The Pupil Becomes the Master: Eye-Tracking Feedback for Tuning LLMs

Samuel Kiegeland[1] **David R. Reich**[2] **Ryan Cotterell**[1] **Lena A. Jäger**[2 3] **Ethan Wilcox**[1]

Abstract

Large language models often require alignment with explicit human preferences, which can be sparse and costly. We propose a framework to leverage eye-tracking data as an implicit feedback signal to tune LLMs for controlled sentiment generation using Direct Preference Optimization. Our study demonstrates that eye-tracking feedback can be a valuable signal for tuning LLMs. This motivates future research to investigate the impact of eye-tracking feedback on various tasks, highlighting the potential of integrating eye-tracking data with LLMs to improve their performance and alignment with human preferences.

1. Introduction

Recent advancements in large language models (LLMs) have significantly transformed natural language processing (NLP). While LLMs offer impressive capabilities in NLP, they often require fine-tuning and alignment techniques, such as Reinforcement Learning from Human Feedback (RLHF; Christiano et al., 2017; Ouyang et al., 2022), to optimize their performance for specific tasks and to better align their outputs with human preferences. By incorporating human feedback into the training process, RLHF has significantly improved the ability of LLMs to follow human instructions while decreasing the generation of toxic or harmful content (Stiennon et al., 2020; Ouyang et al., 2022).

However, RLHF, particularly when done using Proximal Policy Optimization (PPO; Schulman et al., 2017), is overly sensitive to hyperparameters and can be unstable (Casper et al., 2023; Rafailov et al., 2023; Ahmadian et al., 2024). Moreover, RLHF typically requires fitting a reward model, which increases the complexity of the procedure (Rafailov et al., 2023).

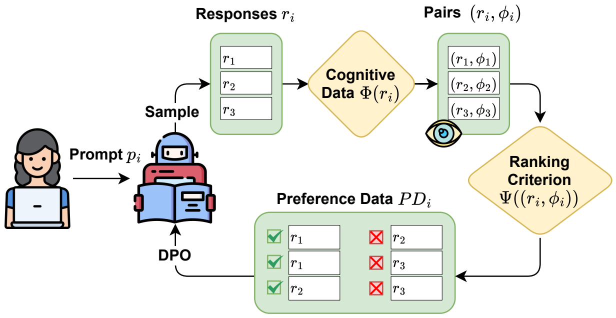

Figure 1. Framework for using cognitive data such as eye-tracking data for tuning LLMs. We sample responses from a model and obtain cognitive data for them. Each pair of responses is ranked using various criteria, such as minimizing fixation durations resulting in a preference dataset for DPO. See §3 for details.

To address this, Rafailov et al. (2023) introduced Direct Preference Optimization (DPO), which directly optimizes a language model to human preferences without needing an external reward model. While optimization with DPO can lead to higher stability than PPO, it still relies on explicit human feedback to construct a preference dataset. This human feedback typically consists of individuals rating or ranking language model responses, which presents challenges due to its sparsity and high cost (Casper et al., 2023). Moreover, this type of human judgment occurs after reading a model's response, which may not capture the real-time cognitive processes involved in language understanding.

Other types of responses, such as eye-tracking signals, can be captured directly while reading and offer a real-time measure of cognitive processing during reading. A growing body of research demonstrates the potential of eye-tracking data to shed light on cognitive processes during reading, such as attention allocation, information integration, and text comprehension (Just and Carpenter, 1980; Rayner et al., 1989; Reichle et al., 1998).

This paper explores to what extent cognitive data, such as eye-tracking data, can be used to align LLMs. We propose a framework that uses eye-tracking data as a feedback signal to optimize large language models for controlled sentiment

[*]Equal contribution [1]ETH Zürich [2]University of Potsdam [3]University of Zürich. Correspondence to: Samuel Kiegeland <skiegeland@ethz.ch>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

generation. We compare the difference between explicit (offline) judgments via ratings and implicit (online) judgments via eye-tracking to construct preference datasets and tune language models using DPO. Our study shows that eye-tracking feedback can be a valuable signal for tuning LLMs on these tasks.

2. Background & Related Work

We seek to tune language models with DPO using eye-tracking data as a feedback signal.

2.1. Direct Preference Optimization (DPO)

Unlike traditional methods for RLHF (Christiano et al., 2017; Stiennon et al., 2020; Ouyang et al., 2022), which involve training a reward model from a dataset of human ratings, DPO (Rafailov et al., 2023) offers an alternative approach for optimizing language models based on human feedback. Rather than relying on a separate reward model, DPO uses direct preference comparisons between pairs of model-generated responses. This method removes the dependency on the explicit reward model and instead trains an implicit reward model.

2.2. Eye-Tracking in NLP

Recent research has demonstrated the benefit of integrating eye-tracking data with NLP models to enhance performance on downstream tasks such as part-of-speech tagging (Barrett et al., 2016), text simplification (Klerke et al., 2016; Higasa et al., 2024), relation classification (Hollenstein et al., 2019; McGuire and Tomuro, 2021), text readability (González-Garduño and Søgaard, 2017; Hollenstein et al., 2022) and sarcasm detection and understanding (Mishra et al., 2016a;b; 2017).

Yang and Hollenstein (2023) enhance sentiment classification using human scanpaths, which are index sequences of fixated words. Building on these results, Deng et al. (2023a) achieve comparable improvements with synthetically generated scanpaths. Khurana et al. (2023) further validate synthetic scanpaths across various GLUE tasks (Wang et al., 2018). These studies highlight the potential of eye-tracking data, both human-generated and synthetic, as a valuable resource for improving NLP models across various tasks. However, there is a lack of research on integrating eye-tracking for tuning large language models, e.g., via DPO.

3. Real-Time Feedback for tuning LLMs

We present a framework for fine-tuning language models using cognitive data, such as eye-tracking data. The framework consists of several key components: a language model $\mathrm{p_{LM}}$ (defined as a distribution over Σ^*, where Σ is an alphabet), a set of prompts $P = \{p_1, p_2, \ldots, p_n\}$ (where $p_i = x_1, \ldots, x_m$ and $x_j \in \Sigma$), responses r_i sampled from the language model ($r_i \sim \mathrm{p_{LM}}(\cdot \mid p_i)$), a cognitive data collection function Φ, and a ranking criterion Ψ.

The cognitive data collection function Φ takes a response r_i as input and returns the corresponding cognitive data ϕ_i, such as eye-tracking data:

$$\Phi : \mathcal{R} \to \mathcal{C}, \quad \Phi(r) = \phi$$

where $\mathcal{R}$ is the space of possible responses and $\mathcal{C}$ is the space of cognitive data. The ranking criterion Ψ takes two response-cognitive data pairs (r_i, ϕ_i) and (r_j, ϕ_j) as input and returns a ranking, where $r_i \succ r_j$ (response i is preferred over response j), $r_i \prec r_j$ (response j is preferred over response i), or $r_i \simeq r_j$ (responses are equally preferred):

$$\Psi : (\mathcal{R} \times \mathcal{C}) \times (\mathcal{R} \times \mathcal{C}) \to \{\succ, \prec, \simeq\}$$

Applying the framework consists of the following steps, see Figure 1 for an overview. For each prompt $p_i \in P$ the language model $\mathrm{p_{LM}}$ generates a set of responses $\mathcal{R}_i = \{r_{i1}, r_{i2}, \ldots, r_{iK}\}$, where K is the number of responses sampled for prompt p_i. The cognitive data collection function Φ is then applied to each response in $\mathcal{R}_i$ to obtain the corresponding cognitive data $\mathcal{C}_i = \{\phi_{i1}, \phi_{i2}, \ldots, \phi_{iK}\}$, where $\phi_{ij} = \Phi(r_{ij})$ for $j = 1, 2, \ldots, K$. The ranking criterion Ψ is applied to all pairs of response-cognitive data pairs in $(\mathcal{R}_i, \mathcal{C}_i)$ to create preference data

$$\mathcal{PD}_i = \{(p_i, \Psi((r_{ij}, \phi_{ij}), (r_{il}, \phi_{il}))) \mid 1 \leq j < l \leq K\}$$

The complete preference dataset $\mathcal{PD}$ is obtained by aggregating the preference data from all prompts: $\mathcal{PD} = \bigcup_{i=1}^{N} \mathcal{PD}_i$ and the language model $\mathrm{p_{LM}}$ is optimized using the preference data $\mathcal{PD}$, e.g., using DPO.

4. Approach

Building on previous research highlighting the benefits of eye-tracking data for sentiment detection, we choose controlled sentiment generation to test our framework. We condition the language model on sentiment tags $t \in \{[\text{positive}], [\text{negative}]\}$ to generate completions for prefixes x, where x represents the first three words of a movie review, guiding the sentiment of the generated text based on the tags.

4.1. Data

Due to the lack of an existing dataset that includes human judgments and eye-tracking data for language model-generated responses, we directly apply DPO to the ETSA-II dataset introduced by Mishra et al. (2016a). Figure 2 presents an overview of the complete pipeline. The ETSA-II dataset contains 994 sentences from websites with sarcastic quotes, Tweets, and movie reviews (Pang and Lee, 2004), each labeled for sentiment (positive or negative) and sarcasm presence. 38.5% of the sentences have a positive sentiment, while 61.5% have a negative sentiment. 35.2% of the sentences are labeled as sarcastic, with 93.7% of these also having negative sentiments. The dataset includes eye-tracking data from 7 participants reading the sentences, along with their subjective sentiment judgments, making it suitable for studying explicit (offline) and implicit (online) judgments. The participants are graduate students who are non-native English speakers with ToEFL-iBT scores of 100 or higher.

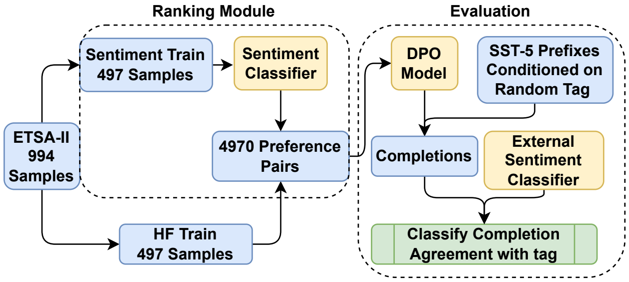
Ranking Module
Evaluation
Sentiment Train 497 Samples
Sentiment Classifier
ETSA-II 994 Samples
4970 Preference Pairs
HF Train 497 Samples
DPO Model
SST-5 Prefixes Conditioned on Random Tag
Completions
External Sentiment Classifier
Classify Completion Agreement with tag

Figure 2. We randomly split the ETSA-II data into two subsets with 497 instances each. We train sentiment classifiers on the Sentiment Train split and create preference datasets from the HF Train split by classifying sentences and sampling ten sentences from the opposite class. After DPO, we sample tags and generate completions, then classify if the completion matches the tag's sentiment.

4.2. Ranking Criteria

To construct preference datasets ($\mathcal{PD}$) for DPO, we apply various ranking criteria (Ψ) based on the dataset's annotations and classifiers that utilize textual features and eye-tracking data (see Table 1 for an overview). As an additional baseline, we randomly pair sentences from the ETSA-II dataset. We conduct experiments using the ground truth annotation and individual participant annotations.

We test several eye-tracking criteria using the word-level fixation durations and scanpaths. Motivated by Mishra et al. (2016b), who found that sarcastic sentences, often carrying a negative sentiment, are typically associated with more complex scanpaths and longer fixation durations, we adopt the following approach: Responses exhibiting higher values in mean fixation duration, mean number of refixations, total fixation duration, or total number of refixations are classified as expressing a negative sentiment.

For classifiers trained on the Sentiment Train split (BERT, PLM-AS, SYNSP) of the ETSA-II dataset, we adapt the implementations from Deng et al. (2023a)[1] to our custom data split. BERT denotes the pre-trained, cased English model from Huggingface[2]. PLM-AS is a re-implementation of the architecture presented by Yang and Hollenstein (2023), which augments BERT with a scanpath encoder. The scanpath encoder re-orders the textual embeddings from BERT according to the fixation sequence of the human reader. This output is then fed into a Gated Recurrent Unit (GRU; Cho et al., 2014) for the final sentiment classification. The SYNSP model is a modification presented by Deng et al. (2023a), which replaces human scanpaths with synthetic ones generated by the Eyettention model (Deng et al., 2023b). We use the default parameters to generate 7 synthetic scanpaths. Both PLM-AS and SYNSP models exclusively use text and scanpaths for predicting sentiment labels.

Table 1. Ranking criteria to construct preference datasets. All features (text, eye-tracking, and annotations) are from the ETSA-II dataset. For the BERT, PLM-AS, and SYSNP models, we adapt the implementations from Deng et al. (2023a) to our data split.

Criterion Ψ	Pre-training	Type of Data
Random	-	-
Ground Truth	-	Annotation
Subj. Judgement	-	Annotation
Mean Fix Dur	-	Human Reading Time
Mean Refixations	-	Human Scanpath
Total Fix Dur	-	Human Reading Time
Total Refixations	-	Human Scanpath
BERT	ETSA-II$_{\text{Sentiment}}$	Text
PLM-AS	ETSA-II$_{\text{Sentiment}}$	Text + Human Scanpath
PLM-AS Subj.	ETSA-II$_{\text{Sentiment}}$	Text + Human Scanpath
SYNSP Model	ETSA-II$_{\text{Sentiment}}$	Text + Synth. Scanpath
DistilBERT	IMDB	Text

4.3. Training

We use the DPO implementation from the Transformer Reinforcement Learning (TRL) library (von Werra et al., 2020) to tune language models for generating sentences with positive or negative sentiments. We select GPT2-small [3], which has been fine-tuned on the IMDB movie review dataset (Maas et al., 2011), as our base model. We fine-tune the model on the ETSA-II dataset's texts to address the distribution shift. In this step, we exclude tags used for controlled sentiment generation (see App. D for details).

We randomly divide the 994 instances into two equally sized

[1] https://github.com/aeye-lab/EMNLP-SyntheticScanpaths-NLU-PretrainedLM

[2] https://huggingface.co/google-bert/bert-base-cased

[3] https://huggingface.co/lvwerra/gpt2-imdb

datasets, as illustrated in Figure 2. The Sentiment Train split is exclusively used for training Sentiment classifiers (see Figure 2). In contrast, the HF Train split is used to construct preference datasets for DPO (see App. A for the parameters). We train three models for each criterion using three random seeds and report means and standard errors.

4.4. Evaluation

We sample sentence prefixes from the SST-5 dataset (Socher et al., 2013) for evaluation. We randomly select 500 prefixes x from the test split, each consisting of the first three words of a sentence. We randomly sample a sentiment tag t for each prefix and construct prompts in the format $p = t$ EOS x, where EOS represents the end-of-sentence token. We then generate completions of 50 tokens using beam search. See App. A for the complete generation settings. To assess the sentiment of the generated text, we utilize two sentiment classifiers: a DistilBERT (Sanh et al., 2019) classifier[4] pre-trained on the IMDB movie review dataset, and GPT-4o (OpenAI, 2024). We compare the predicted sentiment of the generated completions with the assigned tag t.

5. Results

Figure 3 shows the performance of both classifiers on the ETSA-II dataset, evaluated against the ground truth annotations. The results indicate that both classifiers perform well in instances without sarcasm. However, sarcastic sentences prove challenging, especially for the DistilBERT classifier.

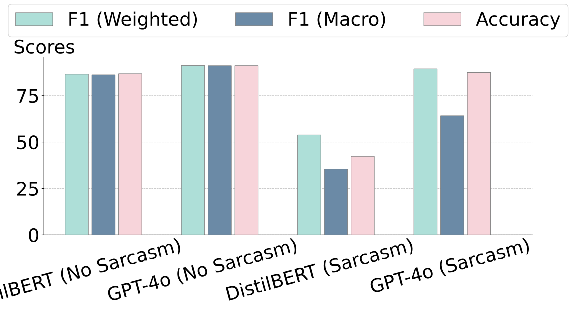

Figure 3. Accuracies and F1 scores for sentiment classifiers on the ETSA-II dataset indicating strong performance on non-sarcastic instances but difficulties with sarcastic ones. DistilBERT, in particular, shows a notable performance drop on sarcastic instances compared to non-sarcastic ones.

5.1. Comparing Ranking Criteria

The results in Table 2 detail the performance of various ranking criteria on the ETSA-II dataset. The random baseline does not lead to any improvements in either model's performance. Similarly, we observed no significant effects for eye-tracking metrics based on fixation duration or refixation count. This suggests that while sarcasm may be associated with reading times or scanpath complexity, these effects may not generalize to all negatively-valenced texts or are too subtle to be useful for fine-tuning language models for sentiment generation.

In contrast, the ground truth ranking criterion significantly outperforms the fine-tuned baseline (FT BL). The results for criteria pre-trained on the sentiment split of ETSA-II indicate that incorporating eye-tracking data can enhance performance. Notably, PLM-AS and SYSNP outperform the BERT classifier, underscoring the value of eye-tracking information. While the DistilBERT-based ranking criterion acts as an upper bound when employed for training and evaluation, PLM-AS and SYNSP exhibit comparable performance when evaluated using GPT-4o. Due to the challenges presented by sarcastic instances, we also conduct experiments excluding them from the dataset (see App. C).

Table 2. Mean accuracy, F1 scores, and standard errors for different ranking criteria on the ETSA-II dataset. The random ranking shows no improvements, but all criteria pre-trained on the sentiment split significantly outperform the baseline. Eye-tracking-based metrics (PLM-AS, SYNSP) result in higher performance than the text-only BERT classifier. Significant improvements over the fine-tuned baseline (FT BL) are marked with * ($p < 0.01$).

Model	$\text{F1}_{\text{Distilbert}}$	$\text{F1}_{\text{GPT-4o}}$	$\text{Acc}_{\text{Distilbert}}$	$\text{Acc}_{\text{GPT-4o}}$
FT BL	53.12	52.48	53.11	53.31
Random	$52.26_{1.78}$	$50.84_{0.82}$	$52.34_{1.75}$	$51.40_{0.91}$
Ground Truth	$86.43^{*}_{0.55}$	$78.45^{*}_{0.31}$	$86.52^{*}_{0.54}$	$79.37^{*}_{0.24}$
Mean Fix Dur	$52.59_{1.65}$	$51.73_{1.12}$	$52.67_{1.65}$	$52.27_{1.13}$
Mean Refixations	$52.73_{1.65}$	$51.89_{1.38}$	$52.81_{1.65}$	$52.47_{1.40}$
Total Fix Dur	$50.97_{1.54}$	$51.52_{1.56}$	$51.05_{1.45}$	$52.05_{1.65}$
Total Refixations	$51.38_{1.53}$	$51.52_{1.65}$	$51.45_{1.45}$	$52.15_{1.75}$
BERT	$82.45^{*}_{0.13}$	$73.96^{*}_{0.33}$	$82.78^{*}_{0.11}$	$75.50^{*}_{0.26}$
PLM-AS	$84.69^{*}_{1.35}$	$78.89^{*}_{1.34}$	$84.85^{*}_{1.33}$	$79.77^{*}_{1.25}$
SYNSP	$85.70^{*}_{0.73}$	$78.53^{*}_{1.45}$	$85.78^{*}_{0.71}$	$79.44^{*}_{1.32}$
DistilBERT	$85.90^{*}_{1.15}$	$77.83^{*}_{2.00}$	$85.98^{*}_{1.13}$	$78.84^{*}_{1.78}$

5.2. Individual Differences

We evaluate the effect of using each individual's judgments to rank sentences and compare the results with specific PLM-AS classifiers, where each classifier is trained only on those individuals' scanpaths and the respective text. The results in Figure 4 show that using the custom classifiers

[4] lvwerra/distilbert-imdb

leads to similar or superior performance compared to the individual's judgments in every case (See App. B for detailed results). These findings suggest that personalized classifiers leveraging implicit (online) eye-tracking feedback can be beneficial for improving performance compared to relying solely on explicit (offline) judgments.

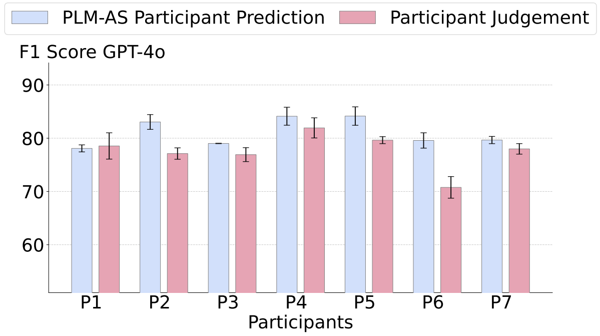

Figure 4. Comparison of F1 scores between participants' ratings and classifiers trained on eye-tracking data and text. For each participant, the classifier trained on their eye-tracking data and the text leads to comparable or improved performance in relation to the participants' own ratings.

5.3. Ranking Model Generated Responses

In a final set of experiments, we use our pre-trained model to generate completions for IMDB movie reviews. We rank responses generated by the baseline model using criteria that do not rely on actual eye-tracking data. The results in Table 3 demonstrate that both the DistilBERT and SYNSP criteria lead to significant improvements. These findings provide additional motivation for collecting eye-tracking on model-generated responses to investigate its potential for tuning language models.

Table 3. Mean accuracy, F1 scores, and standard errors for ranking model-generated responses by different criteria. SYNSP notably improves performance, motivating the collection of eye-tracking data for tuning language models. Significant improvements over the fine-tuned baseline (FT BL) are marked with * ($p < 0.01$).

Model	$\text{F1}_{\text{Distilbert}}$	$\text{F1}_{\text{GPT-4o}}$	$\text{Acc}_{\text{Distilbert}}$	$\text{Acc}_{\text{GPT-4o}}$
FT BL	53.12	52.48	53.11	53.31
Random	$52.69_{1.30}$	$51.69_{0.65}$	$52.74_{1.31}$	$52.14_{0.67}$
SYNSP	$90.97^{*}_{0.36}$	$94.59^{*}_{0.58}$	$90.99^{*}_{0.35}$	$94.59^{*}_{0.57}$
DistilBERT	$93.38^{*}_{0.51}$	$93.85^{*}_{0.30}$	$93.39^{*}_{0.51}$	$93.86^{*}_{0.29}$

6. Conclusion

We introduced a framework for utilizing direct feedback via eye-tracking data to optimize LLMs for controlled sentiment generation. Our study suggests that this type of feedback can improve model adaptability and performance, encouraging further research into using cognitive data to tune large language models and promoting more human-centric and cognitively informed NLP systems.

Limitations

Our study presents several limitations. First, we focus on tuning models for controlled sentiment generation. Future studies could investigate using eye-tracking data for tuning models on other tasks or explore the general impact of tuning large language models on eye-tracking data. Second, our study is limited to one specific dataset, and it would be beneficial to investigate the effectiveness of our approach on other datasets. Third, we focused on a particular method of incorporating eye-tracking data into the training process by defining a ranking criterion and creating different preference datasets for Direct Preference Optimization. Future research could explore alternative ways to use this information for model improvement. Finally, future studies could directly collect eye-tracking data for model-generated responses to gain deeper insights into how humans perceive and process the output of language models.

Impact Statement

Our research introduces a framework to align LLMs with human preferences using eye-tracking data and has many potential societal consequences. Whenever direct feedback via eye-tracking is available, it could serve as an alternative or complementary feedback signal, improving the accuracy and nuance of language models. However, it is important to be aware of any potential privacy issues associated with collecting or processing eye-tracking data (Jäger et al., 2020; Makowski et al., 2021). In our study, all personally identifiable information was anonymized in the datasets before our access. Additionally, it is important to consider and mitigate potential biases in both the eye-tracking data (Prasse et al., 2022) and language models.

References

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. 2024. Back to basics: Revisiting REINFORCE style optimization for learning from human feedback in LLMs. *Preprint*, arXiv:2402.14740.

Maria Barrett, Joachim Bingel, Frank Keller, and Anders Søgaard. 2016. Weakly supervised part-of-speech tagging using eye-tracking data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 579–584, Berlin, Germany. Association for Computational Linguistics.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Tong Wang, Samuel Marks, Charbel-Raphael Segerie, Micah Carroll, Andi Peng, Phillip Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, Anand Siththaranjan, Max Nadeau, Eric J Michaud, Jacob Pfau, Dmitrii Krasheninnikov, Xin Chen, Lauro Langosco, Peter Hase, Erdem Biyik, Anca Dragan, David Krueger, Dorsa Sadigh, and Dylan Hadfield-Menell. 2023. Open problems and fundamental limitations of reinforcement learning from human feedback. *Transactions on Machine Learning Research*. Survey Certification.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Shuwen Deng, Paul Prasse, David Reich, Tobias Scheffer, and Lena Jäger. 2023a. Pre-trained language models augmented with synthetic scanpaths for natural language understanding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6500–6507, Singapore. Association for Computational Linguistics.

Shuwen Deng, David R. Reich, Paul Prasse, Patrick Haller, Tobias Scheffer, and Lena A. Jäger. 2023b. Eyettention: An attention-based dual-sequence model for predicting human scanpaths during reading. *Proceedings of the ACM on Human-Computer Interaction*, 7(ETRA):1–24.

Rudolf Franz Flesch. 1948. A new readability yardstick. *The Journal of applied psychology*, 32 3:221–33.

Ana Valeria González-Garduño and Anders Søgaard. 2017. Using gaze to predict text readability. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 438–443, Copenhagen, Denmark. Association for Computational Linguistics.

Taichi Higasa, Keitaro Tanaka, Qi Feng, and Shigeo Morishima. 2024. Keep eyes on the sentence: An interactive sentence simplification system for english learners based on eye tracking and large language models. CHI EA '24, New York, NY, USA. Association for Computing Machinery.

Nora Hollenstein, Maria Barrett, Marius Troendle, Francesco Bigiolli, Nicolas Langer, and Ce Zhang. 2019. Advancing NLP with cognitive language processing signals. *CoRR*, abs/1904.02682.

Nora Hollenstein, Itziar Gonzalez-Dios, Lisa Beinborn, and Lena Jäger. 2022. Patterns of text readability in human and predicted eye movements. In *Proceedings of the Workshop on Cognitive Aspects of the Lexicon*, pages 1–15, Taipei, Taiwan. Association for Computational Linguistics.

Lena A. Jäger, Silvia Makowski, Paul Prasse, Sascha Liehr, Maximilian Seidler, and Tobias Scheffer. 2020. Deep Eyedentification: biometric identification using micro-movements of the eye. In *Machine Learning and Knowledge Discovery in Databases*, pages 299–314, Cham. Springer International Publishing.

Marcel A. Just and Patricia A. Carpenter. 1980. A theory of reading: From eye fixations to comprehension. *Psychological Review*, 87(4):329–354.

Varun Khurana, Yaman Kumar, Nora Hollenstein, Rajesh Kumar, and Balaji Krishnamurthy. 2023. Synthesizing human gaze feedback for improved NLP performance. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1895–1908, Dubrovnik, Croatia. Association for Computational Linguistics.

Sigrid Klerke, Yoav Goldberg, and Anders Søgaard. 2016. Improving sentence compression by learning to predict gaze. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1528–1533, San Diego, California. Association for Computational Linguistics.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. 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, Portland, Oregon, USA. Association for Computational Linguistics.

Silvia Makowski, Paul Prasse, David R. Reich, Daniel Krakowczyk, Lena A. Jäger, and Tobias Scheffer. 2021.

DeepEyedentificationLive: oculomotoric biometric identification and presentation-attack detection using deep neural networks. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 3(4):506–518.

Erik McGuire and Noriko Tomuro. 2021. Relation classification with cognitive attention supervision. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pages 222–232, Online. Association for Computational Linguistics.

Abhijit Mishra, Kuntal Dey, and Pushpak Bhattacharyya. 2017. Learning cognitive features from gaze data for sentiment and sarcasm classification using convolutional neural network. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 377–387, Vancouver, Canada. Association for Computational Linguistics.

Abhijit Mishra, Diptesh Kanojia, and Pushpak Bhattacharyya. 2016a. Predicting readers' sarcasm understandability by modeling gaze behavior. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1).

Abhijit Mishra, Diptesh Kanojia, Seema Nagar, Kuntal Dey, and Pushpak Bhattacharyya. 2016b. Harnessing cognitive features for sarcasm detection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1095–1104, Berlin, Germany. Association for Computational Linguistics.

OpenAI. 2024. Hello GPT-4o .

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 271–278, Barcelona, Spain.

Paul Prasse, David R. Reich, Silvia Makowski, Lena A. Jäger, and Tobias Scheffer. 2022. Fairness in oculomotoric biometric identification. In *2022 Symposium on Eye Tracking Research and Applications*, ETRA '22, New York, NY, USA. Association for Computing Machinery.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pages 53728–53741. Curran Associates, Inc.

Keith Rayner, Sara C. Sereno, Robin K Morris, A. Réne Schmauder, and Charles Clifton Jr. 1989. Eye movements and on-line language comprehension processes. *Language and Cognitive Processes*, 4(3-4):SI21–SI49.

Erik D. Reichle, Alexander Pollatsek, Donald L. Fisher, and Keith Rayner. 1998. Toward a model of eye movement control in reading. *Psychological Review*, 105(1):125–157.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. 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, Seattle, Washington, USA. Association for Computational Linguistics.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*, volume 33, pages 3008–3021. Curran Associates, Inc.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, and Shengyi Huang. 2020. TRL: Transformer Reinforcement Learning.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Duo Yang and Nora Hollenstein. 2023. PLM-AS: pre-trained language models augmented with scanpaths for sentiment classification. *Proceedings of the Northern Lights Deep Learning Workshop*.

A. Training & Evaluation Settings

To train the models using DPO, we classify the text from the ETSA-II dataset as either positive or negative based on the ranking criterion Ψ and choose tags $t \in \{$[positive], [negative]$\}$. We format prompts by concatenating the tag with the EOS token and selecting the respective text as the chosen string. For the rejected string, we sample 10 texts from the opposite sentiment as classified by Ψ. We create a validation split using 10% of the training data and select the final checkpoint based on the minimum loss on the validation split. All models are trained with DPO for 10 epochs using the parameters in Table 4. We experiment with different values for β and find that models perform better with higher values of `0.5` compared to lower values of `0.2`.

For evaluation, we condition the models on randomly sampled tags $t \in \{$[positive], [negative]$\}$ and generate completions for prefixes (consisting of three words) from the SST-5 dataset. Our settings for generating responses to evaluate the models are shown in Table 5. To avoid potential bias, we exclude any prefixes that overlap with the movie review samples in the ETSA-II dataset.

Parameter	Setting
β	0.5
Learning Rate	5.0e-7
LR Scheduler	Cosine
Max Length	100
Max Prompt Length	256
Train Epochs	10
Optimizer	AdamW
Batch Size	2
Warmup Ratio	0.1
Seed values	$42, 8, 64$

Table 4. DPO training parameters used for all runs.

Parameter	Setting
Max Length	50
Beams	4
Sample	False
Return Sequences	1
Dataset	SST-5
Sentences	500
Prefix Words	3
Repetition Penalty	1.2
No Repeat N-gram Size	3

Table 5. Evaluation parameters used for all runs.

To evaluate the generations, we classify the sentiment of the generated completions using a DistilBERT classifier [5] and compare the predictions with the sentiment tags, the model was conditioned on. For evaluating models with GPT-4o, we use a temperature of `0.5` and prompt the model using the following messages:

```
{”role”: ”system”, ”content”: ”You are a sentiment classification assistant.”}

{”role”: ”user”, ”content”: f”Detect whether the following text has a positive or
    negative sentiment. Reply with ’positive’ if the sentiment is positive, and
   ’negative’ if it is negative.\n\nText: {text}”}
```

To test for significance, we use the paired permutation test from SciPy [6] for pairings with 1000 resamples.

B. Detailed Results

Table 6 presents a detailed overview of all experimental results, including additional metrics such as text readability measured using the Flesch Reading Ease score (Flesch, 1948) via `textstat`[7] and lexical diversity, which is the ratio of unique words to total words. We observe that all ranking criteria, which increase the alignment of sampled tags with the sentiment of generated text lead to a small drop in text readability while lexical diversity remains unchanged. Random ranking and pure eye-tracking criteria do not lead to improvements in controlled sentiment generation compared to the baseline (FT BL). However, all other criteria significantly improve accuracy and F1 scores for both DistilBERT and GPT-4o classifiers.

[5] `lvwerra/distilbert-imdb`

[6] `https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.permutation_test.html`

[7] `https://github.com/textstat/textstat`

Model	$\text{F1}_{\text{Distilbert}}$	$\text{F1}_{\text{GPT-4o}}$	$\text{Accuracy}_{\text{Distilbert}}$	$\text{Accuracy}_{\text{GPT-4o}}$	Readability	Diversity
FT BL	53.12	52.48	53.11	53.31	87.83	81.06
Random	$52.26_{1.78}$	$50.84_{0.82}$	$52.34_{1.75}$	$51.40_{0.91}$	$88.44_{0.41}$	$81.06_{0.11}$
Ground Truth	$86.43^{*}_{0.55}$	$78.45^{*}_{0.31}$	$86.52^{*}_{0.54}$	$79.37^{*}_{0.24}$	$80.46_{1.37}$	$81.09_{0.21}$
DistilBERT (trained on IMDB)	$85.90^{*}_{1.15}$	$77.83^{*}_{2.00}$	$85.98^{*}_{1.13}$	$78.84^{*}_{1.78}$	$82.39_{1.18}$	$81.80_{0.23}$
Pure Eye-Tracking Criteria						
Mean Fixation Duration	$52.59_{1.65}$	$51.73_{1.12}$	$52.67_{1.65}$	$52.27_{1.13}$	$88.69_{0.34}$	$81.01_{0.14}$
Mean Number Refixations	$52.73_{1.65}$	$51.89_{1.38}$	$52.81_{1.65}$	$52.47_{1.40}$	$88.60_{0.28}$	$81.06_{0.13}$
Total Fixation Duration	$50.97_{1.54}$	$51.52_{1.56}$	$51.05_{1.45}$	$52.05_{1.65}$	$88.62_{0.49}$	$81.22_{0.05}$
Total Number Refixations	$51.38_{1.53}$	$51.52_{1.65}$	$51.45_{1.45}$	$52.15_{1.75}$	$88.56_{0.47}$	$81.20_{0.04}$
Classifier Trained on the Sentiment Split of ETSA-II (see Figure 2)						
BERT	$82.45^{*}_{0.13}$	$73.96^{*}_{0.33}$	$82.78^{*}_{0.11}$	$75.50^{*}_{0.26}$	$82.67_{1.00}$	$81.02_{0.37}$
PLM-AS	$84.69^{*}_{1.35}$	$78.89^{*}_{1.34}$	$84.85^{*}_{1.33}$	$79.77^{*}_{1.25}$	$79.42_{1.33}$	$80.73_{0.43}$
SYNSP	$85.70^{*}_{0.73}$	$78.53^{*}_{1.45}$	$85.78^{*}_{0.71}$	$79.44^{*}_{1.32}$	$82.08_{1.12}$	$80.35_{0.25}$
Comparison of Participant Judgments vs. PLM-AS Trained on Individual Data						
PLM-AS P1	$85.72^{*}_{0.97}$	$78.09^{*}_{0.66}$	$85.85^{*}_{0.92}$	$79.04^{*}_{0.62}$	$79.43_{1.05}$	$80.60_{0.42}$
P1 Judgement	$82.47^{*}_{1.84}$	$78.55^{*}_{2.47}$	$82.51^{*}_{1.82}$	$79.17^{*}_{2.35}$	$78.98_{2.86}$	$80.55_{0.36}$
PLM-AS P2	$87.65^{*}_{0.07}$	$83.05^{*}_{1.39}$	$87.72^{*}_{0.07}$	$83.51^{*}_{1.33}$	$81.17_{1.13}$	$80.33_{0.13}$
P2 Judgement	$83.67^{*}_{0.45}$	$77.12^{*}_{1.07}$	$83.78^{*}_{0.45}$	$78.17^{*}_{1.00}$	$81.64_{1.12}$	$81.11_{0.47}$
PLM-AS P3	$84.46^{*}_{0.79}$	$79.02^{*}_{0.04}$	$84.65^{*}_{0.80}$	$79.91^{*}_{0.05}$	$81.85_{0.94}$	$80.75_{0.08}$
P3 Judgement	$82.89^{*}_{0.66}$	$76.92^{*}_{1.31}$	$83.04^{*}_{0.66}$	$77.97^{*}_{1.04}$	$83.45_{1.11}$	$80.58_{0.15}$
PLM-AS P4	$88.68^{*}_{0.85}$	$84.13^{*}_{1.69}$	$88.72^{*}_{0.85}$	$84.51^{*}_{1.61}$	$82.68_{1.60}$	$80.60_{0.14}$
P4 Judgement	$86.71^{*}_{1.64}$	$81.94^{*}_{1.89}$	$86.78^{*}_{1.60}$	$82.51^{*}_{1.77}$	$82.83_{0.67}$	$80.98_{0.29}$
PLM-AS P5	$87.43^{*}_{1.24}$	$84.16^{*}_{1.73}$	$87.45^{*}_{1.24}$	$84.51^{*}_{1.68}$	$80.98_{1.65}$	$80.15_{0.04}$
P5 Judgement	$84.80^{*}_{0.60}$	$79.64^{*}_{0.66}$	$84.98^{*}_{0.59}$	$80.44^{*}_{0.62}$	$83.38_{0.96}$	$81.66_{0.07}$
PLM-AS P6	$86.46^{*}_{0.95}$	$79.58^{*}_{1.44}$	$86.58^{*}_{0.92}$	$80.38^{*}_{1.32}$	$83.41_{1.77}$	$80.56_{0.21}$
P6 Judgement	$80.77^{*}_{1.26}$	$70.77^{*}_{2.03}$	$81.18^{*}_{1.22}$	$72.96^{*}_{1.72}$	$82.76_{2.46}$	$80.51_{0.08}$
PLM-AS P7	$86.16^{*}_{0.55}$	$79.66^{*}_{0.68}$	$86.25^{*}_{0.54}$	$80.44^{*}_{0.64}$	$78.95_{0.21}$	$80.59_{0.24}$
P7 Judgement	$84.88^{*}_{0.33}$	$78.00^{*}_{0.99}$	$84.91^{*}_{0.35}$	$78.84^{*}_{0.93}$	$80.36_{1.51}$	$81.14_{0.01}$
Ranking Model-Generated Responses on the IMDB Dataset						
Random	$52.69_{1.30}$	$51.69_{0.65}$	$52.74_{1.31}$	$52.14_{0.67}$	$88.30_{0.43}$	$81.02_{0.06}$
SYNSP	$90.97^{*}_{0.36}$	$94.59^{*}_{0.58}$	$90.99^{*}_{0.35}$	$94.59^{*}_{0.57}$	$79.38_{0.76}$	$80.07_{0.27}$
DistilBERT	$93.38^{*}_{0.51}$	$93.85^{*}_{0.30}$	$93.39^{*}_{0.51}$	$93.86^{*}_{0.29}$	$82.09_{0.38}$	$80.73_{0.21}$

Table 6. Detailed results for all models, including accuracies and F1 scores for evaluation using DistilBERT and GPT-4o, as well as automated readability and diversity metrics. All results are the mean and standard error over three seeds. Significant improvements over the fine-tuned baseline (FT BL) are marked with * ($p < 0.01$).

C. On the Role of Sarcasm

As shown in Figure 3, sarcasm presents a major challenge, particularly to the DistilBERT classifier. To further investigate the role of sarcasm, we repeat the experiments from §5.1 and exclude sentences labeled as containing sarcasm from the dataset. This reduces the number of training instances from 497 to 329. The results in Table 7 show that excluding sarcastic instances improves the performance of models for controlled sentiment generation, particularly for all criteria using sentiment classifiers (DistilBERT, BERT, PLM-AS, SYSNP). The impact on criteria using the ground truth or participant annotations is less clear, indicating that the classifiers, in particular, struggle with sarcastic instances.

Model	F1$_{\text{Distilbert}}$	F1$_{\text{GPT-4o}}$	Accuracy$_{\text{Distilbert}}$	Accuracy$_{\text{GPT-4o}}$	Readability	Diversity
FT BL	53.12	52.48	53.11	53.31	87.83	81.06
Random	$52.39_{1.79}$	$50.82_{1.27}$	$52.47_{1.78}$	$51.27_{1.36}$	$88.45_{0.28}$	$80.95_{0.14}$
Ground Truth	$86.77_{0.22}$	$82.56_{1.35}$	$86.78_{0.22}$	$82.98_{1.29}$	$81.36_{2.14}$	$81.02_{0.14}$
DistilBERT (trained on IMDB)	$88.38_{1.13}$	$83.89_{0.92}$	$88.38_{1.14}$	$84.25_{0.86}$	$80.99_{1.30}$	$81.46_{0.17}$
Pure Eye-Tracking Criteria						
Mean Fixation Duration	$54.37_{2.05}$	$52.42_{0.81}$	$54.48_{2.06}$	$53.27_{1.23}$	$89.22_{0.35}$	$81.21_{0.16}$
Mean Number Refixations	$53.59_{1.48}$	$51.89_{0.51}$	$53.74_{1.50}$	$52.80_{0.74}$	$89.16_{0.27}$	$81.04_{0.07}$
Total Fixation Duration	$53.26_{1.51}$	$51.54_{0.99}$	$53.41_{1.48}$	$51.87_{0.90}$	$89.14_{0.35}$	$81.29_{0.17}$
Total Number Refixations	$52.54_{1.47}$	$50.92_{0.47}$	$52.60_{1.46}$	$51.27_{0.46}$	$88.75_{0.34}$	$81.18_{0.15}$
Classifier Trained on the Sentiment Split of ETSA-II (see Figure 2)						
BERT	$85.72_{0.83}$	$80.42_{0.06}$	$85.78_{0.83}$	$81.11_{0.05}$	$82.64_{0.42}$	$80.21_{0.27}$
PLM-AS	$87.00_{1.19}$	$80.38_{1.89}$	$87.05_{1.18}$	$81.11_{1.75}$	$79.52_{1.79}$	$80.32_{0.16}$
SYNSP	$87.30_{1.21}$	$81.84_{0.98}$	$87.32_{1.21}$	$82.31_{0.92}$	$79.42_{2.43}$	$80.42_{0.28}$
Comparison of Participant Judgments vs. PLM-AS Trained on Individual Data						
PLM-AS P1	$88.29_{0.67}$	$83.37_{1.69}$	$88.32_{0.67}$	$83.78_{1.62}$	$80.50_{0.67}$	$80.56_{0.21}$
P1 Judgement	$83.34_{1.27}$	$77.29_{0.53}$	$83.38_{1.28}$	$78.04_{0.62}$	$82.01_{1.76}$	$81.19_{0.21}$
PLM-AS P2	$87.31_{0.37}$	$83.91_{0.75}$	$87.32_{0.36}$	$84.25_{0.72}$	$80.47_{1.37}$	$80.08_{0.32}$
P2 Judgement	$85.08_{0.98}$	$80.80_{1.47}$	$85.11_{0.97}$	$81.38_{1.40}$	$82.59_{1.11}$	$80.33_{0.21}$
PLM-AS P3	$89.50_{0.35}$	$85.19_{0.91}$	$89.52_{0.35}$	$85.45_{0.89}$	$83.43_{1.96}$	$80.51_{0.28}$
P3 Judgement	$81.18_{0.56}$	$75.31_{0.72}$	$81.31_{0.58}$	$76.57_{0.62}$	$84.45_{0.32}$	$80.53_{0.11}$
PLM-AS P4	$88.63_{0.22}$	$83.43_{0.40}$	$88.65_{0.23}$	$83.84_{0.34}$	$80.41_{2.13}$	$80.45_{0.04}$
P4 Judgement	$87.92_{1.03}$	$83.11_{1.32}$	$87.92_{1.03}$	$83.45_{1.31}$	$82.98_{0.64}$	$80.64_{0.14}$
PLM-AS P5	$86.82_{0.99}$	$80.66_{2.33}$	$86.85_{0.99}$	$81.24_{2.17}$	$80.83_{0.93}$	$80.46_{0.41}$
P5 Judgement	$83.14_{0.64}$	$79.45_{0.39}$	$83.24_{0.59}$	$80.11_{0.33}$	$83.26_{1.10}$	$81.55_{0.13}$
PLM-AS P6	$88.96_{0.20}$	$84.08_{0.44}$	$88.99_{0.20}$	$84.45_{0.43}$	$83.52_{1.06}$	$80.49_{0.14}$
P6 Judgement	$85.02_{1.36}$	$77.30_{2.37}$	$85.11_{1.33}$	$78.37_{2.15}$	$83.26_{1.44}$	$81.01_{0.10}$
PLM-AS P7	$87.42_{0.47}$	$81.70_{1.53}$	$87.45_{0.46}$	$82.24_{1.42}$	$79.82_{2.17}$	$80.35_{0.18}$
P7 Judgement	$86.64_{0.29}$	$82.75_{1.42}$	$86.65_{0.28}$	$83.11_{1.37}$	$80.46_{1.46}$	$81.12_{0.17}$

Table 7. Accuracies, F1 score, text readability, and diversity metrics for models after excluding sarcastic sentences. The accuracies and F1 scores are obtained by comparing the tags against the scores on the model's generations produced by the evaluation model. The table shows the mean and standard errors over three different random seeds.

D. Fine-Tuning

We fine-tune the base model for 5 epochs on sentences from the ETSA-II HF split using a learning rate of $5e{-}5$ and a batch size of 2. As shown in Table 8 and Table 9, fine-tuning leads to a lower baseline performance (BL vs. FT BL) but increased performance after optimizing the model with DPO. Both tables show results when excluding sarcastic instances from the training data.

Model	F1$_{\text{Distilbert}}$	Acc$_{\text{Distilbert}}$	Readability	Diversity
BL	55.18	55.20	85.85	72.89
GT	$79.36_{0.98}$	$79.73_{0.96}$	$84.48_{1.03}$	$72.13_{0.34}$
PLM-AS	$78.72_{1.13}$	$79.00_{1.11}$	$84.63_{0.99}$	$71.42_{0.40}$
DistilBERT	$81.00_{1.36}$	$81.40_{1.29}$	$84.35_{0.62}$	$72.36_{0.36}$

Table 8. Results for models using the base model.

Model	F1$_{\text{Distilbert}}$	Acc$_{\text{Distilbert}}$	Readability	Diversity
FT BL	53.12	53.11	87.83	81.06
GT	$86.77_{0.22}$	$86.78_{0.22}$	$81.36_{2.14}$	$81.02_{0.14}$
PLM-AS	$87.00_{1.19}$	$87.05_{1.18}$	$79.52_{1.79}$	$80.32_{0.16}$
DistilBERT	$88.38_{1.13}$	$88.38_{1.14}$	$80.99_{1.30}$	$81.46_{0.17}$

Table 9. Results for models using the fine-tuned model.