Investigating Direct Preference Optimization for GPT-2 Downstream Task Performance

Stanford CS224N Default Project

Onyinyechi Okoye
Department of Computer Science
Stanford University
onyie@stanford.edu

Natalie Shell
Department of Computer Science
Stanford University
natalie1@stanford.edu

Abstract

Our project investigates the results of fine-tuning the GPT-2 model for sentiment analysis, paraphrase detection, and sonnet generation. Transformer-based models, such as GPT-2, are highly capable, yet aligning them with human language preferences poses a challenge. Our work addresses this challenge by first building an implementation of GPT-2. Then, we fine-tuned the model, and evaluated its performance on sentiment analysis using the Stanford Sentiment Treebank and CFIMDB datasets, paraphrase detection with Quora Question Pairs database, and sonnet generation using sonnets written by William Shakespeare. To improve our model's general performance while moving it to align with human judgments and preferences, we integrated Direct Preference Optimization (DPO), an alternative method to reinforcement learning. DPO directly optimized the language model to align with human preferences without explicitly training a separate reward model, simplifying the process of policy fine-tuning. In our experiments, DPO yielded improved performance in paraphrase detection and sonnet generation tasks, compared to our implementations that used the standard maximum likelihood estimation. Furthermore, we forced that the DPO implementations enhanced the quality of our GPT-2 outputs and better aligned them to human preferences.

1 Introduction

Transformer-based language models such GPT-2 have invoked a revolution in natural language processing as they have demonstrated strong capabilities in language comprehension and generation tasks. However, a remaining challenge lies in effectively aligning these models' outputs with human preferences. Traditional approaches, such as reinforcement learning from human feedback, involve a two-step process that requires first training a separate reward model and then subsequently fine-tuning the language model through reinforcement learning, which risks introducing instability and significant computational overhead.

In this project, we implement and fine-tune GPT-2, exploring its performance in three downstream tasks: sentiment analysis, paraphrase detection, and sonnet generation. Sentiment analysis involves classifying the positivity and negativity of texts, paraphrase detection entails identifying semantically equivalent pairs of sentences to determine if one is a paraphrase of the other, and sonnet generation evaluates the model's ability to generate Shakespearean sonnets given the first few lines. To address the limitations associated with RLHF, we leverage Direct Preference Optimization (DPO), a more recently introduced alternative method that directly optimizes our language model to align with human preferences using a simplified and stable classification-based objective (Rafailov et al., 2023).

Our work demonstrates the advantages of DPO compared to conventional fine-tuning and RLHF approaches. Experimental results on datasets such as the Stanford Sentiment Treebank, Quora Question Pairs, and sonnet generation tasks show that DPO enhances our GPT-2 performance, aligning the model outputs more closely with human judgment. These findings substantiate DPO's

potential as a practical, computationally efficient method for fine-tuning language models, and its suitability for broader application in NLP tasks.

2 Related Work

Reinforcement learning from human feedback (RLHF) has been widely explored as a promising and potent method for aligning language models with human values and preferences. Christiano et al. (2017) initially demonstrated the effectiveness of using human feedback to guide reinforcement learning policies. Further research by Ziegler et al. (2019) and Stiennon et al. (2020) has shown improvements in language model alignment through RLHF, notably in text summarization and dialogue generation tasks. Despite such advances in research and implementation, RLHF methods typically involve complex and computationally demanding procedures due to their reliance on training separate reward models.

Recently, alternative approaches like Direct Preference Optimization (DPO) introduced by Rafailov et al. (2023) have sought to simplify the preference alignment process. DPO directly optimizes language models using a classification-based objective without the explicit reinforcement learning phase, thus offering a stable, computationally efficient solution. Preliminary evaluations indicated that DPO can achieve performance that is either comparable or superior to traditional RLHF methods in tasks such as sentiment modulation and text summarization while requiring less computational overhead, demonstrating its potential for more widespread applications in improving GPT output quality and reducing computation demands.

Our work leverages and extends the methodology proposed by Rafailov et al. (2023), investigating its effectiveness in sentiment analysis, paraphrase detection, and sonnet generation. By building upon these foundational studies, we aim to demonstrate the generalizability and efficacy of DPO in diverse NLP scenarios.

3 Approach

Our approach leverages the GPT-2 architecture [1], fine-tuned using the Direct Preference Optimization method introduced by Rafailov et al. (2023) [2]. We apply GPT-2 across three NLP tasks: sentiment analysis, paraphrase detection, and sonnet generation, each requiring specific adaptations in training strategy.

The core of our implementation is the GPT-2 model, which comprises three main components: embedding layers, transformer layers, and final prediction heads tailored to the specific tasks.

The GPT-2 embedding layers incorporate both token embeddings and positional embeddings to encode input sequences as follows:

$$\mathbf{H}_0 = \text{Embedding}(X_{\text{tokens}}) + \text{PositionalEmbedding}(X_{\text{positions}}) \tag{1}$$

Each Transformer layer within GPT-2 employs a causal self-attention mechanism, computed as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}} + M\right)V \tag{2}$$

where Q, K, and V represent queries, keys, and values respectively, and M denotes a causal attention mask enforcing autoregressive behavior by masking out future token information. The self-attention is followed by a position-wise feed-forward neural network, normalization layers, residual connections, and dropout operations.

For each specific task, we add distinct prediction heads atop GPT-2's final hidden states to generate appropriate outputs:

- For **sentiment analysis**, we use a linear classification head producing logits over sentiment classes.
- For **paraphrase detection**, we use a linear classification head over two possible outcomes (paraphrase or not).

- For **sonnet generation**, the model directly predicts logits over the vocabulary tokens, modeling the probability distribution for generating each token of the sonnet.

To align GPT-2 outputs with human preferences efficiently, we employ DPO. Here, DPO simplifies and stabilizes model alignment by directly optimizing model predictions towards human-preferred outputs.

The DPO objective is defined as follows, as proposed by Rafailov et al. (2023):

Given a dataset $\mathcal{D}$ containing human-labeled pairs of preferred (y_w) and dispreferred (y_l) responses for each prompt x, the DPO loss function is:

$$L_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \left[\log \sigma \left(\beta \log \frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta \log \frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)} \right) \right] \tag{3}$$

σ denotes the logistic sigmoid function, β controls the strength of optimization towards preferred outputs, and π_{ref} is a fixed reference model (initially set to the GPT-2 model before fine-tuning). Intuitively, DPO maximizes the likelihood ratio between preferred and dispreferred outputs, effectively guiding GPT-2 to generate outputs aligned with human judgments.

For sentiment classification, we fine-tune GPT-2 on the Stanford Sentiment Treebank (SST) and the CFIMDB dataset, using the following training objective based on standard cross-entropy loss:

$$L_{\text{CE}} = -\frac{1}{N} \sum_{i=1}^{N} \log p_\theta(y_i|x_i) \tag{4}$$

We experimented with fine-tuning either just the final classification head or the entire GPT-2 model.

In paraphrase detection, we structure inputs using the format:

"Is "{sentence1}" a paraphrase of "sentence2"? Answer "yes" or "no":"

We then predict either the "yes" or "no" token as a classification task. When using DPO loss, predictions from the fine-tuned model (π_θ) were directly compared against the frozen reference model (π_{ref}):

$$L_{\text{DPO}} = -\frac{1}{N} \sum_{i=1}^{N} \log \sigma \left(\beta \log \frac{p_\theta(y_i^+|x_i)}{p_{\text{ref}}(y_i|x_i)} - \beta \log \frac{p_\theta(y_l^{(i)}|x_i)}{p_{\text{ref}}(y_l^{(i)}|x_i)} \right) \tag{5}$$

where $y_w^{(i)}$ denotes the preferred output token and $y_l^{(i)}$ the non-preferred output token.

For generating sonnets, we applied DPO loss to guide GPT-2 in generating coherent sonnet continuations. Our method computes the likelihood of good and bad (suboptimally generated) continuations:

$$L_{\text{DPO}} = -\mathbb{E}\left[\log \sigma \left(\beta[(\log p_\theta(y_{\text{good}}|x) - \log p_{\text{ref}}(y_{\text{good}}|x)) - (\log p_\theta(y_{\text{bad}}|x) - \log p_{\text{ref}}(y_{\text{bad}}|x))]\right)\right] \tag{6}$$

Bad continuations were generated by sampling at higher temperatures or top-p thresholds to intentionally reduce the quality of their outputs.

All models were trained using the AdamW optimizer with appropriate hyperparameters, selected through validation set performance. We ensured reproducibility by setting a fixed random seed across all experiments. The training was performed using NVIDIA GPUs.

This approach combines the strength of transformer-based language models with DPO's straightforward alignment strategy,

4 Experiments

4.1 Data

In our project, we employ several datasets to evaluate our fine-tuned GPT-2 model. Below, we provide details on each dataset and explain their relevance to the respective tasks.

Stanford Sentiment Treebank (SST) The Stanford Sentiment Treebank (SST) [3] is a popular dataset for sentiment classification tasks. It contains sentences extracted from movie reviews, each labeled with one of five sentiment classes ranging from very negative (0) to very positive (4). The dataset includes 11,855 sentences for training, 872 for validation, and 1,821 for testing. It serves as a benchmark for evaluating models' abilities to capture fine-grained sentiment nuances.

CFIMDB Dataset The CFIMDB dataset [4] is a large-scale movie review dataset commonly used for sentiment classification tasks. The dataset includes 50,000 movie reviews labeled as positive or negative, split evenly into 25,000 training and 25,000 test reviews. This dataset was chosen to evaluate our model's ability to generalize sentiment classification performance across larger and more diverse textual data compared to SST.

Quora Question Pairs (QQP) For paraphrase detection, we utilize the Quora Question Pairs dataset [5], consisting of sentence pairs labeled as either paraphrases or non-paraphrases. QQP comprises approximately 404,290 question pairs, designed to test models' capacity for semantic equivalence identification. This dataset allows us to assess how effectively our fine-tuned GPT-2 model distinguishes subtle differences in meaning between sentences.

Shakespearean Sonnets Dataset For sonnet generation, we use a dataset consisting of 154 Shakespearean sonnets, providing structured poetic text for evaluating generative models. The dataset contains sonnets following traditional poetic structures, allowing us to examine GPT-2's ability to generate creative, coherent, and stylistically consistent text when conditioned on partial prompts. Specifically, we provide the model with the first three lines of a sonnet and evaluate its continuation capabilities qualitatively in alignment with poetic form and style.

For each dataset, input text sequences were tokenized using the GPT-2 tokenizer provided by Hugging Face Transformers. Tokenized sequences were padded or truncated to a fixed maximum sequence length, appropriate for computational constraints. Sentences from SST and CFIMDB datasets were directly tokenized and padded. In paraphrase detection, sentence pairs were formatted into a natural-language prompt as follows:

> "Is "sentence-one" a paraphrase of "sentence-two"? Answer "yes" or "no":"

This framing allows the model to leverage its language-modeling capabilities to perform classification by predicting tokens corresponding to "yes" or "no."

For sonnet generation, inputs consisted of tokenized lines of poetry, and the model was fine-tuned to produce coherent poetic continuations from provided prompts.

Overall, these diverse datasets allowed us to rigorously evaluate GPT-2 and the efficacy of Direct Preference Optimization across various NLP tasks, from classification to creative generation.

4.2 Evaluation method

We evaluate our fine-tuned GPT-2 models across all three NLP tasks—sentiment analysis, paraphrase detection, and sonnet generation—using both quantitative and qualitative metrics appropriate to each task.

For sentiment analysis (on Stanford Sentiment Treebank (SST) [3] and CFIMDB [4]), we use two standard metrics: accuracy and macro-F1 score. Accuracy measures the percentage of predictions matching the correct labels, providing an intuitive measure of overall classification performance. The macro-F1 score was a tool in our development process but excluded in our final evaluations.

For paraphrase detection on the Quora Question Pairs (QQP) dataset [5], we also employ accuracy and macro-F1.

For sonnet generation, we evaluate our generated sonnets using the chrF score [**?**], a character-level metric used to assess the quality of text generation. chrF computes the F-score of character n-grams between the generated text and reference sonnets, thus effectively capturing both fluency and stylistic similarity:

$$\text{chrF} = (1 + \beta^2) \cdot \frac{\text{chrPrecision} \cdot \text{chrRecall}}{(\beta^2 \cdot \text{chrPrecision}) + \text{chrRecall}} \tag{7}$$

where β determines the relative importance between precision and recall.

4.3 Experimental details

For all experiments, we consistently used the AdamW optimizer [**?**] with a learning rate of 1×10^{-5}, a batch size of 8, and dropout probability set at 0.1. Each experiment was trained for exactly 10 epochs, ensuring consistency and reproducibility. Training and evaluation runs were done on NVIDIA T4 GPUs.

For sentiment analysis on the Stanford Sentiment Treebank (SST) and CFIMDB datasets, each training run completed within approximately 10 minutes.

Paraphrase detection, trained on the larger Quora Question Pairs dataset, required approximately 9 hours per run with and without DPO enabled.

For sonnet generation, training time was approximately 15 minutes per run with DPO enabled, and 10 minutes with standard loss calculations.

All models and experiments used fixed random seeds to guarantee reproducibility and consistency across experimental runs.

4.4 Results

Table 1 summarizes the sentiment classification results. On the Stanford Sentiment Treebank (SST), fine-tuning the full GPT-2 model outperformed fine-tuning only the last linear layer and improved significantly over baseline results. On the CFIMDB dataset, our model closely matched the baseline performance.

Model	**SST Test Accuracy**	**CFIMDB Dev Accuracy**
Baseline (last linear layer)	46.2%	86.1%
Our model (last linear layer)	46.6%	85.7%
Baseline (full model)	51.3%	97.6%
Our model (full model)	**55.2%**	**97.6%**

Table 1: Comparison of sentiment analysis performance on SST and CFIMDB datasets.

Our GPT-2 model showed substantial improvement over baseline performance on the SST dataset when fully fine-tuned, while maintaining comparable results on CFIMDB.

For paraphrase detection on the Quora Question Pairs dataset, our GPT-2 model trained using Direct Preference Optimization achieved a test accuracy of **87.9%**.

In the sonnet generation task, our GPT-2 model trained with DPO achieved a chrF score of **35.902**, surpassing the standard GPT-2 fine-tuning baseline (chrF approximately 34.9) by about 1 point. This improvement highlights the advantage of the DPO approach in generating stylistically closer poetic text.

We found that DPO didn't demonstrate as large of a qualitative gap as expected - while it outperformed our standard loss, especially in sonnet generation, it remained close, regardless of hyperparameter tuning. We found that if the model performed poorly with standard loss, it performed poorly with DPO, and if the model performed well with standard loss, it performed well with DPO. This inclines us to consider other implementation methods of DPO to see if DPO performed strongly in situations where other methods perform poorly.

5 Analysis

In sentiment analysis on the SST dataset, fine-tuning the entire GPT-2 model provided clear improvements over the baseline by effectively capturing nuanced sentiment. On the CFIMDB dataset, differences between partial fine-tuning (final linear layer only) and full model fine-tuning were more pronounced, with the fully fine-tuned model demonstrating significantly greater robustness on challenging examples.

In paraphrase detection, our GPT-2 model fine-tuned with DPO achieved higher accuracy compared to traditional fine-tuning approaches. Qualitative inspection revealed stronger sensitivity to subtle semantic differences in sentence pairs, though the model occasionally misclassified pairs with significant lexical similarity but different underlying meanings.

In sonnet generation, employing DPO improved the chrF score by approximately 1 point compared to the standard GPT-2 fine-tuning baseline. Qualitatively, sonnets produced by our DPO-trained model exhibited better adherence to the formal poetic structure and style. However, we observed that even with DPO, generated sonnets frequently suffered from repetition, limiting their overall creativity and diversity. For example, here is one of the generated sonnets in our final model:

"Poor soul, the center of my sinful earth,
Pressed with these rebel powers that thee array,
Why dost thou pine within and suffer dearth,
Pressed and deost thou, of these dost, my, the earth, that thou and my deost,
Pressed with thee, with that, my, dost thou and my earth,
Pressed with thee, my, with my deost, my deost, thee, the earth,
Pressed with thee, my, the deost and thee, my, that with thee
Pressed and deost thou and that with that, my earth and my deost, the earth, my, that
Pressed and dost thee and that, my earth, and my deost, my"

Despite careful experimentation and hyperparameter tuning—including variations in temperature, top-k, and nucleus sampling (top-p)—the model consistently demonstrated strong repetition, indicating challenges inherent in the task and dataset, but also significant room for improvement.

While DPO provided improvements in stylistic alignment and training stability, limitations in creative diversity and output repetition suggest areas for further improvement, refinement, and experimentation.

6 Conclusion

In this project, we explored fine-tuning GPT-2 using both traditional fine-tuning methods and Direct Preference Optimization (DPO) in sentiment analysis, paraphrase detection, and sonnet generation. Our experiments demonstrated that leveraging DPO provides clear benefits in aligning GPT-2 outputs more closely with human preferences, particularly evident in the paraphrase detection task (achieving 87.9% accuracy) and the sonnet generation task (improving the chrF score by approximately 1 point over the baseline), but still has room for adjusting for ideal application.

While the DPO approach effectively enhanced performance and provided greater stability during training, we encountered notable limitations. In particular, our DPO-trained model struggled with repetitiveness during sonnet generation. Despite careful tuning of hyperparameters such as temperature, top-p, and other sampling parameters, the generated outputs often lacked diversity and creative variation, indicating inherent limitations in the current DPO methodology or the inherent complexity of the creative generation task itself.

Future work could focus on addressing these shortcomings by exploring advanced sampling strategies, such as nucleus sampling with varying thresholds, or employing additional regularization methods during fine-tuning. Further investigation into alternative methods of preference alignment, such as incorporating human feedback directly in the generation loop or applying parameter-efficient fine-tuning methods, may also yield promising improvements.

Cur findings highlight the potential of Direct Preference Optimization as a practical and efficient method for aligning transformer-based language models with human preferences, while also emphasizing the challenges that remain in producing diverse and creative outputs in generative tasks.

Team contributions (Required for multi-person team)

Provide a brief summary of what each team member did for the project (about 1 or 2 sentences per person).

Onyie Okoye: Setup GCP, wrote most of the code, ran training for all models. Natalie Shell: Wrote the project proposal, milestone, and final project report, aided with bug fixing for code.

References

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

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

[3] Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

[4] Andrew L Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, 2011.

[5] Zhiguo Wang, Wael Hamza, and Radu Florian. Bilateral multi-perspective matching for natural language sentences. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4144–4150, 2017.