# Explain Like I'm Five: Leveraging Successful Online Interactions for Finetuning LLMs for Education

**Vuk Vukovic, Julian Schnitzler, Veniamin Veselovsky**
firstname.lastname@epfl.ch

## Abstract

Education-focused Large Language Models (LLMs) have the potential to revolutionize the educational landscape by providing personalized learning and increased accessibility. However, the use of proprietary models not specifically designed for education has limited their effectiveness in addressing the specific needs of students. In this paper, we explore the benefits of fine-tuning LLMs for educational purposes and compare their performance with ChatGPT, a widely used language model. We leverage the rich knowledge of the Web to create an implicitly-derived human preference dataset from social media, which we term `SocialPref`, and then employ parameter-efficient supervised fine-tuning methods to fine-tune an LLM as an effective tutor. Our results demonstrate that our fine-tuned LLM qualitatively outperforms both Alpaca-base and ChatGPT in providing effective explanations and generating preferred responses; however, is less correct than ChatGPT. Code/data: https://github.com/CS-552/project-m3-v2j-vectors-to-jokes.

## 1 Introduction

Education-focused Large Language Models (LLMs) are transforming the educational landscape. These models have started to pervade various aspects of education, fostering a unique blend of personalized learning and broad accessibility, thus opening the door to democratizing education. The implications for education are profound, offering each student access to a personal, artificial intelligence tutor, which research has shown leads to an improved understanding of concepts, better performance, and increased enjoyment among students (Reber et al., 2018; Gao, 2014; Fok and Ip, 2004).

One issue, however, has been the general use of proprietary models that are not trained to act as an effective tutor. This may not be ideal for the educational context for students with very specific needs. Overall, there has been little exploration into how finetuning models for the purpose of education can improve the explanations over these proprietary models. In this paper, we aim to fill this gap by answering two research questions: **RQ1:** Can an LLM fine-tuned on an education corpus outperform ChatGPT? **RQ2:** When does ChatGPT outperform our smaller-capacity model?

We illustrate that by leveraging social media content and user upvotes and downvotes we can finetune an LLM to effectively answer student questions that qualitatively outperforms ChatGPT. In particular, we developed a dataset of question-answer pairs on Reddit and used upvotes as a proxy for preference. We then combined this dataset with synthetic data from ChatGPT to finalize our human preference dataset called `SocialPref`. Additionally, since finetuning these LLMs is expensive, we conducted Parameter-Efficient Fine-tuning (PEFT). These approaches enable efficient adaptation of pre-trained language models to various downstream applications without fine-tuning all of the model's parameters. Instead, only a small number of additional parameters (or model parameters) are fine-tuned, while the majority of the pretrained LLM parameters are frozen.

The main contributions of our paper include:

- An approach for leveraging the rich knowledge of the Web when training a preference model alongside the dataset (`SocialPref`) used for training.
- Operationalization of LoRA (Hu et al., 2021) on LLaMA for the education context.
- A novel evaluation metric using a pre-trained reward model to automatically assess the quality of interactions without the need for manual, human labor
- A rigorous evaluation comparing ChatGPT,

our model, and Alpaca-base across quantitative and qualitative metrics.

## 2 Related Work

**Large Language Models.** In recent years, a plethora of pre-trained Large Language Models (LLMs) were released. Models like GPT-3 (Brown et al., 2020) or GPT-4 (OpenAI, 2023) presented new state-of-the-art results on many benchmarks. Yet, most of these models and their parameters are kept confidential by their respective owners, making the creation of similarly performing models a challenge. LLaMMa (Touvron et al., 2023) was the first large-scale model that was initially released for an academic context, but was later publicly disclosed.

**Reinforcement Learning and LLMs.** While models like GPT-3 were introduced already several years in the past, the recent peak in interest is strongly connected to the combination of LLMs of this size with Reinforcement Learning from Human Feedback (RLHF) (Ziegler et al., 2020), which, unlike earlier fine-tuning approaches on a single correct phase, takes the ambiguity of the task of answering questions into account and rather train a reward model and combining this with human input to order several output candidates. This approach led to the introduction of InstructGPT (Ouyang et al., 2022), which was trained to follow human instructions.

**Finetuning and Imitation.** Several initiatives have shown that it is possible to perform on a similar level compared to large models trained by companies with access to almost unlimited resources, by finetuning versions of Llama (Touvron et al., 2023). For example, the authors of Alpaca (Taori et al., 2023) show how to fine-tune Llama to become an instruction-following language model.

However, this doesn't come without a cost. Typically, when LLaMA is finetuned, it's trained on the exhaust of other LLMs like ChatGPT. For example, Vicuna was trained using 70,000 user outputs of ChatGPT (Chiang et al., 2023). Recent papers, however, have shown that while this leads models to be stylistically complex, they aren't as capable as the underlying model and perform poorly on tasks they aren't finetuned on (Gudibande et al., 2023). Similarly, Shumailov et al. (2023) found that when models are trained on outputs from other models, they have an increased chance of "forgetting" information.

| Source | # Samples |
|---|---|
| Reddit | 1168 |
| EPFL | 483 |

Table 1: Distribution of samples in `SocialPref`

**Low-parameter finetuning.** Several papers deal with the problem of reducing the required resources for training or finetuning LLMs. In (Hu et al., 2021), the authors show how to reduce the number of trainable parameters for fine-tuning GPT-3 by a factor of 10'000 by freezing pre-trained weights and decomposing weight matrices into smaller rank decomposition matrices and updating those.

**Better prompting.** Recent work also showed improvements when prompting models like ChatGPT in certain ways. For example, (Wei et al., 2023) show how to improve the reasoning capabilities of a PaLM model by prompting the model with a few examples with more extensive reasoning. The authors of (Madaan et al., 2023) improve reasoning capabilities by presenting an initial solution back to the same model and asking it for feedback to improve its solution.

## 3 Data & Methodology

### 3.1 Data

In this section, we describe the datasets used to train our reward and generative models. Specifically, we utilize two primary datasets to distinguish instances of effective and less effective explanations. We then merge these two datasets into a unified one, termed `SocialPref`. The individual contribution of each file can be found in Table 1. In total, the `SocialPref` dataset comprises 1651 samples.

**Social media.** We use questions, answers, and upvotes on Reddit as a source of effective and ineffective answers to specific questions. Specifically, we collected all submissions and comments from the subreddit `r/explainlikeimfive` between January 2018 to November 2018. In this subreddit, individuals ask questions and others explain the questions as if to a five-year-old—in other words intuitively and understandably for everyone. When users respond to a question, others can upvote or downvote the answers which acts as an implicit proxy of preference for the answers. Fur-

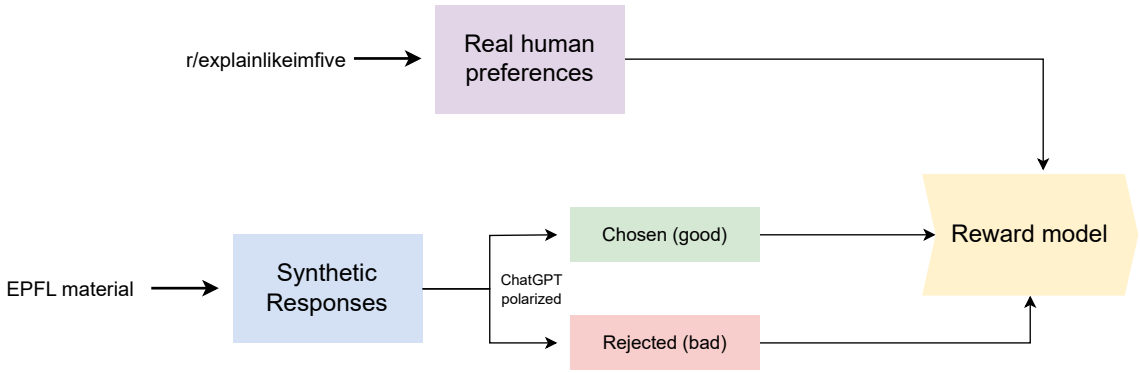


Figure 1: Illustration of our approach for training our reward model. First, we capture real human preferences for explanations through Reddit question answers and upvotes from `r/explainlikeimfive`. Second, we use synthetically generate good and bad answers from a given corpus of EPFL course material. Finally, we combine these two datasets into training our reward model.

thermore, this community has strict moderation rules that are followed by the users, and if the rules are not followed, the comment is removed.

Blindly using comments can be problematic since there can be many reasons why people upvote a comment. Consequently, we take the following pre-processing steps: (1) We limit to all submissions with at least 100 upvotes. (2) Pre-process text to filter answers to be at least 30 characters and at most 100 characters. (3) Remove bot accounts (simple keyword filtering). (4) We quantize answers into four buckets based on upvotes. We then select the best answer, a random comment from the top quartile, the worst answer, and a random comment from the bottom quantile (2 good, and 2 bad) for each submission. In the end, this process results in a dataset of 1168 pairs of good and bad answers to specific questions.

**EPFL course questions.** The second dataset used was based on interactions received from EPFL's dataset. This included thousands of examples of questions from EPFL's bachelor and master's programs, alongside ChatGPT answers to these questions. Students then rated these generations using a confidence score. To generate good and bad answers to these questions, we used ChatGPT to generate two solutions for a sample of 600 questions, one chosen and one rejected. We add these prompts to the appendix. For generation, we use the default `GPT-3.5-turbo` parameters.

### 3.2 Model training

**Preference model.** We use the reward model developed by OpenAssistant (Köpf et al., 2023), fine-tuned on the `SocialPref`. This model is based on DeBERTaV3 and was pretrained on multiple datasets to predict which generated answer is better judged by a human, given a question. For training, a random 80%-20% train-test split was performed on the dataset, with a training batch size of 4. The model was trained using the AdamW optimizer with a learning rate of $2e^{-6}$. We also used a custom loss function:

$$L = -\log(\sigma(x_c - x_r))$$

Where $x_c$ is the chosen responses, $x_r$ is the rejected one, and $\sigma$ represents the sigmoid function. To assess performance, we used accuracy as the evaluation metric, which is the ratio of datapoints for which the score of the chosen answer is greater than the score of the rejected answer. For the baseline comparison, we are using the original OpenAssistant/reward-model-deberta-v3-large model.

**Generative model (tutor).** To train our generative model (tutor), we finetune LLaMA 7B (Touvron et al., 2023) using low-rank adaptation (LoRA) (Hu et al., 2021) on our `SocialPref` dataset. LoRA permits the finetuning of these large models by freezing the pre-trained weights and injecting trainable rank decomposition matrices between the layers. This approach has been shown to achieve comparable results, at a fraction of the computation cost (Zhang et al., 2023). We train on one 40GB A100 GPU and include the following hyperparameters: batch size of 32, 5 epochs, learning rate of $3e^{-4}$, and a LoRA dropout of $0.05$.[1]

[1]The full list of hyperparameters are available at https://github.com/CS-552/project-m3-v2j-vectors-to-jokes/blob/main/src/gen_training.md.

### 3.3 Evaluations

We conduct three sets of evaluations. One evaluation of our reward model and two evaluations to measure how well our model is able to generate explanations.

**Reward model evaluation.** We evaluate our reward model in a paired setting, where we conduct a forward pass for both the rejected and chosen sample and then measure the accuracy for when the model scores higher the chosen answer. More information about the loss function and training setup are presented above; see Section 3.2.

**Generative model evaluation using reward model.** The trained reward model is capable of discriminating between chosen and rejected samples. Consequently, we run this reward model across our four different settings as a proxy for how likely the given given generations are to be "chosen". We then report the average score across our two testsets.

**Generative model evaluation using human.** We took a random sample of 10 questions and then generated answers using (1) our model, (2) ChatGPT, (3) Alpaca-base. The authors then went through each question and rated their preference across the three generations selecting the best explanation. Afterwards, we measure which model outputs were most-preferred by us. Additionally, here we also measured the accuracy of the models by seeing how many of them are correctly answered.

## 4 Experiments

### 4.1 Reward model

We begin by reporting the performance of our model on the task of comparing two sentences, i.e. relatively rating two interactions, correctly. We evaluate our reward model on a subset of `SocialPref`, which we separate from our training data before to ensure that we evaluate the model on unseen data. Hence, as with all interactions in `SocialPref`, the dataset consists of a question, and then two types of interactions, a relatively better one which we denote as "chosen", and a relatively worse one which is referred to as "rejected". For evaluation, we then compare the reward scores for each of the interactions, count the number of chosen/rejected pairs for which the model gives the correct relative scoring, and report the accuracy.

Our reward model, which is the OpenAssistant reward model finetuned on `SocialPref` minus our test set, reaches up to 76 % accuracy, as compared to 65 % for the initial OpenAssistant reward model, which was trained on a similar dataset with different interactions. This shows that for our reward model is more reliable to rate interactions as would be taking the one by OpenAssistant, yet still, about 24 % of interactions are wrongly classified. We assume that this difference is negligible since on expectation three fourth of the training samples would push our final model toward the desired preference.

| Model | Accuracy |
|---|---|
| Our reward model | 76 % |
| OpenAssistant | 65 % |

Table 2: Accuracy of reward models on our testset

| Model | Our Testset | MNLP Testset |
|---|---|---|
| ChatGPT | **3.77** ±1.72 | **3.61** ±1.44 |
| Our model | 2.64 ±1.35 | 2.47 ±1.35 |
| Only Synthetic | 2.06 ±1.44 | 2.40 ±1.40 |
| Alpaca | 0.33 ±1.24 | 0.38 ±1.09 |

Table 3: Average Reward and Standard Deviation on Testset for multiple Models.

### 4.2 Generative model

**Reward model evaluation.** As described in section (3.3), as an empirical evaluation, we compare the average reward on both our test set and the test set offered by the instructors of the MNLP course. Please refer to Table 2 for results.

We can observe that ChatGPT performs best on both test sets, relatively closely followed by our model. For baseline comparison, the Alpaca model which was finetuned on an instruction dataset performs significantly worse with respect to our reward model. ChatGPT achieves on average a reward of 3.77 on our test set and 3.61 on the test set given by the instructors, while our model achieves 2.64 and 2.47, respectively. Alpaca, on the other hand, scores 0.33 and 0.38. While this result does not allow us to compare the quality linearly, we can deduce that on average, the reward model would give a higher score to answers produced by ChatGPT or our model compared to answers given by Alpaca.

Finally, we conducted an ablation study by removing the Reddit comments and training a model

only on the synthetic ChatGPT answers to the questions. Here we find that the model performs worse than also having the Reddit data across both tasks. We include a brief discussion about this in the Appendix (A.2).

**Qualitative evaluation.** Here we report the results of the qualitative evaluation of our generative model. Here we randomly selected questions and asked each of the three models to answer the question. Two of the co-authors then went through the results and manually annotated which one they preferred. We find that our model was preferred 11 times, ChatGPT was preferred 8 times, and Alpaca was preferred once. So while the reward model usually preferred ChatGPT, human evaluation preferred our model.

This qualitative analysis reviewed a few idiosyncrasies that these different models possessed. The first observation we made about ChatGPT was that the responses were overly verbose. Usually, we had to read through a large block of text before getting the answer we cared about. ChatGPT did perform well, however, when there were long computations that were required to answer the questions. On the other hand, Alpaca rarely provided a helpful explanation but instead responded with the value. For example, when asked a simple addition math problem, Alpaca replied with a wrong numerical value, with no explanation. We found that our model was a nice in-between. It wasn't as verbose as ChatGPT, but still provided helpful explanations, unlike Alpaca.

We note that ideally, we would have more people rank preferences, as one person can potentially bias outputs. To avoid some of this, we implemented a blind rating, where the annotator did not know which model the generation was coming from.

Additionally, we evaluated the accuracy of the different models on ten randomly selected questions (which all happened to do with mathematics). Here we find that Alpaca and our model was right 0% of the time, whereas ChatGPT was correct 50%.

## 5 Discussion & Conclusion

As LLMs become increasingly integrated into the fabric of our society, it will be increasingly important to train them in a way that's beneficial for specific downstream tasks. In this paper, we illustrate one approach to doing this. By utilizing existing online texts that teach and explain content, alongside upvotes as an implicit notion of preference, we present an easily scalable way to create a reward model for evaluating LLM performance alongside a dataset for parameter-efficient finetuning. Using our training approach, we found that the model is capable of performing on both the reward model and in qualitative analysis. In fact, by manually annotating the responses for preference, we found that our model outperforms ChatGPT most of the time; however, still lacks correctness compared to ChatGPT. It's worth noting, that our model included only 7B parameters, whereas ChatGPT is based on a model (GPT-3) with 175B parameters.

We believe that our general approach is highly scalable to applications beyond education. For the scope of this paper, we limited our analysis to the education space; however, there are other possible examples of where our infrastructure can work, here we list one example: Mental health bots. Reddit consists of hundreds of communities dedicated to self-help and counseling. By correctly filtering for "healthy" comments, we believe it can guide the model to act as a form of mental health assistant. Another added advantage of our approach is that we no longer rely on human annotated datasets from crowd workers, as was done in training previous reward models (Taori et al., 2023). Recent work has shown that we can no longer be certain that humans are the ones annotating data on crowd working sites Veselovsky et al. (2023b).

However, all this being said, there are a few limitations that are important to mention about our approach. First, our preference model was trained implicitly to prefer the texts of ChatGPT. This is problematic since then the model is trained to like the text that ChatGPT produces, which biases the empirical scores found in Table 3 in favor of ChatGPT — leading to inflated scores. For the next steps, it makes sense to have more human input when training the preference model. Second, our qualitative analysis section included only two annotators. In general, it would make more sense to hire a collection of students who all rank their preferences and describe why, and then report annotator agreement scores. Otherwise, we (co-authors) may be biased by understanding how the different outputs from the different models appear. Third, we don't conduct a rigorous examination of ablations. Our dataset consists of two parts (synthetically polarized answers from EPFL course questions) and social media, but it would be beneficial to explore

the role each of these datasets plays in the performance of the model. We began making progress in this direction by finetuning a model without the Reddit dataset, but didn't have time to check the evaluation. The final limitation we see is that we are currently doing supervised fine-tuning instead of RLHF, which may lead to better results.

## 6 Legal & Ethical Considerations

**Reddit dataset.** We used the Pushshift API to access publicly avaiable Reddit data. However, in the time of collecting the data, Pushshift has taken down these archives and now the data is only available through Reddit's API. This considered, all the data we use is publicly available and so we don't see any legal or ethical considerations in using these datasets.

**ChatGPT.** Using ChatGPT to create synthetic data seems like a legal grey area. In the terms of use for ChatGPT by OpenAI, it says "You may not [...] (ii) reverse assemble, reverse compile, decompile, translate or otherwise attempt to discover the source code or underlying components of models, algorithms, and systems of the Services (except to the extent such restrictions are contrary to applicable law); (iii) use output from the Services to develop models that compete with OpenAI;". Whether or not our models compete with OpenAI and whether we in some way reproduce underlying components of the model might be open to debate, however, since we do not work for profit and do not intend to use the models outside of the class project, this should be acceptable.

## References


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

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Apple WP Fok and Horace HS Ip. 2004. Personalized education: An exploratory study of learning pedagogies in relation to personalization technologies. In *Advances in Web-Based Learning–ICWL 2004: Third International Conference, Beijing, China, August 8-11, 2004. Proceedings 3*, pages 407–415. Springer.

Pengyu Gao. 2014. Using personalized education to take the place of standardized education. *Journal of Education and Training Studies*, 2(2):44–47.

Arnav Gudibande, Eric Wallace, Charlie Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, and Dawn Song. 2023. The false promise of imitating proprietary llms.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *CoRR*, abs/2106.09685.

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*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback.

OpenAI. 2023. Gpt-4 technical report.

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. 2022. Training language models to follow instructions with human feedback.

Rolf Reber, Elizabeth A Canning, and Judith M Harackiewicz. 2018. Personalized education to increase interest. *Current directions in psychological science*, 27(6):449–454.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Yarin Gal, Nicolas Papernot, and Ross Anderson. 2023. The curse of recursion: Training on generated data makes models forget.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Veniamin Veselovsky, Manoel Horta Ribeiro, Akhil Arora, Martin Josifoski, Ashton Anderson, and Robert West. 2023a. Generating faithful synthetic data with large language models: A case study in computational social science. *arXiv preprint arXiv:2305.15041*.

Veniamin Veselovsky, Manoel Horta Ribeiro, and Robert West. 2023b. Artificial artificial artificial intelligence: Crowd workers widely use large language models for text production tasks.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models.

Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. 2023. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2020. Fine-tuning language models from human preferences.

# A Appendix

## A.1 Prompting

Chosen instruction prompt:

> Act as a science teacher that answers the question correctly and provides a short reasoning and explanation to improve student's learning experience. Answer in the following format: The answer is/answers are ... because ...

Rejected instruction prompt:

> Act as a bad science teacher that answers the question by providing bad reasoning, incorrect answers or the reasoning that students cannot understand. As a bad teacher, you are not improving student's learning experience, but you should still sound professional. Answer in the following format: The answer is/answers are ... because ...

## A.2 Ablations

In addition to training our model on both the ChatGPT polarized text and Reddit, we also trained just on the ChatGPT text. Here we find that the model performs signifcantly worse from the model trained *with Reddit texts*. We believe that this is due to a lack of diversity in the ChatGPT generations as was found in previous work (Veselovsky et al., 2023a).

## A.3 Contributions

**Vuk**: Research of Reward and Generative Models; Creation of Fine-tuning Pipeline and Execution for both Reward Model and Generative Model
**Julian**: Collection & Preprocessing of ChatGPT interactions for Synthetic responses; Quantitative Evaluation Pipeline; Paper Writing, Qualitative Evaluation
**Veniamin**: Collection & Preprocessing of Reddit Dataset; Majority of Paper Writing and Ideation, Qualitative Evaluation