Humor in AI: Massive Scale Crowd-Sourced Preferences and Benchmarks for Cartoon Captioning

Jifan Zhang[1*], **Lalit Jain**[2*], **Yang Guo**[1*], **Jiayi Chen**[1†], **Kuan Lok Zhou**[1†], **Siddharth Suresh**[1], **Andrew Wagenmaker**[2], **Scott Sievert**[1], **Timothy Rogers**[1], **Kevin Jamieson**[2], **Robert Mankoff**[3], **Robert Nowak**[1]
[1]University of Wisconsin-Madison, [2]University of Washington, Seattle,
[3]Air Mail and Cartoon Collections
lalitj@uw.edu, {jifan,yguo}@cs.wisc.edu

Abstract

We present a novel multimodal preference dataset for creative tasks, consisting of over 250 million human ratings on more than 2.2 million captions, collected through crowdsourcing rating data for The New Yorker's weekly cartoon caption contest over the past eight years. This unique dataset supports the development and evaluation of multimodal large language models and preference-based fine-tuning algorithms for humorous caption generation. We propose novel benchmarks for judging the quality of model-generated captions, utilizing both GPT4 and human judgments to establish ranking-based evaluation strategies. Our experimental results highlight the limitations of current fine-tuning methods, such as RLHF and DPO, when applied to creative tasks. Furthermore, we demonstrate that even state-of-the-art models like GPT4 and Claude currently underperform top human contestants in generating humorous captions. As we conclude this extensive data collection effort, we release the entire preference dataset to the research community, fostering further advancements in AI humor generation and evaluation.

1 Introduction

This paper presents a dataset and benchmark for investigating alignment in Large Language Models (LLMs). Our dataset contains over a quarter of a billion human ratings from the New Yorker's cartoon caption contest. Writing funny captions presents significant challenges due to the subjectivity of humor and variability in human judgments. This benchmark offers a unique challenge for AI alignment, reflecting complexities found in tasks where expert humans consistently outperform current AI systems. Our study examines fundamental questions about aligning them to generate funny captions similar to the winning captions that are most highly rated by the New Yorker readers.

We explore humor expression in LLMs, investigating whether these models can recognize humor and generate amusing captions that resonate with human audiences. While LLMs are not specifically designed for humor, their training on diverse content suggests a potential for humor recognition and expression. We propose a benchmark for evaluating a model's humor capabilities using advanced systems like GPT-4.

Our empirical analysis shows that current LLMs can generate humorous captions but significantly underperform compared to high-ranking human submissions in the New Yorker's caption contests. Generating successful captions requires multiple advanced capabilities: understanding of cultural references, recognition of humor patterns, logical reasoning, systematic planning, and visual analysis. The multi-component nature of caption generation makes this benchmark an effective test of broad LLM capabilities. Progress in aligning LLMs for this task will require both advancing these individual capabilities and developing methods to integrate them effectively. This benchmark therefore provides a comprehensive integration test for LLM capabilities.

[*]Equal contribution.
[†]Equal contribution.

38th Conference on Neural Information Processing Systems (NeurIPS 2024) Track on Datasets and Benchmarks.

 https://doi.org/10.52202/079017-3978

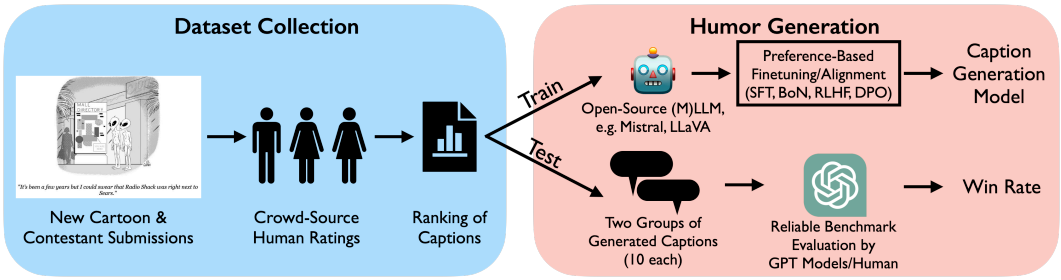


Figure 1: Overview of our workflow. During data collection, a new cartoon is released each week and thousands of captions are submitted. We then collect caption ratings through a crowd-sourcing procedure driven by a bandit algorithm. Our dataset is a collection of 365 contests, over 2.2M captions and over 250M human ratings. This dataset is utilized for our Humor generation task and benchmark. We experiment with finetuned open-source models and close-sourced API calls (both LLMs and MLLMs). Our novel and low-cost evaluator provides better reliability in evaluating captions.

Our main contributions and findings are:

Dataset: We present a large-scale dataset of human-rated cartoon captions from The New Yorker's weekly contest. Each week, the New Yorker hosts a contest with a new cartoon, where thousands submit their funny captions. Hundreds of thousands of ratings are collected for each contest, and the winning captions are determined by those receiving the highest ratings. This dataset enables researchers to explore humor generation in LLMs and represents the first large-scale dataset with human judgments for evaluating creative tasks. With over 250 million ratings, it offers diverse examples for studying humor expression and perception in AI systems.

Benchmark: We introduce new metrics for evaluating humor quality in LLM-generated content, using GPT4 and group based techniques. These metrics provide a standardized framework for assessing AI-generated humor. Our benchmark allows for systematic comparisons between human and AI-generated humor.

Evaluation of State-of-the-Art Models: We assess the performance of models such as GPT-4 and Claude in generating humorous content, comparing their outputs to human-generated examples. This analysis offers insights into the current capabilities and limitations of LLMs in humor generation, identifying areas of strength and potential improvement.

Alignment Strategy Analysis: We use our benchmark to evaluate various alignment strategies, including Reinforcement Learning from Human Feedback (RLHF), Direct Preference Optimization (DPO), and Best-of-N sampling (BoN). By comparing these strategies, we provide insights into their effectiveness in enhancing humor generation in LLMs and aligning AI systems with human preferences.

In summary, our paper advances LLM capabilities in humor evaluation and generation through a comprehensive dataset, a new evaluation benchmark, and analysis of model performance and alignment strategies. This work enhances our understanding of humor in AI systems and provides a foundation for future research in this field. We open-source our dataset and code as detailed in Appendix A.

2 Related Work

New Yorker Caption Contest. Since its original conception as part of the NEXT crowdsourcing system [25, 47], the New Yorker Caption Contest Dataset has been updated on a weekly basis for the last several years. During this time, the dataset has been primarily used for the evaluation of online algorithms and, similar to this work, to study the nature of humor. Works in the former camp include [34, 52, 59]. Perhaps the most relevant work to ours is [22]. They formulated three tasks, matching, quality ranking and explanation generation for studying whether current AI systems *understand* humor. Additional prior work includes [45, 40, 28], which utilize judgements made by the editors of the New Yorker directly to analyze a smaller number of contests (< 50) and attempt to identify features that correlate with caption performance such as length, perplexity, readability and sentiment.

Alignment of LLMs. Finetuning of LLMs has proved a critical step in aligning the behavior of pretrained models to downstream tasks. A standard pipeline is to first finetune the pretrained model via *supervised fine-tuning* (SFT)—to imitate expert demonstrations—followed by *reinforcement learning from human feedback* (RLHF) [14]—where a reward model is trained on human preferences, and then the SFT model is trained to maximize this reward via PPO [44]. This pipeline has been successfully applied for finetuning frontier models [68, 4, 38, 54], and has inspired a vast amount of follow-up work refining and extending the SFT [63, 64, 20, 35, 18] and RLHF [6, 16, 33, 48, 36, 51, 9, 11] methodologies. *Direct preference optimization* (DPO) methods [41] have recently emerged as a simpler yet still effective replacement to the RLHF paradigm. Instead of training a reward model and then optimizing this reward, DPO combines these steps by directly optimizing the SFT model on offline human preference data, and has inspired a variety of extensions [21, 3, 49, 43, 53, 61]. While the aforementioned works focus on finetuning on human feedback, a related line of works has sought to finetune on AI-generated feedback [60, 57, 30, 8, 13, 62, 5]. Despite extensive research into various fine-tuning methodologies, understanding their effectiveness for creative tasks remains nascent. While several studies have explored when and why different methods are most effective [19, 29, 56, 10, 66, 46], they primarily address standard tasks like reducing harmfulness and increasing helpfulness, and do not assess fine-tuning methods for tasks requiring creativity, the focus of this work.

RLHF Datasets. Existing Reinforcement Learning with Human Feedback (RLHF) datasets, consisting of various responses to a prompt along with a preference ordering of those responses, have been critical for aligning existing AI systems to human preferences. We briefly review some of the most popular ones. Anthropic's HH-RLHF dataset [4] consists of chosen and rejected texts focusing on helpfulness and harmlessness. Stanford's SHP Dataset [17] and Stack Exchange preferences dataset [1] have aggregated questions and answers along with their ratings from various online platforms. OpenAI's summarization dataset [24] includes rankings of paired answers derived from human evaluations of text summaries. The data comes from a variety of sources, such as news articles and scientific papers, where human annotators compare the quality, coherence, and relevance of two different AI-generated summaries for the same text. The WebGPT comparisons [37] offer a dataset of human comparisons of AI-generated web search results, emphasizing the importance of high-quality, relevant information retrieval. Finally, we mention the Nectar dataset [67], which consists of a large series of prompts along with a list of five answers generated by various LLM's along with a ranking of these prompts by GPT-4.

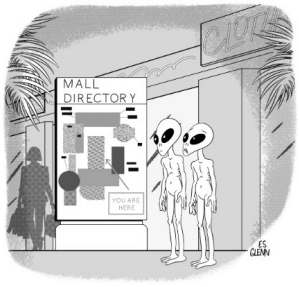
Humor in LLMs. Several recent works have studied humor capabilities of large language models, in addition to the ones studying the New Yorker Caption Contest. Concurrent to our work, Zhong et al. [65] also studies humor in a multi-modal setting, focusing on a different humor game Oogiri. Their experiments also suggests that existing chain of thought techniques are insufficient for LLMs to generate and understand humor. Similarly, Jentzsch and Kersting [26] also shows GPT-3 still lacks humor abilities despite the good performance on other factual knowledge benchmarks. Furthermore, several works have focused on LLMs' capabilities in understanding humor, including humor detection [15], puns [58], and humor explanation [12].

3 New Yorker Caption Contest

Every week The New Yorker publishes an uncaptioned cartoon and solicits humorous captions from its readers through their website. The cartoon editors then review this list of captions and choose the top three funniest ones according to their judgement. The contest began in 2005, and at the time this work was written, there have been roughly 900 contests. For the last eight years, starting with contest 530, the New Yorker has utilized an online crowdsourced rating system (see Figure 2) where users are presented with captions and can rate whether the caption is funny (a reward of 3), somewhat funny (a reward of 2), or unfunny (a reward of 1). Each week a large number of captions are submitted (on average more than 6,000). These captions are first filtered by the New Yorker's editorial staff to remove captions that are not humorous or include personal information and/or offensive content, and then are sent to the crowdsourcing platform for large-scale rating. Finally, the New Yorker editors make their final decisions based on the crowdsourced ratings.

The rating process utilizes a multi-armed bandit-based algorithm, namely a UCB-variant (see [25, 52] and Appendix D for details), to present users with higher-performing captions more frequently in order to efficiently identify the best caption. Additionally, since many of the captions are unfunny,

 https://doi.org/10.52202/079017-3978



"It's been a few years but I could swear that Radio Shack was right next to Sears."

UNFUNNY | SOMEWHAT FUNNY | FUNNY

Figure 2: Example voting page for contest 895

Table 1: Dataset statistics

Number of contests	365
Number of cartoons	365
Average #captions/contest	6044
STD #captions/contest	1794
Total number of ratings	284,183,913
Average #ratings/contest	778,586
STD #ratings/contest	325,156
Max #ratings/contest	2,249,813
Min #ratings/contest	31,173
Average rating	1.214($\pm$0.12)
Top 10 average rating	1.824 ($\pm$0.15)

this keeps the rating engaging by presenting users interesting captions to rate compared to random sampling. On average the contest receives close to 780,000 ratings per week. The top 5% of captions receive an average of 821 ratings, and the bottom 50% of captions receive around 85 ratings.

The crowdsourced voting system for the New Yorker Caption Contest (NYCC) has resulted in an extensive dataset on human preferences and is a key contribution of this work. The dataset can be accessed at `https://huggingface.co/datasets/yguooo/newyorker_caption_ranking`. It consists of the cartoons, captions, and ratings for each one of 365 contests from contests 530 to 895. It provides an extensive labeled dataset on humor for researchers across multiple domains to study. In the related works, we describe some other works that have utilized this dataset. See Table 1 for more dataset statistics.

4 HumorousAI Benchmark: Funny Cartoon Caption Generation

In this section, we establish a benchmark method for evaluating the ability of large language models to generate funny captions. We start by describing the tasks in Section 4.1 followed by our proposed evaluation methods described in Section 4.2. Lastly, in Section 4.3, we give a brief overview of the various finetuning methods we explore in this paper.

4.1 Task

We focus on the cartoon captioning task in this paper, where a model is given the information about the cartoon and is asked to generate funny captions about it. Specifically, we evaluate both multimodel large language models (MLLMs) and language-only models (LLMs). For MLLMs, we provide the raw cartoon images. For language-only models, we instead provide the descriptions and object entities of the cartoons. The text format of these descriptions are either written by human [22] or generated by MLLMs by given the images (see Appendix B.1 for details). See Table 7 for the example descriptions.

We hold out a set of 91 out of the 358 contests for evaluation by an evaluator (see Section 4.2). For each contest and its corresponding cartoon, we ask the language model to generate ten captions. This group of ten captions is then compared against four groups of past human submissions by the evaluator. For each contest, the four groups are captions ranked #1-10, #200-209, #1000-1009 and the ten captions that received median ranking. The evaluations are conducted along three dimensions:

1. **Overall comparison**: In this setting, the evaluator compares the overall funniness of the group of model-generated captions against each group of contestant-submitted captions. Win rates of the model-generated captions will be reported in Section 5 and Table 3.
2. **Best pick comparison**: We ask the evaluator to first pick the funniest caption from each of the two groups and then choose the funnier caption accordingly. Win rates are reported similarly to above.
3. **Caption diversity**: We measure the diversity of captions within each group of captions either generated by language models or submitted by human contestants in the past. Similarly to the study [29] on measuring the output diversity for non-creative tasks (summarization and instruction

Table 2: **Evaluation reliability measure**: Ranking accuracy of captions ranked #1-10 vs captions ranked #1000-1009 averaged over 200 pairs. See Appendix B.1 for details on how the cartoon descriptions are generated.

Comparison Method	Evaluator	Description/Image	Ranking Accuracy(%)
Pairwise	Human (worker)	GPT4o-vision	61.67±3.45
	Human (worker)	Cartoon Image	60.79±3.46
	GPT4-Turbo-vision	Cartoon Image	61±3.46
	GPT4o-vision	Cartoon Image	60.5±3.47
	GPT4o	GPT4o-vision	65±3.38
	GPT4-Turbo	**GPT4o-vision**	**67±3.33**
	GPT4-Turbo	GPT4-vision	66±3.36
	GPT4-Turbo	Hessel et al. [22]	66.5±3.35
Group (Overall)	Human (worker)	GPT4o-vision	59.23±1.45
	Human (worker)	Cartoon Image	57.5.42±1.37
	Human (expert)	Cartoon Image	94.28±2.79
	GPT4-Turbo-vision	Cartoon Image	63±3.42
	GPT4o-vision	Cartoon Image	74±3.11
	GPT4-Turbo	GPT4o-vision	73±3.15
	GPT4-Turbo	GPT4-vision	74±3.11
	GPT4-Turbo	**Hessel et al. [22]**	**77.5±2.96**
Group (Best Pick)	Human (worker)	GPT4o-vision	56 ± 2.22
	Human (worker)	Cartoon Image	63.66±1.96
	GPT4o-vision	**Cartoon Image**	**70.5±3.23**
	GPT4-Turbo	GPT4o-vision	61.5±3.45
	GPT4-Turbo	Hessel et al. [22]	60±3.47

following), we use the expectation-adjusted distinct N-grams (denoted as **Average EAD**) [31] and the Sentence-BERT embedding cosine similarity (denoted as **SBERT**) [42] to measure the per-contest diversity. **Average EAD** measures the token-level similarity of the generated captions, while **SBERT** measures the semantic-level similarity. We do not use the NLI diversity from [50] as it is conversation-specific.

Our evaluation primarily focuses on comparing groups of captions since evaluation reliability can be significantly improved as we now discuss below.

4.2 Evaluation Method

Humor is notoriously subjective. Humans cannot infallibly predict what other humans will find funny. If they could, no joke would ever fall flat. We just do the best we can, always hoping we can do better. Likewise for these models. –Bob Mankoff, former cartoon editor of The New Yorker

In this section, we aim to find a comparably reliable evaluation method for judging model-generated captions against human submissions. We experimented with various versions of GPT-4 and also human evaluations from Prolific [39]. This task has been studied widely before within the context of humor [45, 40, 28, 22]. However, unlike these previous studies that only evaluate two candidate captions at a time (denoted by **Pairwise**), we introduce the novel group comparison techniques for evaluation (denoted by **Group Overall** and **Group Best Pick**). As described in Section 4.1, we compare groups of ten captions from different sources, such as human submissions from different ranking levels, or captions generated by different language models. To measure the reliability of different evaluators, as reported in Table 2, we compare their accuracy in judging human-submitted captions from top #10 versus #1000-1009 across 200 different contests. For the **Pairwise** comparisons, we uniformly at random choose one caption from each of the two groups, which exactly corresponds to the *ranking* task proposed by Hessel et al. [22]. For group comparisons, we provide all ten captions from each group to a single query to an LLM/human rater. The detailed prompts can be found in Appendix B for various language models. All of the prompts for evaluation utilize the 5-shot in-context prompting technique, which provides five caption comparison examples from other contests before asking the model to rank the pair/groups of captions for the given cartoon.

 https://doi.org/10.52202/079017-3978

As shown in Table 2, language models are generally more accurate in detecting the higher-ranked favorable captions in a group comparison paradigm compared to the pairwise paradigm. These models also outperform average humans (crowd workers) in judging the funniness across all three comparison settings. Notably, in the overall group comparisons we also included evaluations from a human expert (the former cartoon editor for The New Yorker). The expert significantly outperforms all other evaluators (AI and human), exposing a significant gap between human experts and SOTA AI systems in this domain. Also, the group comparisons are somewhat more challenging for crowd workers than pairwise comparisons, but group comparisons make the language model evaluations much more reliable and accurate. Further details about the evaluations can be found in Appendix C.

In conclusion, we establish two benchmark evaluation methods for the rest of this paper: **Group Comparison (Overall)** using GPT4-Turbo as evaluator with descriptions from Hessel et al. [22] and **Group Comparison (Best Pick)** using GPT4o-vision as evaluator with raw cartoon images.

4.3 Alignment Finetuning Methods

In our study, we compare the performance of a 0-shot model (with standard and Best-of-N sampling) to that of an SFT finetuned model, an RLHF finetuned model, and a DPO finetuned model. We briefly outline these methods here, and refer the reader to [14, 4, 38, 41] for further details. In all cases, we adopt the implementation from the TRL package [55].

Supervised Finetuning (SFT): SFT assumes access to a dataset $\mathcal{D}_{\text{sft}} = \{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$ of prompt-completion pairs, where $y^{(i)}$ is assumed to be an "expert" completion for prompt $x^{(i)}$. SFT then tunes the weight of the base model to maximize the likelihood of completions $y^{(i)}$ given prompt $x^{(i)}$.

Reinforcement Learning from Human Feedback (RLHF): RLHF assumes access to a preference dataset $\mathcal{D}_{\text{pref}} = \{(x^{(i)}, y_w^{(i)}, y_l^{(i)})\}_{i=1}^{M}$, where $x^{(i)}$ is a prompt, and $y_w^{(i)}, y_l^{(i)}$ two possible completions to $x^{(i)}$, where $y_w^{(i)}$ is preferred over $y_l^{(i)}$. RLHF assumes these preferences are consistent with an (unknown) reward function $r^\star$, typically assumed to follow the Bradley-Terry model [7]. It first trains a reward model $\widehat{r}$ on $\mathcal{D}_{\text{pref}}$, and then finetunes the base language model to maximize $\widehat{r}$, typically running PPO [44] and regularizing the training to ensure it does not deviate significantly from the SFT model.

Direct Preference Optimization (DPO): DPO operates under the same assumptions as RLHF, but skips the reward modeling step entirely, and instead finetunes the base language model on $\mathcal{D}_{\text{pref}}$ directly, tuning it to produce next-token likelihoods with orderings consistent with $\mathcal{D}_{\text{pref}}$.

Best-of-N Sampling (BoN): Best-of-N sampling does not modify the weights of the base model. Instead, it samples N completions from the base model for any prompt x, and chooses the completion with the highest reward, as quantified by the reward $\widehat{r}$ obtained from the RLHF reward-learning step.

Preference Dataset Construction: In our setting, we take $\mathcal{D}_{\text{sft}}$ to be a dataset of cartoon-caption pairs, where the captions $y^{(i)}$ are drawn at random from the entire training set of captions for cartoon $x^{(i)}$. $\mathcal{D}_{\text{pref}}$ is constructed by taking a cartoon $x^{(i)}$ and then two captions $y_w^{(i)}$ and $y_l^{(i)}$, where $y_w^{(i)}$ is set to a caption with a higher human rating then $y_l^{(i)}$. Specifically, we sample the pair to be at least 3 standard deviation apart from each other, i.e.

$$\text{Rating}(y_w^{(i)}) - \text{Rating}(y_l^{(i)}) \geq 3 \cdot \sqrt{\text{STD}(y_w^{(i)})^2 + \text{STD}(y_l^{(i)})^2}, \tag{1}$$

where $\text{Rating}(y)$ is the average score of caption y from human raters according to rewards defined in Section 3 (note this is different from the rewards from the reward model of RLHF). $\text{STD}(y)$ is the corresponding standard deviation of scores from human raters.

5 Experiments

In this study, we evaluate the performance of caption generation. We experiment with two open-source large language models, Mistral 7b Instuct (mistralai/Mistral-7B-Instruct-v0.1)[27] and the multimodal model LLaVa 7b (llava-hf/llava-v1.6-mistral-7b-hf)[32] finetuned with methods in Section 4.3. We also evaluate state-of-the-art close-sourced models including GPT4o and Claude 3 Opus. See Appendix C.3 for more details. Our code is available at `https://github.com/yguooo/cartoon-caption-generation`.

Table 3: Evaluation of captions generated by various language models. We utilize group comparison strategies mentioned in Section 4.2. The generated captions are compared against four groups of human contestant entries at different ranking levels. Win rates are based on 91 held-out cartoons.

Generated Caption Model	Overall Win Rate (%) ↑				Best Pick Win Rate (%) ↑			
	Top 10	#200-#209	#1000-#1009	Contestant Median	Top 10	#200-#209	#1000-#1009	Contestant Median
LLaVA	3.85	2.20	4.40	13.19	2.75	6.59	4.95	12.64
LLaVA SFT	2.75	3.30	7.14	17.03	2.20	4.95	6.59	10.99
Mistral-7B 0-Shot	4.95	8.79	11.54	25.82	1.65	1.65	3.85	12.64
Mistral-7B BoN	6.59	**16.48**	**21.43**	**35.71**	1.65	2.20	3.30	10.44
Mistral-7B SFT	3.85	4.40	7.14	14.29	0.55	2.20	1.65	8.24
Mistral-7B RLHF	8.79	9.34	11.54	24.73	2.20	3.30	8.24	13.19
Mistral-7B DPO	**9.34**	13.74	17.58	31.32	**10.44**	**15.93**	**14.29**	**30.22**
GPT-3.5 Turbo	33.52	52.75	62.09	76.92	23.63	46.7	48.35	70.88
GPT-4o	44.51	69.23	79.12	86.81	42.86	59.89	73.63	79.67
GPT-4o Vision	42.31	63.74	76.92	85.16	**47.80**	**65.93**	**79.67**	**85.71**
Claude-3-Opus	**54.40**	**70.88**	**81.87**	**88.46**	40.11	59.89	63.74	79.67

5.1 Experimental Results

In Table 3, we report the result for pretrained and finetuned model generations evaluated by GPT models. In Table 4, we ask human workers and expert to evaluate the captions generated by SOTA models. Below, we document some of our findings and research questions they inspire.

MLLMs vs LLMs. Surprisingly, language-only models such as the pretrained Mistral model outperform the multimodal LLaVa model that has access to the entire cartoon images. Similarly, for overall group comparison, GPT-4o is also preferred over GPT-4o with vision. To further investigate this issue, we conducted more experiments and obtained the following results:

1. GPT4-Turbo as evaluator given GPT4o descriptions (as reported in Table 2). Accuracy: 67%.
2. GPT4-Turbo-vision as evaluator given cartoon and GPT4o descriptions. Accuracy: 60.5%.
3. GPT4-Turbo-vision as evaluator given a blank image and GPT4o descriptions. Accuracy: 61.5%.

For bullet points 2 and 3 above, we are running the exact same model, the only difference is that one has access to the cartoon + text description, while the other has access only to the text description, thus isolating the effect of the image on the generation quality. We find that the visual element integration into the LLMs is negatively biasing the model's accuracy. This observation is also consistent for overall and best pick group comparisons. Since giving a blank image also hurts performance compared to bullet point 1, GPT4 without vision, it is unlikely that the performance of the vision model is dragged down by the visual understanding capabilities.

One possible reason for the above observation is that the training corpus for multimodal LLMs can be much less diverse than the training corpus for the LLM. For example, LLaVa is only trained on a small multi-modal instruction following dataset (∼80K unique images) [32], whereas generic LLMs like Mistral or Llama are trained on much larger dataset. Overall, these findings suggest there is still much research to be done in better integrating multi-modal capabilities into large language models.

> Proposed Research Question #1: The multimodal large language models still underperform their language-only counterparts in caption generation. Can the vision-language integration in MLLMs be further improved to close this gap?

Finetuning Open Source Models. We observe that supervised fine-tuning hurts the model performance in the humor generation task in general. We believe this is primarily because we are aligning to captions in the top 1000, most of which are not particularly funny. However, we note this is an important step before RLHF and DPO training, as it trains the models to generate captions in the correct format. We also find that BoN sampling is able to substantially increase the Overall Win Rate metric, but falls short on the Best Pick Win Rate, which suggests the reward model is favoring a small set of good captions, but none of which generates particularly outstanding captions. We also observe in the next section that BoN indeed results in a less diverse group of generations.

https://doi.org/10.52202/079017-3978

As compared to BoN, running RLHF on the same reward model is unable to achieve as high a level of performance. As we show in Appendix E, running PPO does indeed yield generations with higher reward score as given by the reward model, and as our BoN results indicate, filtering captions based on their reward does give better performance. This suggests that, while our reward model is able to effectively filter generations, tuning a model to maximize it does not necessarily lead to improved performance. We hypothesize that this is due to the complex nature of humor and the potential for out-of-distribution generations when running RLHF. While our reward model may effectively rank captions within a set of reasonable and in-distribution captions (for example those generated by the 0-shot model), small deviations from the training distribution could lead to an erroneous reward signal. Furthermore, for tasks such as humor generation very subtle changes (for example, minor changes in word choice) can drastically change how humorous a caption is—the distribution of humorous captions is extremely sensitive. Together, we believe these phenomenon make it challenging for PPO to effectively finetune the weights to obtain significantly more humorous generations.

Proposed Research Question #2: Can we train a reward model able to better capture humor? Can RLHF still be effectively applied to settings where the distribution of correct responses is highly sensitive?

In contrast to RLHF, DPO does yield a significant increase over the 0-shot model for the Best Pick Win Rate metric. Note that DPO only optimizes the model on offline preference data and, as such, does not require an evaluation of any out-of-distribution samples. We hypothesize that, in settings such as humor generation where the desired distribution is extremely sensitive, this could lead to better performance, as it avoids the aforementioned issue where RLHF may quickly drift to producing out-of-distribution samples, for which the reward signal is erroneous.

Proposed Research Question #3: Does DPO lead to better in-distribution generation, and produce a model more effectively able to match the distribution of the finetuning data?

Human Evaluation. We also ran a human evaluation using six workers from Prolific [39] along with a humor expert (a former New Yorker editor) to understand how often people preferred caption generations from Claude vs top 10 ranked captions generated by humans. We find that people only prefer Claude's generations 34% of the time. Our expert preferred Claude's generation only 1.6% of the time. He said, *"I think I preferred human captions because from my "expert" vantage point they were better phrased and more concise even independent from being funny. At this point AI tends to be too verbose in almost any task but, for me that is a liability when it comes to creating a good caption."*

Table 4: Rate of Claude-3-Opus generated captions preferred over Human Top 10.

Evaluator	Preference Rate
Human (expert)	**1.6%**
Human (worker)	**35.4%**

This suggests that, though SOTA LLMs can generate a diverse set of funny captions, there remains a significant gap in their humor and creativity when judged from the perspective of human experts.

Example Generation and Qualitative Analysis. As shown in Table 5, we provide some generation samples for the cartoon in Figure 2. Indeed, we see that LLMs generally produce longer and more verbose captions than top human ones. Moreover, we generate multiple additional captions with GPT-4o-vision and Claude-3-Opus for the cartoon in Figure 2. Below, we make a qualitative analysis around the shortcoming of these generated captions from SOTA LLMs.

- **Missing visual details resulting in LLMs generating out-of-context captions.** As an example, GPT-4o-vision generated another caption of *"So the alien abduction statistics were right. Malls are the prime hunting grounds!"* This caption does not match the cartoon though, since the aliens look missing and worried. Their bodies look skinny and weak. In other words, they don't seem to be here to hunt humans.
- **Many generated captions are forms of word/phrase modification and creation.** An example of this caption is **"Do they have a Black Hole Friday sale?"**, also generated by GPT4o-vision. Another example from Claude-3-opus is *"I don't see 'Invasion Supplies' listed anywhere..."* Both of these captions are inventing new words and phrases to make the caption funny. While these can be somewhat funny, they are usually not rated highly by the New Yorker audiences.

- **Winning captions tend to appeal to readers with multiple interpretations through different lenses.** LLMs currently lack the ability to produce such captions. The New Yorker's editorial pick of the final caption was *"Oh sure, now you look at a map."* While this caption makes fun of the aliens blaming each other, it also references the experience of a driver missing the direction but claiming they knew the way. On the contrary, GPT4o-vision generated the caption *"We travel light-years, and we still need directions!"*, which despite making fun of the characters in the cartoon with the same concept, lacks the relatability from an additional perspective.

Overall, we think the GPT4 models are capable of generating humorous content. However, to rank among the top requires much deeper understanding of cultural references and more steps of reasoning before arriving at a high quality caption.

5.2 Diversity Evaluation

We evaluated the token-level and semantic-level diversity of the generation with results given in Table 6. We found the Average EAD and SBERT share the same trend when the base model is the same. Within the human generated caption group, we noticed that their diversity scores are very similar under both metrics. And the human generated texts regardless of their funniness are much more diverse than any model-generated captions.

For pretrained models, the commercial models like GPT, Claude-3 generally outperform the open-source model, like Mistral or LLaVa, in terms of diversity. Introducing the SFT and PPO procedure can moderately improve the diversity metrics for the Mistral model. This is in contrast to the findings of [29], which observed the opposite effect, that RLHF reduced diversity in regular text generation tasks. We also found that running DPO can yield a significant increase in the diversity of the model generations as compared to any other method. We hypothesize that this may be due to our finetuning dataset: for each cartoon, we run DPO with a variety of human-generated captions and it therefore learns not to prefer a single caption or type of caption, but a diversity of captions.

Proposed Research Question #4: DPO exhibits surprisingly good diversity metrics as compared to PPO and SFT. Does the data diversity used for finetuning explain this, or are other mechanisms at play?

6 Future Work and Societal Impact

This paper opens a suite of research problems and challenges going forward and we are excited to continue working on multiple directions of future work.

Improving creativity in LLM generation. While LLMs are largely applauded for their creativity today, our experiments reveal there is still a significant gap between top human generated content and SOTA LLMs and MLLMs, especially when judged by an expert. We believe addressing the proposed research questions can not only improve funny caption generation, but also improve existing models on the creative generation tasks in general.

Gamified evaluation of AI generated captions by a crowd. As the nature of the funny cartoon captioning task is an engaging game by nature, we plan on building an AI versus Human battle ground rating game. Our envisioned game will allow users to submit their own captions. During rating, participants are presented with two sets of captions from different sources (human vs human, human vs AI and AI vs AI). This also provides us a more reliable system for evaluating new captions on new cartoons. At the same time, researchers are encouraged to submit AI model entries to test out their latest model/alignment methods.

Humor vs offensiveness tradeoff. Optimizing for humor abilities may result in increasing offensiveness and toxicity of model generated content. We believe an important next step is to study the challenge of balancing humor with potential offensiveness. As the boundary between humorous and offensive are often blurred, the subjective nature of humor and cultural sensitivities needs to be futher studied to ensure AI models align with human values.

 https://doi.org/10.52202/079017-3978

Table 5: Example caption generations for contest #895 (cartoon in Figure 2)

Mistral-7B 0-shot	When your GPS leads you to the wrong galaxy.
Mistral-7B BoN	What do you call it when aliens invade your favorite mall? A takeover by outer space retailers!
Mistral-7B DPO	I just assumed we were the only ones who knew how to pronounce ‘‘H&M’’.
GPT4-o-vision	We travel light-years, and we still need directions!
Claude-3-Opus	Let’s hit the food court first. I’m craving some Jupiter fries.
Human Winner	Do you think death rays would be considered electronics or sporting goods?

Table 6: Diversity evaluation on the generated captions. We use the expectation-adjusted distinct N-grams (Average EAD) [31] and the Sentence-BERT embedding cosine similarity (SBERT) [42] to measure the per-contest diversity on the token level and semantic level.

Caption Source	Average EAD ↑	SBERT ↑
Human (Top 10)	0.9456	0.7452
Human (#200-#209)	0.9564	0.7496
Human (#1000-#1009)	0.9608	0.7522
Human (Median)	0.9597	0.7489
LLaVA	0.8986	0.5220
LLaVA SFT	0.9002	0.5173
Mistral-7B Instruct 0-Shot	0.9037	0.5349
Mistral-7B Instruct BoN	0.8663	0.4868
Mistral-7B Instruct SFT	0.9043	0.5806
Mistral-7B Instruct RLHF	0.9006	0.5994
Mistral-7B Instruct DPO	0.9206	0.7075
GPT4-o	0.9602	0.5789
Claude-3-Opus	0.9533	0.6813

Acknowledgments and Disclosure of Funding

Funding (financial activities supporting the submitted work): This work was partially supported by the NSF projects 2023239 and 2112471.

Competing Interests (financial activities outside the submitted work): R. Mankoff is formerly the cartoon editor of The New Yorker magazine and currently works with CartoonStock, a commercial cartoon company. L. Jain and R. Nowak formerly provided crowdsourcing services to The New Yorker magazine.

References

[1] Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, et al. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*, 2021.

[2] Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

[3] Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pages 4447–4455. PMLR, 2024.

[4] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

[5] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

[6] Michiel Bakker, Martin Chadwick, Hannah Sheahan, Michael Tessler, Lucy Campbell-Gillingham, Jan Balaguer, Nat McAleese, Amelia Glaese, John Aslanides, Matt Botvinick, et al. Fine-tuning language models to find agreement among humans with diverse preferences. *Advances in Neural Information Processing Systems*, 35:38176–38189, 2022.

[7] 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.

[8] Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, et al. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. *arXiv preprint arXiv:2312.09390*, 2023.

[9] Souradip Chakraborty, Jiahao Qiu, Hui Yuan, Alec Koppel, Furong Huang, Dinesh Manocha, Amrit Singh Bedi, and Mengdi Wang. Maxmin-rlhf: Towards equitable alignment of large language models with diverse human preferences. *arXiv preprint arXiv:2402.08925*, 2024.

[10] Alex J Chan, Hao Sun, Samuel Holt, and Mihaela van der Schaar. Dense reward for free in reinforcement learning from human feedback. *arXiv preprint arXiv:2402.00782*, 2024.

[11] Jonathan D Chang, Wenhao Shan, Owen Oertell, Kianté Brantley, Dipendra Misra, Jason D Lee, and Wen Sun. Dataset reset policy optimization for rlhf. *arXiv preprint arXiv:2404.08495*, 2024.

[12] Yuyan Chen, Yichen Yuan, Panjun Liu, Dayiheng Liu, Qinghao Guan, Mengfei Guo, Haiming Peng, Bang Liu, Zhixu Li, and Yanghua Xiao. Talk funny! a large-scale humor response dataset with chain-of-humor interpretation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17826–17834, 2024.

[13] Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models. *arXiv preprint arXiv:2401.01335*, 2024.

[14] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

[15] Victor De Marez, Thomas Winters, and Ayla Rigouts Terryn. Thinc: A theory-driven framework for computational humor detection. *arXiv preprint arXiv:2409.01232*, 2024.

 https://doi.org/10.52202/079017-3978

[16] Vincent Dumoulin, Daniel D Johnson, Pablo Samuel Castro, Hugo Larochelle, and Yann Dauphin. A density estimation perspective on learning from pairwise human preferences. *arXiv preprint arXiv:2311.14115*, 2023.

[17] Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. Understanding dataset difficulty with $\mathcal{V}$-usable information. In *International Conference on Machine Learning*, pages 5988–6008. PMLR, 2022.

[18] Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

[19] Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, pages 10835–10866. PMLR, 2023.

[20] Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, et al. Reinforced self-training (rest) for language modeling. *arXiv preprint arXiv:2308.08998*, 2023.

[21] Joey Hejna, Rafael Rafailov, Harshit Sikchi, Chelsea Finn, Scott Niekum, W Bradley Knox, and Dorsa Sadigh. Contrastive prefence learning: Learning from human feedback without rl. *arXiv preprint arXiv:2310.13639*, 2023.

[22] Jack Hessel, Ana Marasović, Jena D Hwang, Lillian Lee, Jeff Da, Rowan Zellers, Robert Mankoff, and Yejin Choi. Do androids laugh at electric sheep? humor" understanding" benchmarks from the new yorker caption contest. *arXiv preprint arXiv:2209.06293*, 2022.

[23] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[24] Shu Hu, Yiming Ying, Xin Wang, and Siwei Lyu. Sum of ranked range loss for supervised learning. *Journal of Machine Learning Research*, 23(112):1–44, 2022.

[25] Kevin G Jamieson, Lalit Jain, Chris Fernandez, Nicholas J Glattard, and Rob Nowak. Next: A system for real-world development, evaluation, and application of active learning. In *Advances in Neural Information Processing Systems*, pages 2656–2664, 2015.

[26] Sophie Jentzsch and Kristian Kersting. Chatgpt is fun, but it is not funny! humor is still challenging large language models. *arXiv preprint arXiv:2306.04563*, 2023.

[27] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

[28] Ben King, Rahul Jha, Dragomir Radev, and Robert Mankoff. Random walk factoid annotation for collective discourse. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 249–254, 2013.

[29] Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. Understanding the effects of rlhf on llm generalisation and diversity. *arXiv preprint arXiv:2310.06452*, 2023.

[30] Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Lu, Thomas Mesnard, Colton Bishop, Victor Carbune, and Abhinav Rastogi. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*, 2023.

[31] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*, 2015.

[32] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.

[33] Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J Liu, and Jialu Liu. Statistical rejection sampling improves preference optimization. *arXiv preprint arXiv:2309.06657*, 2023.

[34] Blake Mason, Lalit Jain, Ardhendu Tripathy, and Robert Nowak. Finding all ϵ-good arms in stochastic bandits. *Advances in Neural Information Processing Systems*, 33:20707–20718, 2020.

[35] Gabriel Mukobi, Peter Chatain, Su Fong, Robert Windesheim, Gitta Kutyniok, Kush Bhatia, and Silas Alberti. Superhf: Supervised iterative learning from human feedback. *arXiv preprint arXiv:2310.16763*, 2023.

[36] Rémi Munos, Michal Valko, Daniele Calandriello, Mohammad Gheshlaghi Azar, Mark Rowland, Zhaohan Daniel Guo, Yunhao Tang, Matthieu Geist, Thomas Mesnard, Andrea Michi, et al. Nash learning from human feedback. *arXiv preprint arXiv:2312.00886*, 2023.

[37] Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

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

[39] Stefan Palan and Christian Schitter. Prolific. ac—a subject pool for online experiments. *Journal of Behavioral and Experimental Finance*, 17:22–27, 2018.

[40] Dragomir Radev, Amanda Stent, Joel Tetreault, Aasish Pappu, Aikaterini Iliakopoulou, Agustin Chanfreau, Paloma de Juan, Jordi Vallmitjana, Alejandro Jaimes, Rahul Jha, et al. Humor in collective discourse: Unsupervised funniness detection in the new yorker cartoon caption contest. *arXiv preprint arXiv:1506.08126*, 2015.

[41] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.

[42] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[43] Corby Rosset, Ching-An Cheng, Arindam Mitra, Michael Santacroce, Ahmed Awadallah, and Tengyang Xie. Direct nash optimization: Teaching language models to self-improve with general preferences. *arXiv preprint arXiv:2404.03715*, 2024.

[44] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[45] Dafna Shahaf, Eric Horvitz, and Robert Mankoff. Inside jokes: Identifying humorous cartoon captions. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1065–1074, 2015.

[46] Archit Sharma, Sedrick Keh, Eric Mitchell, Chelsea Finn, Kushal Arora, and Thomas Kollar. A critical evaluation of ai feedback for aligning large language models. *arXiv preprint arXiv:2402.12366*, 2024.

[47] Scott Sievert, Daniel Ross, Lalit Jain, Kevin Jamieson, Robert Nowak, and Robert Mankoff. Next: A system to easily connect crowdsourcing and adaptive data collection. In *SciPy*, pages 113–119, 2017.

[48] Anand Siththaranjan, Cassidy Laidlaw, and Dylan Hadfield-Menell. Distributional preference learning: Understanding and accounting for hidden context in rlhf. *arXiv preprint arXiv:2312.08358*, 2023.

[49] Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. Preference ranking optimization for human alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18990–18998, 2024.

 https://doi.org/10.52202/079017-3978

[50] Katherine Stasaski and Marti A Hearst. Semantic diversity in dialogue with natural language inference. *arXiv preprint arXiv:2205.01497*, 2022.

[51] Gokul Swamy, Christoph Dann, Rahul Kidambi, Zhiwei Steven Wu, and Alekh Agarwal. A minimaximalist approach to reinforcement learning from human feedback. *arXiv preprint arXiv:2401.04056*, 2024.

[52] Ervin Tanczos, Robert Nowak, and Bob Mankoff. A kl-lucb bandit algorithm for large-scale crowdsourcing. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 5896–5905, 2017.

[53] Yunhao Tang, Zhaohan Daniel Guo, Zeyu Zheng, Daniele Calandriello, Rémi Munos, Mark Rowland, Pierre Harvey Richemond, Michal Valko, Bernardo Ávila Pires, and Bilal Piot. Generalized preference optimization: A unified approach to offline alignment. *arXiv preprint arXiv:2402.05749*, 2024.

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

[55] Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, and Nathan Lambert. TRL: Transformer Reinforcement Learning. URL `https://github.com/huggingface/trl`.

[56] Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A Smith, Mari Ostendorf, and Hannaneh Hajishirzi. Fine-grained human feedback gives better rewards for language model training. *Advances in Neural Information Processing Systems*, 36, 2024.

[57] Canwen Xu, Corby Rosset, Luciano Del Corro, Shweti Mahajan, Julian McAuley, Jennifer Neville, Ahmed Hassan Awadallah, and Nikhil Rao. Contrastive post-training large language models on data curriculum. *arXiv preprint arXiv:2310.02263*, 2023.

[58] Zhijun Xu, Siyu Yuan, Lingjie Chen, and Deqing Yang. " a good pun is its own reword": Can large language models understand puns? *arXiv preprint arXiv:2404.13599*, 2024.

[59] Fanny Yang, Aaditya Ramdas, Kevin G Jamieson, and Martin J Wainwright. A framework for multi-a (rmed)/b (andit) testing with online fdr control. *Advances in Neural Information Processing Systems*, 30, 2017.

[60] Kevin Yang, Dan Klein, Asli Celikyilmaz, Nanyun Peng, and Yuandong Tian. Rlcd: Reinforcement learning from contrast distillation for language model alignment. *arXiv preprint arXiv:2307.12950*, 2023.

[61] Yueqin Yin, Zhendong Wang, Yi Gu, Hai Huang, Weizhu Chen, and Mingyuan Zhou. Relative preference optimization: Enhancing llm alignment through contrasting responses across identical and diverse prompts. *arXiv preprint arXiv:2402.10958*, 2024.

[62] Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*, 2024.

[63] Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. Rrhf: Rank responses to align language models with human feedback without tears. *arXiv preprint arXiv:2304.05302*, 2023.

[64] Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. Slic-hf: Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425*, 2023.

[65] Shanshan Zhong, Zhongzhan Huang, Shanghua Gao, Wushao Wen, Liang Lin, Marinka Zitnik, and Pan Zhou. Let's think outside the box: Exploring leap-of-thought in large language models with creative humor generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13246–13257, 2024.

[66] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36, 2024.

[67] Banghua Zhu, Evan Frick, Tianhao Wu, Hanlin Zhu, and Jiantao Jiao. Starling-7b: Improving llm helpfulness & harmlessness with rlaif, November 2023.

[68] 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. *arXiv preprint arXiv:1909.08593*, 2019.

 https://doi.org/10.52202/079017-3978

A Links to Resources

Our dataset is available at `https://huggingface.co/datasets/yguooo/newyorker_caption_ranking` under Creative Commons Attribution Non Commercial 4.0. Our codebase is available at `https://github.com/yguooo/cartoon-caption-generation` under Apache 2.0.

B Language Model Prompts

B.1 Description Generation

We use GPT-4o to generate descriptions for each cartoon. In the dataset from Hessel et al. [22] each cartoon has a canny description, an uncanny description, a location, and a list of entity. Entity are words that is related to the cartoon. We used the five shot method to generate a set of descriptions. The five examples are randomly selected from the testing set, and we use the these same five example for every cartoon descriptions generation. An example of our prompt is shown below.

User: In this task, you will see a cartoon, then write two descriptions about the cartoon, one uncanny description and one canny description, then write the cartoon's location, and the entities of the cartoon. I am going to give you five examples first and you write the last sets of description.
User: <Insert Cartoon Image>
Assistant: The canny description is <insert canny description> and the uncanny description is <insert uncanny description>, and the cartoon's location is <insert location>, and the entities of the cartoon are <insert entities>
.....Repeat user/assistant for four more examples.....
User: <Insert Cartoon Image>. The set of description is

Table 7: Examples of Generated Cartoon Descriptions

Type of descriptions	GPT-4o	Human Written [22]
Canny description	`A knight in armor is riding a horse, holding a lance with a traffic light on top. A line of businessmen in suits follows behind him.`	`There are two men on a horse. They are wearing soldier outfits. Businessmen follow behind them.`
Uncanny Description	`It's unusual to see a medieval knight leading modern businessmen as if going into battle.`	`There are businessmen following a two guys on horses who are soldiers.`
Location	`an open field`	`a hilly path`
Entities	`Knight, Horse, Businessmen, Traffic light`	`Warrior, Horses in warfare, Businessperson`

B.2 Caption Evaluation

We evaluate various models that generate captions by comparing the generated captions against four groups of human contestant entries at different ranking levels, which include top10, #200-#209, #1000-#1009, and contestant median. As concluded based on Table 2, we use GPT4-Turbo as evaluator with descriptions from Hessel et al. [22] in Overall Comparison and GPT4o-vision as evaluator with raw cartoon images in Best Pick Comparison. For both group comparison methods, we utilize the 5-shot in-context prompting technique, as mentioned in Section 4.2.

An example of Overall Comparison is shown below.

System: You are a judge for the new yorker cartoon caption contest.
User: In this task, you will see two description for a cartoon. Then, you will see two captions that were written about the cartoon. Then you will choose which captions is funnier. I am going to give you five examples first and you answer the last example with either A or B.
User: For example, the descriptions for the images are <Insert Canny Description> and <Insert Uncanny Description>. The two captions are A: <Insert CaptionA> B: <Insert CaptionB>
Assistant: The caption that is funnier is <Insert Answer>
.....Repeat user/assistant for four more examples.....
User: The descriptions for the images are <Insert Canny Description> and <Insert Uncanny Description>. The two groups of captions are group A: <Insert Caption Group A> group B: <Insert Caption Group B>
User: Choose the group of captions that is funnier. Answer with only one letter A or B, and nothing else.

An example of Best Pick Comparison is shown below.

System: You are a judge for the new yorker cartoon caption contest. Your job is to find the funniest caption.
User: In this task, you will see a cartoon first and two captions that were written about it then. The task is to choose which caption is funnier. I am going to show you five cartoons, corresponding captions and their answers first. In the end, for the last cartoon, answer with only one letter A or B, and nothing else.
User: <Insert Cartoon Image>
User: For this example, the two captions are A: <Insert CaptionA> B: <Insert CaptionB>. The answer is
Assistant: <Insert Answer>
.....Repeat user/assistant for four more examples.....
User: <Insert Cartoon Image>
User: Find the funniest caption for each group. Then choose the funnier group based on these funniest captions. Think step by step but finish the last line of your answer with only one letter A or B, and nothing else. A: <Insert Caption Group A> or B: <Insert Caption Group B>

B.3 Caption Generation

We used GPT-3.5-turbo, Claude-3-opus, and GPT-4-o to generate captions for each cartoons. We first use the system role to prompt it to generate 10 captions. Then we provide the image descriptions and then the image itself. For GPT-3.5-turbo, we simply only provided the image descriptions. For GPT-4-o, we have two versions where in one we provide the image itself, and the other we only provided the image descriptions. For Claude, we always provide both image description and image itself.

System: I want you to act as a sophisticated reader of The New Yorker Magazine. You are competing in The New Yorker Cartoon Caption Contest. Your task is to generate funny captions for a cartoon. Here are some ideas for developing funny captions. First think about characteristics associated with the objects and people featured in the cartoon. Then consider what are the unusual or absurd elements in the cartoon. It might help to imagine conversations between the characters. Then think about funny and non-obvious connections that can be made between the objects and characters. Try to come up with funny captions that fit the cartoon, but are not too direct. It may be funnier if the person reading the caption has to think a little bit to get the joke. Next, I will describe a cartoon image and then you should generate 10 funny captions for the cartoon along with an explanation for each.
User: <Insert Cartoon Image>
User: The cartoon's description is: <insert canny description>.The uncanny description is: <insert uncanny description>. The location of the cartoon is:<insert location>. The entities of the cartoon are: <insert image entities>

https://doi.org/10.52202/079017-3978

C Additional Experiment Setups

C.1 Human Experiment Details

Each participant provided informed consent in compliance with our Institutional IRB and was compensated for their time. We paid participants $12 an hour and spent about $600 on data collection. The following instructions were used for the human experiments.

C.1.1 Human Pairwise with description generated by GPT4o-vision

In each trial of this task, you will see a description of a cartoon and two captions: the cartoon description is on the top, and the two caption choices are beneath the cartoon description. For each trial, please select the caption that is the funniest for the cartoon.

C.1.2 Human Pairwise with Cartoon Image

In each trial of this task, you will see one cartoon and two captions: the cartoon is on top, and the two caption choices are beneath the cartoon. For each trial, please select the caption that is the funniest for the cartoon.

C.1.3 Human Group (Overall) with description generated by GPT4o-vision

In each trial of this task, you will see a description of a cartoon and two groups of captions: the cartoon description is on the top, and the two grouped caption choices are beneath the cartoon description. For each trial, please select the group of captions that is the funniest for the cartoon.

C.1.4 Human Group (Overall) with Cartoon Image

In each trial of this task, you will see a cartoon and two groups of captions: the cartoon is on the top, and the two grouped caption choices are beneath the cartoon. For each trial, please select the group of captions that is the funniest for the cartoon.

C.1.5 Human Group (Best Pick) with description generated by GPT4o-vision

In each trial of this task, you will see a description of a cartoon and two groups of captions: the cartoon description is on the top, and the two grouped caption choices are beneath the cartoon description. For each trial, please select the group of captions that contains the funniest caption for the cartoon. First, pick the funniest caption in each group, and then compare between the two captions to pick the funniest group.

C.1.6 Human Group (Best Pick) with Cartoon Image

In each trial of this task, you will see a cartoon and two groups of captions: the cartoon is on the top, and the two grouped caption choices are beneath the cartoon. For each trial, please select the group of captions that contains the funniest caption for the cartoon. First, pick the funniest caption in each group, and then compare between the two captions to pick the funniest group.

C.1.7 Human top 10 vs Claude generated captions

In each trial of this task, you will see a cartoon and two groups of captions: the cartoon is on the top, and the two grouped caption choices are beneath the cartoon. For each trial, ¡strong¿please select the group of captions that is the funniest for the cartoon.

C.2 Recalibration of GPT Models for Ranking

For group comparisons without chain of thought, we observe a strong bias of GPT4 models choosing A over B. In other words, for some examples, the model always chooses option A even after we flip the two groups. Therefore, this suggests we need to calibrate the model predictions. We adopt a simple approach by readjusting the decision threshold. Let s_i^A, s_i^B denote the log probabilities of choosing A and B by the GPT4 model for two groups of human submitted captions x_i^A and

x_i^B respectively. We use a small validation set of m examples $\{x_i^A, x_i^B\}_{i=1}^m$ with sigmoid scores $\{s_i^A, s_i^B\}_{i=1}^m$ and ground truth preference by the crowd denoted as $\{y_i \in \{A, B\}\}_{i=1}^m$. The current decision rule takes the form of $\widehat{f}(x_i^A, x_i^B) = \begin{cases} A & \text{if } s_i^A - s_i^B > 0 \\ B & \text{otherwise} \end{cases}$.

We simply set a different threshold τ, which induces $\widehat{f}_\tau(x_i^A, x_i^B) = \begin{cases} A & \text{if } s_i^A - s_i^B > \tau \\ B & \text{otherwise} \end{cases}$. The threshold τ^* is chosen so that the accuracy over the validation set is maximized:

$$\tau^* = \arg\max_\tau \sum_{i=1}^m \mathbf{1}\{y_i = \widehat{f}_\tau(x_i^A, x_i^B)\}.$$

Ties are broken arbitrarily above. We then use the recalibrated decision rule with τ^* for all of our evaluations.

C.3 Finetuning Experiment Details

Our training and test split for finetuning range from contest 530 to 890. In particular, our dataset includes all the data of [22] with ranking information within this range. ([22] only contains contests up to #763.) Thus, we choose our test split to be the combination of testing (47 contests) and validation split (44 contests) of [22] within the 530-890 range. The rest available contests form our training split.

Our finetuning methods are trained from Mistral 7B Instruct v0.1 and LLaVa v1.6 Mistral (multimodal case) via LoRA updates [23]. We use a variant of Mistral 7b model as our initial reward model to finetune from [3]. The choice of reward is based on our benchmarking results of top reward models on our caption generation dataset (Table 8). For SFT methods, we train on 1000 pairs of captions from each contest, with the preferred caption from the top 1000 captions and the alternative randomly sampled from the rest. For reward modeling, DPO and RLHF, we train on 1000 pairs of captions with three standard deviations apart according to Equation (1) per contest. Additionally, we train our model using the default choice of optimizer from TRL up to 1 epoch. Then, we search for the best hyper-parameter over the neighborhood of default parameters and pick the best performing model under our GPT-based group comparison metrics. For our reward model, we pick the best model based on the reward evaluation on the holdout set. For both pretrained and finetuned models, we use the same generation configuration file with temperature 0.7, top-p sampling probability 0.95, repetition penalty 1.15. When evaluating using the Best-of-N (BoN) method, we pick the top 10 captions based on the trained reward model, out of 50 generated candidates from caption generation models. Our choice of batch is 64 for SFT and reward model, and 128 for all other settings.

During the training process of DPO, PPO, SFT, we create a separate padding tokens and resize the token embedding of the pretrained model so that the text generation can terminate properly. Furthermore, in the loss design of SFT case, we only evaluate the next-token prediction loss on the caption segment, as all the training texts contain similar prompts. Since we only reported the iteration with the best results, early stopping occurs before a single epoch for the choice of best iterations.

We also noted that PPO performs the best when starting from the pretrained Mistral Instruct 7B model, whereas DPO performs the best from a sft checkpoint of Mistral. This SFT checkpoint needs to be tuned on simple prompts and does not render a better performance than the sft tuned on the best prompt (with all those descriptions).

Choice of Prompts In Table 10, we document the best prompt we found for each training algorithm. Generally speaking, the zero-shot, SFT, preference learning algorithm each require simpler prompts than the one preceding them.

Computation Cost Finetuning a SFT, DPO, PPO model usually takes 2-4 days to train till convergence on a A100 machine. Evaluating a single number of each scenario cost roughly $5 on the openai platform.

[3]We use the pretrained reward model from `https://huggingface.co/weqweasdas/RM-Mistral-7B`

 https://doi.org/10.52202/079017-3978

D Crowdsourced Caption Contest Ratings

Algorithm 1 Upper Confidence Bound (UCB) Algorithm

1: **Initialization:** For each caption x, initialize $N_x(0) = 0$ and $\hat{\mu}_x = 0$.
2: **for** $t = 1$ to T **do**
3: Select caption $x_t = \arg\max_x \left(\hat{\mu}_x + \sqrt{\frac{2\ln(4N_x(t)^2)}{N_x(t)}}\right)$.
4: Observe the reward $r_t \in \{1, 2, 3\}$ for caption x_t.
5: Update the number of times action x_t has been selected: $N_{x_t}(t) = N_{x_t}(t-1) + 1$.
6: Update the empirical mean reward of action x_t:

$$\hat{\mu}_{x_t} = \frac{N_{x_t}(t-1) \cdot \hat{\mu}_{x_t} + r_t}{N_{x_t}(t)}$$

7: **end for**

As described in the text, we used a UCB [2] variant to encourage high-performing captions to recieve the votes. We experimented with standard UCB (see Algorithm 1) and KL-UCB specifically optimized for discrete rewards [52]. The data repository labels datasets according to which algorithm was employed for each contest. In practice, using UCB in high-traffic asynchronous environments faces specific challenges. For example, we wanted to ensure that voters could only vote on one caption at a time, that the model sent batches of captions to users to reduce round trips to the server, and that the underlying model was able to update as frequently as possible. For more details on overcoming such challenges, see [25].

E Additional Results

We benchmark the performance of different reward model as in Table 8. It is worth noting that our goal here is to understand the effect of the ranking model for the downstream preference learning algorithms, thus we evaluate on the same dataset as in Equation (1) instead of the setting of Table 2. weqweasdas/RM-Mistral-7B and Eurus-RM-7B Instruct are the top two models with the highest reward ranking accuracy. We choose to use weqweasdas/RM-Mistral-7B because it generally achieves better ranking accuracy for various data settings that we experimented on.

In our experiment, we noticed that PPO algorithm requires a much more aggressive early stopping scheme than DPO and SFT. Thus, we further look at the training dynamics of the PPO algorithm in Table 9. Here, the batch size is 128. It is worth noting that the result at iteration 0 has an lower overall win rate than the zero shot result in Table 3. The reason is that our PPO and DPO algorithms need to use a simpler prompt as in Table 10 to generate meaningful texts. From Table 9, we verified the steady increase of the mean reward and decrease of the training loss. However, the improvement on these metrics does not corresponds to an improvement of the overall humorous generation. We hypothesize that this is due to the complex nature of humor and the potential for out-of-distribution generations when running RLHF.

Table 8: Reward model benchmark

	Reward Ranking Acc (%)
Mistral-7B Instruct	73.17
Llama-3-8B Instruct	74.01
Llama-2-7B Chat	72.63
weqweasdas/RM-Mistral-7B	**74.05**
Eurus-RM-7B	**74.18**
FsfairX-LLaMA3-RM-v0.1	73.72
Qwen1.5-7B-Chat	72.26

Table 9: Training Dynamics of PPO

Iteration	0	10	20	30	40	50
Contestant Median (Overall Win Rate (%)) ↑	17.03	**24.73**	16.48	9.89	6.04	4.95
Mean Reward ↑	0.0057	0.0260	0.0186	0.1309	0.1356	**0.2587**
Loss ↓	0.3592	0.2001	0.1773	0.1709	0.0848	**0.0584**

Table 10: Best choice of prompts for each training algorithm

	Best Choice of Prompt
Zero-Shot	[INST] <> I want you to act as a sophisticated reader of The New Yorker Magazine. You are competing in The New Yorker Cartoon Caption Contest. Your task is to generate funny captions for a cartoon. Here are some ideas for developing funny captions. First think about characteristics associated with the objects and people featured in the cartoon. Then consider what are the unusual or absurd elements in the cartoon. It might help to imagine conversations between the characters. Then think about funny and non-obvious connections that can be made between the objects and characters. Try to come up with funny captions that fit the cartoon, but are not too direct. It may be funnier if the person reading the caption has to think a little bit to get the joke. Next, I will describe a cartoon image and then you should generate 1 funny caption for the cartoon along with an explanation for each. scene: <*scene*> description: <*description*> uncanny description: <*uncanny description*> entities: <*entities*> <> funny caption: [/INST] <*sample caption*>
SFT	[INST]I want you to act as a sophisticated reader of The New Yorker Magazine. You are competing in The New Yorker Cartoon Caption Contest. Your task is to generate funny captions for a cartoon. Here are some ideas for developing funny captions. First think about characteristics associated with the objects and people featured in the cartoon. Then consider what are the unusual or absurd elements in the cartoon. It might help to imagine conversations between the characters. Then think about funny and non-obvious connections that can be made between the objects and characters. Try to come up with funny captions that fit the cartoon, but are not too direct. It may be funnier if the person reading the caption has to think a little bit to get the joke. Next, I will describe a cartoon image and then you should generate 1 funny caption for the cartoon[/INST] scene: <*scene*> description: <*description*> uncanny description: <*uncanny description*> entities: <*entities*> funny caption: <*sample caption*>

https://doi.org/10.52202/079017-3978

	Best Choice of Prompt
LLaVA	[INST] I want you to act as a sophisticated reader of The New Yorker Magazine. You are competing in The New Yorker Cartoon Caption Contest. Your task is to generate funny captions for a cartoon. Here are some ideas for developing funny captions. First think about characteristics associated with the objects and people featured in the cartoon. Then consider what are the unusual or absurd elements in the cartoon. It might help to imagine conversations between the characters. Then think about funny and non-obvious connections that can be made between the objects and characters. Try to come up with funny captions that fit the cartoon, but are not too direct. It may be funnier if the person reading the caption has to think a little bit to get the joke. Next, I will provide a cartoon image with descriptions and then you should generate 1 funny caption for the cartoon along with an explanation for each. image: <*image*> scene: <*scene*> description: <*description*> uncanny description: <*uncanny description*> entities: <*entities*> Generate a funny caption for the image: [/INST] <*sample caption*>
DPO/PPO/Reward Model	scene: <*scene*> description: <*description*> uncanny description: <*uncanny description*> entities: <*entities*> funny caption: <*sample caption*>

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Section 1.
   (b) Did you describe the limitations of your work? [Yes] See Section 6.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section 6.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] Our data collection and experiment procedures conform to the ethics review guidelines.
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments (e.g. for benchmarks)...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See Section 3 for details on dataset and see Section 5 for details on the main experimental results. The dataset is available at `https://huggingface.co/datasets/yguooo/newyorker_caption_ranking` and the codebase is available at `https://github.com/yguooo/cartoon-caption-generation`.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] The training details is available in Appendix C.3.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Table 1 and Table 2.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix C.3.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] We cited the relevant assets in Section 2.
   (b) Did you mention the license of the assets? [Yes] See Appendix C. Our code is under Apache 2.0 and our dataset is under Creative Commons Attribution Non Commercial 4.0.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We include our released datasets and codebase as url. See Section 3 and Section 5.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] See Section 3.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] See Section 3.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [Yes] See Appendix C.1
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [Yes] See Appendix C.1
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [Yes] See Appendix C.1

https://doi.org/10.52202/079017-3978