.4 for specifics about the Reward Model.)

The Reward Model hence serves as an estimate of the quality of the LM's responses, and is trained on the contractors' rankings, which are converted into a reward signal to be used during the next step of the RLHF pipeline.

2.3 How RL powers RLHF

Finally, the Reward Model is used to train the LM via Reinforcement Learning (RL); specifically, Proximal Policy Optimization (PPO) [1] (Schulman et al. (2017)), where the Reward Model contributes principally to the environment's reward signal.

Unlike the Markov Decision Process (MDP) framework used in previous weeks, Ouyang et al. (2022) formulate the problem as a single-step Armed Bandit problem (Sutton & Barto (2018)). See Table 1 for a more detailed description of the RLHF environment.

Component	**Description**
Observation	The user query.
Action	The LM's response to the user query.
Reward	The ”preference” score of the LM's response. Specifically, $r = r_\theta(x, y) - \beta \log(\pi_\phi^{RLHF}(y)), \beta > 0$
Transition Dynamics	Episodes end immediately after the LM completes its response (i.e. 1 action only). Rewards are disbursed via r.

Table 1: Description of InstructGPT's RLHF environment.

By maximising the expected reward, the LM hopefully learns to generate responses which are more ”preferred” by humans in general, while maintaining the robust sequence modelling capabilities it has learned during the pre-training and SFT phases.

[1] Proximal Policy Optimization (PPO) is a popular on-policy, model-free, policy-gradient RL algorithm, and a simplification of Trust Region Policy Optimization (TRPO) (Schulman et al. (2015; 2017)). PPO is best known for its ”clipping” mechanism, which constrains the policy updates such that action distributions do not change too much from update to update. This helps to stabilize training and prevent large, disruptive updates (Schulman et al. (2017)).

2.4 Aside: The Bradley-Terry Model

Ouyang et al. (2022), in aiming to follow the methodology developed in Ziegler et al. (2020) and Stiennon et al. (2020), use *pairwise preferences* to train their Reward Model.

Some sleuthing reveals that this is likely based on the Bradley-Terry model (Bradley & Terry (1952)), which is a statistical model for paired comparisons. For concision, we do not go into the details of the model here, but offer a few sources for the interested reader (Bradley & Terry (1952); Stiennon et al. (2020); Fujii (2024)).

3 Interesting Findings

3.1 InstructGPT vs GPT-3-175B

Perhaps the most interesting finding of Ouyang et al. (2022) is the outsized impact of the RLHF pipeline on the LM's ”utility”. Despite having only 1% of the parameters of GPT-3-175B, and being tuned on just a bit more data (∼100k contractor and user-generated prompts), InstructGPT was overwhelmingly preferred by human evaluators over GPT-3-175B.

This cements RLHF's status as a powerful technique for improving LMs' responses to user queries, and confirms that LMs' *utility* is not as strongly correlated with their size, training set size, or benchmark performance as previously thought.

3.2 Reward Model Generalization

Ouyang et al. (2022) found in a cross-validation study that the reward model was able to generalize well to prompts from unseen contractors. When splitting labelers into 5 folds, Ouyang et al. (2022) found that inter-fold prediction accuracy ($(72.4 \pm 0.4)\%$) (i.e. How often does the model correctly predict the preferred response of a contractor in the test fold?) was higher than intra-fold accuracy ($(69.6 \pm 0.4)\%$) (i.e. How often does the model predict the preferred response of a contractor in one of the training folds?).

This suggests that, while obviously not perfect, the reward model gives a relatively good approximation of the contractors' preferences in general. It remains to be determined if the contractors are representative of the general population.

3.3 RLHF May Not Be A Pareto Improvement

While RLHF so far has appeared to be a very powerful technique so far, Ouyang et al. (2022) found that there may be a trade-off between benchmark performance and human preference. Though Ouyang et al. (2022) tried to mitigate this by incorporating update gradients from the Pre-Train + SFT model, Ouyang et al. (2022) still noticed a very slight drop in performance across several benchmarks.

It is not clear if this is a fundamental limitation of the RLHF pipeline, or if it is simply a result of the limited amount of data or flawed reward model used in the study.

With the benefit of newer interpretability studies, this phenomenon may be justified and explained through certain learned behaviors like sycophancy (Sharma et al. (2023)), where the LM agrees with the user without regard for the truth or helpfulness of the response.

On the other hand, a different RLHF setup due to Zheng et al. (2024) demonstrates RLHF's potential to improve humans' preference to LMs' responses as well as their performance on benchmarks, suggesting that the trade-off may not be fundamental, and that it is possible to achieve a Pareto improvement with RLHF.

4 Future Directions

Having read the paper and analyzed its findings, here are the relatively promising research directions documented in the paper, as well as a few which emerge from cross-referencing other papers.

While these are but a subset of the paper's own proposed future directions, it must be noted that since the paper's publication, RL post-training has progressed significantly, and many of the directions proposed in the paper have been well-explored in subsequent works.

4.1 Improving Reward Modelling

The Reward Model used in InstructGPT appears, at first glance, to be well-grounded in statistical theory. This, however, does not preclude the possibility of having better Reward Model designs for *general-purpose* RL with feedback.

An example of what good Reward Modelling could do would be Reinforcement Learning with Verifiable Rewards (RLVR) (Lambert et al. (2025)), whose potential in improving LM performance on math, computing, and reasoning tasks has been adequately demonstrated by DeepSeek-AI et al. (2025) in Deepseek-R1.

However, it is unclear if there is a strictly better way to model reward signals for RL with feedback *without the need to restrict post-training domains*.

4.2 Scaling RL With (Not Necessarily Human) Feedback

While RLHF is clearly more scalable than other methods like manual SFT and soliciting human feedback on every possible output of the LM, LMs have improved significantly since the paper's publication, and it is possible that human annotations for the LM's responses can be augmented with these more powerful LMs.

A particularly interesting example of this is Anthropic's Constitutional AI framework (Bai et al. (2022)), which leverages AI feedback to improve the harmlessness of LMs. Their approach demonstrates the potential for using advanced LMs to enhance the quality and safety of AI-generated content at massive scales.

4.3 Is RL A Pareto Improvement?

As discussed in Section 3.3, it appears that RLHF may not be a Pareto improvement, and indeed, there appears to be little consensus / exploration (Lin et al. (2024); Berg et al. (2024)) on whether RLHF approaches, if designed well, *DO* meaningfully improve LMs' performance on benchmarks and human preference. Moreover, even if they do, it is not clear *what* the underlying mechanisms behind this improvement are.

5 Conclusion

At the end of this report, we have analyzed the InstructGPT paper by Ouyang et al. (2022), and noted down numerous interesting findings, as well as potential future directions from the paper (and other works) pertaining to RLHF research, as well as RL post-training for LMs in general.

References

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam

Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional ai: Harmlessness from ai feedback, 2022. URL https://arxiv.org/abs/2212.08073.

Cameron Berg, Judd Rosenblatt, Diogo de Lucena, and AE Studio. The case for a negative alignment tax, 2024. URL https://www.lesswrong.com/posts/xhLopzaJHtdkz9siQ/the-case-for-a-negative-alignment-tax.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. ISSN 00063444, 14643510. URL http://www.jstor.org/stable/2334029.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. URL https://arxiv.org/abs/2005.14165.

Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences, 2017. URL https://arxiv.org/abs/1706.03741.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Satoru Fujii. Neural bradley-terry rating: Quantifying properties from comparisons, 2024. URL https://arxiv.org/abs/2307.13709.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. Realtoxicityprompts: Evaluating neural toxic degeneration in language models, 2020. URL https://arxiv.org/abs/2009.11462.

Andrej Karpathy. Deep dive into llms like chatgpt, 2025. URL https://www.youtube.com/watch?v=7xTGNNLPyMI.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, Noah A. Smith, Yizhong Wang, Pradeep Dasigi, and Hannaneh Hajishirzi. Tulu 3: Pushing frontiers in open language model post-training, 2025. URL https://arxiv.org/abs/2411.15124.

Yong Lin, Hangyu Lin, Wei Xiong, Shizhe Diao, Jianmeng Liu, Jipeng Zhang, Rui Pan, Haoxiang Wang, Wenbin Hu, Hanning Zhang, Hanze Dong, Renjie Pi, Han Zhao, Nan Jiang, Heng Ji, Yuan Yao, and Tong Zhang. Mitigating the alignment tax of rlhf, 2024. URL https://arxiv.org/abs/2309.06256.

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 Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022. URL https://arxiv.org/abs/2203.02155.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners, 2019. URL https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf. OpenAI Blog.

John Schulman, Sergey Levine, Philipp Moritz, Michael I. Jordan, and Pieter Abbeel. Trust region policy optimization, 2015. URL https://arxiv.org/abs/1502.05477.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch, Nicholas Schiefer, Da Yan, Miranda Zhang, and Ethan Perez. Towards understanding sycophancy in language models, 2023. URL https://arxiv.org/abs/2310.13548.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback, 2020. URL https://arxiv.org/abs/2009.01325.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018. URL http://incompleteideas.net/book/the-book-2nd.html.

Chen Zheng, Ke Sun, Hang Wu, Chenguang Xi, and Xun Zhou. Balancing enhancement, harmlessness, and general capabilities: Enhancing conversational llms with direct rlhf, 2024. URL https://arxiv.org/abs/2403.02513.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences, 2020. URL https://arxiv.org/abs/1909.08593.