ReaL Stories: RL for Adaptive AI Storytelling

Stanford CS224N Custom Project

Ayaan Chand
Department of Computer Science
Stanford University
ayaan26@stanford.edu

Aniket Mahajan
Department of Computer Science
Stanford University
aniketm@stanford.edu

Aditya Sood
Department of Computer Science
Stanford University
adibsood@stanford.edu

Abstract

Advancements in NLP, particularly in transformer architectures, have revolutionized the capabilities of language models. However, these models struggle with generating creative, cohesive, and engaging stories. This study seeks to enhance storytelling capabilities in four state-of-the-art language models—GPT-2, Meta Llama 3 8B, Mistral 7B, and SOLAR 10.7B—by fine-tuning them on a dataset of human-written stories and prompts, and optimizing them using Proximal Policy Optimization (PPO) reinforcement learning via a web-based feedback interface.

Our evaluation metrics included creativity, coherence, human likeness, and engagement, assessed through human reviewers and GPT-4. Results showed significant improvements across all models, with Meta-Llama-3-8B (PPO-fine-tuned) performing the best. These findings highlight the potential of reinforcement learning to enhance LLM storytelling. In the future, we can incorporate broader datasets to include diverse perspectives and employ more efficient fine-tuning techniques, while maintaining robust content moderation to ensure generated stories are appropriate for all ages.

1 Key Information to include

- Mentor: Zhoujie Ding
- External Collaborators (if you have any): N/A
- Sharing project: No
- Team contributions: Aditya created fine-tuning scripts for training GPT-2 on GCP and developed the back-end and part of the front-end for the web application. Aniket implemented PPO fine-tuning for Llama and created our GPT4 evaluation script. Ayaan adapted scripts to fine-tune our other models on TogetherAI and helped build the front-end with Aditya. All team members collaborated equally to get user testing data, debug different parts of each other's code, and write the final report.

2 Introduction

Due to nascent developments in Natural Language Processing (NLP), transformer-based large language models (LLMs) have become increasingly adept at generating human-like text to perform tasks like neural machine translation. However, these models fall short in their ability to generate cohesive and engaging stories. Creative storytelling demands not only syntactic precision but also a

subtle understanding of narrative structures, character development, and original creativity—elements that current models struggle to emulate.

Enhancing the story-telling capabilities of language models serves practical purposes for various use cases such as entertainment, education, and therapeutic applications. Stories have historically been essential to human culture, serving as a medium of communication and a powerful tool for conveying complex ideas and emotions.

In this study, we focused on four prominent SOTA language models: GPT-2, Meta-Llama-3-8B, Mistral-7B, and SOLAR-10.7B. Each of these models embodies a different architecture and approach to language modeling, providing a diverse set of capabilities and limitations. Our objective is to evaluate and enhance these models' story-telling performance through a systematic approach involving fine-tuning and Proximal Policy Optimization (PPO) reinforcement learning (RL).

We utilized a robust database of human-written stories and prompts to fine-tune each model. This process aims to imbue the models with a deeper understanding of narrative elements and improve their ability to generate coherent and engaging texts. Using a web interface to implement PPO RL, we tailored the model to human feedback for adaptive story-telling. This interface acts as a mechanism to collect data for reinforcement learning by tokenizing user feedback into a reward signal for the LLM. In doing so, we are able to incorporate human preferences into the training process, aligning the models' outputs more closely with human expectations and improving their overall narrative quality.

3 Related Work

Emerging attempts at NLP-based story generation included architectures like RNNs, GRUs, and LSTMs. However, the development of the Transformer architecture led to the birth of modern SOTA language models that exhibit stronger performance in NLP tasks Vaswani et al. (2017). Subsequent studies demonstrated the effectiveness of fine-tuning GPT-2 on various datasets to enhance its performance Radford et al. (2019). Ever since, experiments have shown the profound impact of fine-tuning LLMs on datasets to improve their ability to solve general NLP tasks Howard and Ruder (2018). While central NLP tasks like language translation have been broadly researched, the realm of story-telling remains less explored.

Within RL, PPO rose to prominence for its simplicity and stability when incorporating human preferences Schulman et al. (2017). In our study, we collect user feedback through a web interface and tokenize it into a reward signal for PPO, allowing the models to learn from human preferences and improve their narrative quality.

Evaluating the quality of generated text is challenging and often involves both quantitative and qualitative measures. Metrics such as the BLEU score were proposed, however, they don't properly capture the open-ended and creative nature of storytelling and are insufficient for our task Liu et al. (2016). To address this, we employ a combination of automatic evaluation using a GPT-4 evaluation script and human evaluation.

Comparative analytics of LLMs have been a cornerstone of model evaluation by providing insights into the strengths and weaknesses of different LLMs. A study compared the performance of GPT-3 with other language models, highlighting the importance of model architecture Brown et al. (2020). Our study took inspiration from this and compared four SOTA models and their fine-tuned counterparts in their ability to excel at this task.

Ultimately, our work builds on existing research in transformer-based models, fine-tuning, and human feedback integration via reinforcement learning. Through our comparative analysis of four SOTA language models and their fine-tuned versions, we aim to advance the field of AI-driven storytelling and enhance the narrative capabilities of language models.

4 Approach

4.1 Model Architectures

Throughout this project, we utilized four transformer-based model architectures: Meta-Llama-3-8B, GPT-2, Mistral-7B, and SOLAR-10.7B. We compared their performance before and after fine-tuning each on a common dataset.

4.1.1 Meta-Llama-3-8B

Meta-Llama-3-8B is a SOTA decoder-only language model that uses multi-head self-attention and a collection of 8 billion parameters. The model's emphasis on self-attention and accessing contextual information makes it particularly well-fitted for open-ended text generation.

4.1.2 GPT-2

GPT-2 features 1.5 billion parameters and a decoder, unidirectional model. This means it reads left-to-right and only references past generated text and tokens. As such, this is fitted for text generation and worth adding to our ensemble of models to test. While GPT-3.5 is a larger version of GPT-2 with nearly 100 times the number of parameters, given the associated increased compute requirements, we did not deem it initially worth experimenting.

4.1.3 Mistral-7B

Mistral-7B uses an extensive collection of 7 billion parameters. It utilizes a sparse attention mechanism to reduce computational resources for training without compromising on performance. Unlike other models we tested, Mistral-7B employs a hybrid approach, utilizing both transformer and RNN architectures. Additionally, it pioneers self-attention techniques like group query and sliding window attention.

4.1.4 SOLAR-10.7B

SOLAR-10.7B is a transformer-based language model with multi-head self-attention and 10.7 billion parameters, as well as the only encoder-only model we tested. The authors also use this model to propose depth up-scaling, a method of upscaling LLMs without significant increases in compute or resources.

4.2 Reinforcement Learning: Proximal Policy Optimization

To further fine-tune the model to fulfill the criteria of this task, we turned to reinforcement learning with human feedback. Our model hyperparameters generate two chat completions for a given prompt from which the user selects the response they prefer. The selected output is tokenized into a reward signal and the model is accordingly updated.

We chose PPO as our reinforcement learning technique since it updates on user feedback but does not over-adapt on any given policy due to in-built objective function restraints. The loss function is given as:

$$L(\theta) = \mathbb{E}[\min(r_t(\theta)A_t, \text{clip}(r_T(\theta), 1-\epsilon, 1+\epsilon)A_t)]$$

where r_t is the ratio from the new to old policy, A_t is the improved advantage function, and ϵ is the clipping constant Schulman et al. (2017). By taking the minimum of these two values, we prevent any update to the model exceeding some ϵ of change. We defined a simple binary reward function rooted in the user's selection for some response y_t as,

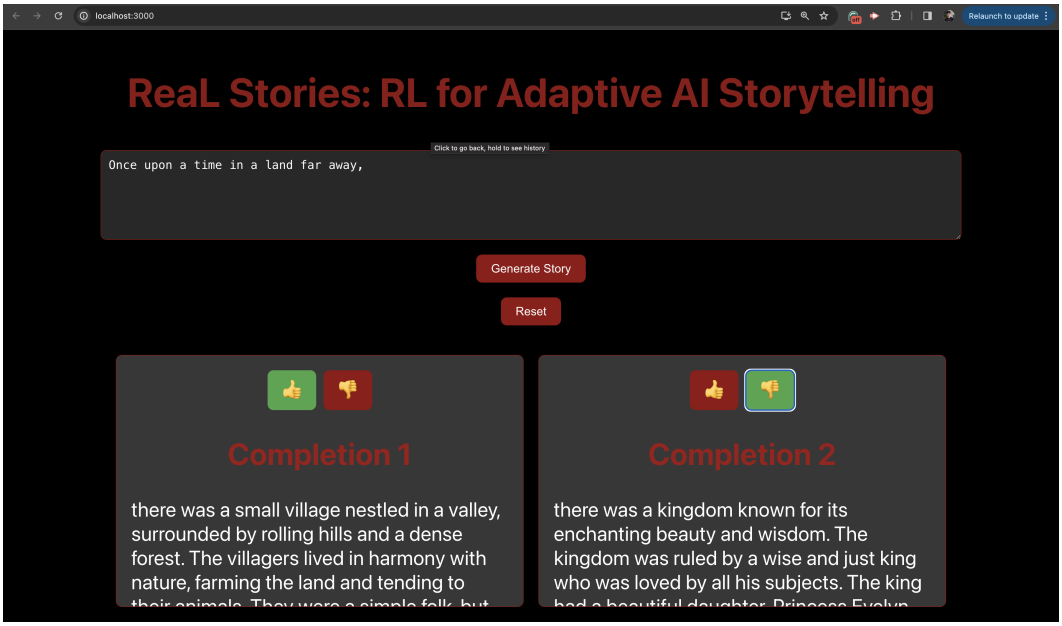
$$R(y_t) = \begin{cases} 1 & \text{if } y_t = y_s \\ 0 & \text{if } y_t \neq y_s \end{cases} \tag{1}$$

where y_s is the selected response from a set of generated responses $\mathbf{y}$. In other words, the selected response is given a score of 1 and all other responses are given a score of 0.

4.3 Web Interface

To enhance user experience for our human testers and demonstrate the real-world applicability of our project, we built a web application around our LLM. The front end, made with ReactJS, enabled users to enter a prompt of their choosing. They then pressed the "generate story" button which triggered a Flask route to our LLM. Using TogetherAI, the language model was deployed to an instance of type

RTX-6000-48GB. Two queries were generated by our model and returned to be displayed to the user, who then chose their favorite. As described in the Reinforcement Learning section of this paper, this information further fine-tuned the model using a reward-based mechanism.



5 Experiments

5.1 Data

We fine-tuned the models using the WritingPrompts dataset from Reddit's r/WritingPrompts, compiled by the Facebook AI Research Team. This dataset consists of prompts and their corresponding short story completions. The goal is to teach our models how to generate short stories based on user-provided prompts wri (2018).

We first preprocessed the data, formatting the raw JSONL file into intuitive writing prompt-response pairings. We also filtered out non-UTF8 symbols from the text as this stumped our model during fine-tuning on prior iterations of experimenting. Lastly, we manually removed violent and inappropriate stories from the dataset, as we intended for this tool to be usable for all ages, including children.

5.2 Evaluation method

To evaluate a model's performance for this task, an otherwise subjective and qualitative measure, we defined a set of four metrics a response must be scored on: (1) creativity, (2) coherence, (3) human-like writing, and (4) engagement. To elicit this scoring we used a combination of human and computer scoring. Having 50 peers manually score generated responses from each model, we were able to aggregate a rough scoring of each model, and to draw a larger set of scores, we turned to GPT-4 prompting.

Using OpenAI API, we queried GPT-4 to generate scores for each model's generated responses to a given prompt on the same metrics our peers scored on. For each model output, we prompted GPT-4:

> *"Please rate the following story on a scale from 0 to 1 with 0.01 increments based on four metrics: creativity, coherence, human-like writing, and engagement. The*

story [example short story from WritingPrompts dataset] *receives a score of 0.50 in each metric."*

We then take the mean across the samples for each category as a metric of the average performance of the model.

5.3 Experimental details

5.3.1 Training Platform

Our choice to use the 8B parameter Llama-3 model as opposed to the 70B model was due to our compute resource limitations on TogetherAI and GCP. Furthermore, considering the laws of scale in model performance, it made sense to equalize the models such that they all had a similar amount of parameters.

Since the GPT series is not currently available for fine-tuning on TogetherAI, we developed a fine-tuning script for GPT-2 and ran it on an NVIDIA VM (with CUDA enabled for GPU training) in Google Cloud. Since the next biggest model (GPT3) had an estimated 175B parameters–which is not comparable to the other models we're evaluating–we used GPT-2 (1.5B parameters).

5.3.2 Hyperparameters

We used a similar suite of hyperparameters across all models. A temperature of 0.9 was optimal for ensuring chat completions maintained originality without being too similar (for the PPO stage). On GCP, we made design decisions to strike a balance between model quality and efficiency. We leveraged batch sizes of 16 and FP16 precision to reduce memory consumption and streamline training. Rather than starting at our learning rate, a warmup period of 500 steps was allocated for stability to build up our learning rate from zero to $5e^{-5}$.

5.4 Results

Table 1: Mean Model Performance on GPT-4 Scored Metrics (Scored out of 1)

Model	Creativity	Coherence	Human-Like	Engagement	Average
GPT-2	0.32	0.27	0.47	0.21	**0.32**
Mistral-7B	0.47	0.34	0.54	0.43	**0.45**
SOLAR-10.7B	0.53	0.51	0.63	0.46	**0.53**
Llama-3-8B	0.67	0.47	0.58	0.47	**0.55**
GPT-2 (Finetune)	0.41	0.25	0.56	0.24	**0.37**
Mistral-7B (Finetune)	0.43	0.33	0.58	0.36	**0.43**
SOLAR-10.7B (Finetune)	0.72	0.48	0.53	0.42	**0.54**
Llama-3-8B (Finetune)	0.66	0.50	0.68	0.46	**0.58**
PPO on Llama-3-8B	0.69	0.52	0.70	0.43	**0.59**

The following data was collected from our GPT-4 evaluation script that tested 200 output samples from each model. In general, fine-tuning the models on the WritingPrompts datasets proved to increase the performance of the model. The only exception was on Mistral-7B, which showed a slight decrease from 0.45 before fine-tuning to 0.43 after fine-tuning.

Table 2: Mean Model Performance on Human-Scored Metrics (Scored out of 1)

Model	Creativity	Coherence	Human-like	Engagement	Average
GPT-2	0.35	0.29	0.50	0.25	**0.35**
Mistral-7B	0.50	0.36	0.57	0.46	**0.47**
SOLAR-10.7B	0.55	0.54	0.65	0.49	**0.56**
Llama-3-8B	0.70	0.49	0.61	0.50	**0.58**
GPT-2 (Fine-tune)	0.45	0.28	0.59	0.29	**0.40**
Mistral-7B (Fine-tune)	0.48	0.42	0.61	0.43	**0.49**
SOLAR-10.7B (Fine-tune)	0.75	0.50	0.56	0.45	**0.57**
Llama-3-8B (Fine-tune)	0.70	0.52	0.71	0.49	**0.61**
PPO on Llama-3-8B	0.73	0.55	0.73	0.67	**0.67**

The results from the 50 human evaluators seem to mirror those from the GPT-4 evaluation script, boosting our confidence in the validity of our data. Since Llama-3-8B (fine-tuned) was consistently the highest performer across all metrics for GPT-4 and human evaluation, we chose to run PPO on it. Given our cost limitations, it was only possible for us to run PPO on one model. While running PPO on the fine-tuned Llama-3 model led to marginal improvements in metrics like creativity, human-like sound, and coherence, it did not improve the engagement metric.

Figure 1: Distribution of Mean Scores for Llama-3-8B series over 200 GPT-4 Scored Responses

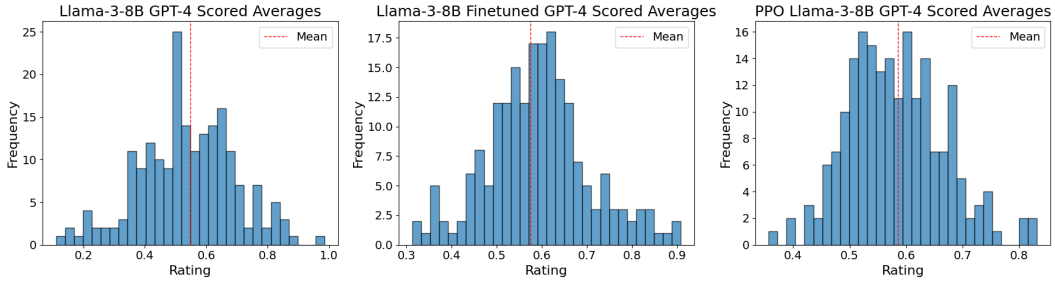
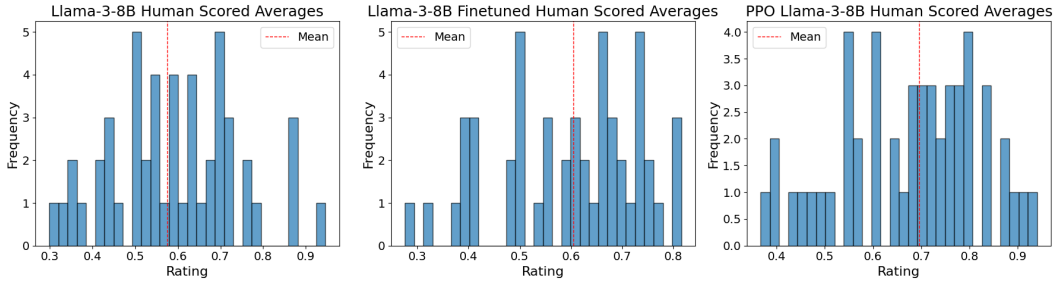


Figure 2: Distribution of Mean Scores for Llama-3-8B series over 50 Human-Scored Responses



Figures 1 and 2 depict the distributions of average scores from GPT-4 and human evaluators, respectively. We found that the scores from GPT-4 followed a Gaussian distribution where there was significant clustering around the mean. The lack of spread and the presence of clustering validate the accuracy of these test metrics and show that the mean score increased as models were fine-tuned (more so after PPO). Figure 2 did not have this type of distribution, likely due in part to variability from human evaluators and the small sample size of 50. However, the results from here seemed to largely mirror the trend seen in Figure 1, where fine-tuning the model led to greater mean scores. This trend held where PPO was concerned.

6 Analysis

For our qualitative analysis, we chose one example of sample prompt and completion from our fine-tuned models. We have included the prompt and, for brevity, the first couple of sentences from each model. The second example demonstrates the differences in sentence quality between Llama-3, Llama-3 fine-tuned, and Llama-3 PPO.

6.1 Exhibit 1: Fine-tuned Model Comparisons

> **Prompt:** *"Once upon a time, there was a kingdom where magic was real."*
>
> **GPT-2:** *"Once upon a time, there was a kingdom. Magic was real and the people were happy."*
>
> **Mistral-7B:** *"Once upon a time, in a faraway kingdom, magic was real. The people lived happily and used magic in their daily lives."*
>
> **SOLAR-10.7B:** *"Once upon a time, a kingdom where magic was real flourished with joy and wonder. Every creature and person had a touch of magic in them, making life extraordinary."*
>
> **Llama-3-8B:** *"Once upon a time, in a distant kingdom where magic was real, the air shimmered with enchantment. The kingdom flourished under the gentle rule of a wise king, and the people thrived in peace and harmony."*

In the excerpts above, it can be observed that GPT-2 provided a very basic continuation of the prompt with minimal creativity. Mistral-7B and SOLAR-10.7B added significant complexity to the chat completion by using more colorful adjectives and settings. However, Llama-3-8B made the most significant improvement by creating a rich narrative that sounds as though it was taken from a children's storybook.

6.2 Exhibit 2: Llama-3-8B Series Comparisons

> **Prompt:** *"Once upon a time, a knight set out on a quest to find a lost treasure."*
>
> **Llama-3-8B:** *"Once upon a time, a fearless knight embarked on a quest for a lost treasure. Through dark forests and over high mountains, he faced many dangers on his journey."*
>
> **Llama-3-8B fine-tuned:** *"Once upon a time, a knight known for his bravery set off to find a hidden treasure. His quest led him through enchanted forests and ancient ruins, facing numerous challenges along the way."*
>
> **Llama-3-8B PPO:** *"Once upon a time, a brave knight began his quest to discover a long-lost treasure. With determination, he faced many trials and ventured deep into forgotten lands, where ancient spells and mythical creatures awaited."*

In exhibit 2, we compare the storytelling capabilities of the Llama-3-8B series side-by-side. We can observe that with greater fine-tuning, the models are able to delve into deeper levels of detail. Standard Llama-3-8B describes "many dangers" like "dark forests" and "high mountains", but fine-tuned Llama-3-8B uses more illustrative terms like "enchanted forests" and "ancient ruins". The PPO is able to learn the user's preference for writing in classic fantasy literature format.

Ultimately, our qualitative analysis demonstrates that all models are able to generate coherent text given an input. However, certain nuances of each model are also present. GPT-2 had a clear tendency to stay very close to the prompt, never deviating too far from it. Mistral-7B and SOLAR-10.7B were a big step up from GPT-2 in that they added imagery and vibrant details. The Llama-3-8B model significantly enhanced the narrative quality by using more creative and engaging writing. Fine-tuning further refined this model, and the Llama-3-8B PPO model excelled in learning user preferences to produce text expected of classical literature.

7 Conclusion

In our project, we used the WritingPrompts dataset to demonstrate the efficacy of fine-tuning four transformer-based language models—Meta-Llama-3-8B, GPT-2, Mistral-7B, and SOLAR-10.7B—to

enhance their storytelling capabilities. All four models showed notable improvements in generating coherent and engaging narratives, with the Meta-Llama-3-8B (PPO-fine-tuned) model exhibiting the best overall performance according to our evaluation metrics. These promising results highlight the effectiveness of using Proximal Policy Optimization (PPO) reinforcement learning to improve the storytelling capabilities of language models.

Despite achieving positive results, our project has several limitations. It is difficult to ensure that the generated stories are appropriate for all audiences, particularly in filtering out biased or inappropriate narratives. Another limit we faced was the substantial computational resources required for fine-tuning these large models, which constrained the frequency and extent of our updates. To fine-tune one model on the full dataset would require approximately 300 USD, so we had to downsample the dataset to 500 user-written prompts and short story completions. Since only a small portion of the dataset was used, selection biases may have influenced the fine-tuning and evaluation processes.

In the future, depending on the availability of computational resources, we could expand and diversify our dataset to include varied styles, genres, and cultural perspectives to enrich the model's ability to generate diverse and creative content. To this end, we could implement Parameter Efficient Fine-Tuning (PEFT) techniques like Low-Rank Adaptation of LLMs (LoRA) to reduce the computational resources and runtime needed for fine-tuning. Transfer learning and content moderation strategies such as automated filtering systems could also be promising avenues for future research.

8 Ethics Statement

One significant ethical challenge our project poses is the potential for bias and unfairness in the generated stories. Language models can perpetuate stereotypes and biases present in the training data, resulting in generated content that may be harmful to certain groups of people. This may reinforce harmful stereotypes and spread misinformation, which could have detrimental effects on society. For example, if our training data contained instances of racial discrimination or gender bias, the model might identify these to be characteristics of a story and generate stories with these components.

Another cause for concern is the enhanced ability of these models to produce highly persuasive content that is able to sway public opinion. These narratives have the ability to manipulate users into adopting specific viewpoints and making certain decisions, which is concerning through a politic lens or in the case of biased information dissemination.

Additionally, story generation can produce content that is inappropriate for certain audiences. The stories produced might include inappropriate, offensive, or harmful content, which can negatively impact users, particularly vulnerable populations such as children or individuals with mental health conditions. It is important to implement safeguards and filters regarding what our model can and cannot produce.

To address the first ethical challenge above, we propose bias mitigation. We could achieve this by implementing bias detection and mitigation techniques, such as using diverse and balanced training datasets that reduce the risk of generating biased content. It is imperative that our models are regularly audited and updated to ensure that emerging biases are curbed.

Second, through promoting transparency, users can have the autonomy to choose which content they interact with. For example, stories generated by AI could be labeled as such, and users could choose not to interact with such content. This could prevent the possibility of user manipulation given that users know the kind of content they are interacting with.

Finally, robust content moderation is necessary to ensure that the generated stories are appropriate and safe. We could develop content moderation systems that filter out inappropriate or harmful content before it reaches users. This could involve implementing safety checks and utilizing human reviewers to oversee the content quality and appropriateness of generated stories, thereby protecting vulnerable populations.

References

2018. Writing prompts dataset. Data provided by Facebook AI Research Team.

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.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Chia-Wei Liu, Ryan Lowe, Iulian V Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv preprint arXiv:1603.08023*.

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

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.