SEEING EYE TO AI: HUMAN ALIGNMENT VIA GAZE-BASED RESPONSE REWARDS FOR LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Advancements in Natural Language Processing (NLP), have led to the emergence of Large Language Models (LLMs) such as GPT, Llama, Claude, and Gemini, which excel across a range of tasks but require extensive fine-tuning to align their outputs with human expectations. A widely used method for achieving this alignment is Reinforcement Learning from Human Feedback (RLHF), which, despite its success, faces challenges in accurately modelling human preferences. In this paper, we introduce GazeReward, a novel framework that integrates implicit feedback – and specifically eye-tracking (ET) data – into the Reward Model (RM). In addition, we explore how ET-based features can provide insights into user preferences. Through ablation studies we test our framework with different integration methods, LLMs, and ET generator models, demonstrating that our approach significantly improves the accuracy of the RM on established human preference datasets. This work advances the ongoing discussion on optimizing AI alignment with human values, exploring the potential of cognitive data for shaping future NLP research.

1 INTRODUCTION

Recent advancements in Natural Language Processing (NLP) have led to the emergence of Large Language Models (LLMs) like GPT (OpenAI, 2023), Llama (Dubey et al., 2024), Claude (Anthropic, 2024), and Gemini (Team et al., 2024), which excel across a range of tasks. These models, often consisting of billions of parameters, are trained on massive datasets and typically require extensive fine-tuning to align their outputs with human expectations [1]. Several works have focused on refining the way LLMs interpret and respond to user intent (Wang et al., 2023b), which has led to the development of novel alignment techniques. A common approach to achieving human alignment involves leveraging explicit human feedback as preference information. Currently, the most widely adopted method is Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022). RLHF has been implemented in many state-of-the-art LLMs (Cui et al., 2024; OpenAI, 2023; Bai et al., 2022b), and has been shown to help align models to human instructions and mitigate the generation of toxic or harmful content (Kiegeland et al., 2024). However, a persistent challenge with this approach is the difficulty of acquiring sufficient high-quality training data (Casper et al., 2023).

To be able to capture the complexities of real-world user instructions, there is a need for meticulously handcrafted data (Wang et al., 2023b), which are resource-expensive and difficult to scale (Yang et al., 2023). Obtaining high-quality feedback from human annotators, usually provided after examining a model response, suffers from several caveats (Casper et al., 2023). For instance, low inter-annotator agreement can result in inconsistent evaluations of the same model output due to varying interpretations, domain expertise, or biases. Moreover, "scalable oversight" – the ability to supervise models effectively with limited resources (Amodei et al., 2016) – remains an open problem. Inconsistent data quality is another issue, as cost-quality tradeoffs often arise when collecting human feedback.

[1] LLMs that are trained only on extensive datasets for language modeling are referred to as "pre-trained" LLMs. Subsequent approaches, such as human alignment, are categorized as "post-training".

To address these challenges, researchers have increasingly turned to LLMs as a form of AI-driven feedback, referred to as Reinforcement Learning from AI Feedback (RLAIF) Bai et al. (2022b). This method offers improved scalability, easier data collection, and cost-efficiency compared to traditional human-driven approaches (Bai et al., 2022b; Wang et al., 2023a; Madaan et al., 2023). However, it remains unclear what type of feedback signals, or a combination of feedback mechanisms, is optimal to align LLM with human goals (Casper et al., 2023). More research is needed to explore the underlying beliefs and expectations of human users (Casper et al., 2023), and how these can be incorporated into human alignment techniques. Furthermore, the alignment success of a language model is dependent on the quality of the underlying RM (Pace et al., 2024). Various alignment methods, such as RLHF, RLAIF, and Direct Preference Optimization (DPO) (Rafailov et al., 2023), rely on RM to incorporate feedback. Reward modelling is also essential for generating synthetic data for preference alignment and is often used in LLM inference to evaluate model outputs in techniques such as best-of-N sampling (Cui et al., 2024).

In this work, we propose a novel approach that incorporates Eye-tracking (ET) as an additional signal to address the challenge of human alignment. ET measures oculomotor behavior i.e. the movements and fixations of the eyes, which offers insight into visual attention and information processing (Kleinke, 1986; Land & Furneaux, 1997). This allows researchers to correlate observable eye movement patterns with underlying cognitive and perceptual processes during reading and language comprehension tasks (Kleinke, 1986; Krasich et al., 2018). Moreover, ET – unlike other (explicit) forms of feedback (e.g., questionnaire data, data annotation) – does not suffer from human biases, and offers a better temporal and spatial resolution (Zhang & Hollenstein, 2024). Several studies have shown a strong correlation between human eye movements and attention patterns in transformer-based models (Wang et al., 2024a; Bensemann et al., 2022; Sood et al., 2020a). Incorporating ET data into NLP tasks has also proven valuable, as demonstrated by numerous works (Huang et al., 2023; Khurana et al., 2023; Hollenstein et al., 2019; Yang & Hollenstein, 2023; Kiegeland et al., 2024; Deng et al., 2023a; Mathias et al., 2018; McGuire & Tomuro, 2021). Recently, Kiegeland et al. (2024) proposed the integration of ET in controlled sentiment generation to create a dataset that can be used in human alignment methods.

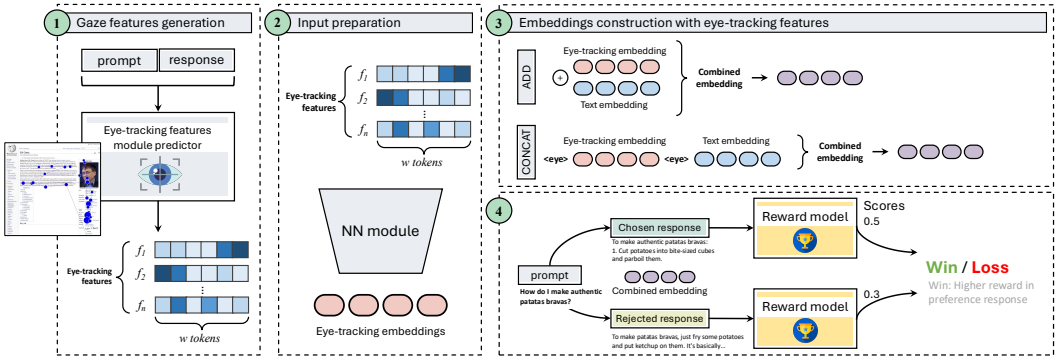


Figure 1: GazeReward Framework for using eye-tracking data for reward modelling. We use a generator model to compute ET features on a preference dataset D and we train the human preference by combining both text and ET embeddings (See section 4 for details.)

Human alignment continues to be one of the biggest challenges in the development of LLM, with RM playing an important role in addressing this issue. This paper investigates how behavioural signals, particularly ET, can be operationalised as implicit feedback to improve human alignment (the proposed approach is shown in Figure 1). Furthermore, we explore the use of ET prediction models that can generate – automatically and with little effort – ET features in response to text input, which makes our solution not only cost-effective but also highly scalable. The main contributions of our work can be summarized as follows:

- We propose **GazeReward**, a novel and scalable framework that integrates implicit feedback in the form of ET data into the RM, a key component in modeling human preferences.

- We perform for the first time an ablation study that examines several state-of-the-art LLMs, various ET prediction models, and different methods for incorporating ET features into the RM.
- We demonstrate experimentally substantial performance improvements with the GazeReward framework, showing accuracy gains of over 20% in RM predictions across diverse human preference datasets.

2 PRELIMINARIES

2.1 LARGE LANGUAGE MODELS-HUMAN ALIGNMENT

LLMs-Human Alignment typically involves training LLMs [2] on datasets curated by humans (learning from human feedback data) (Ouyang et al., 2022). This can be achieved through Supervised Fine-Tuning (SFT), where the model is trained on pairs of prompts (x) and corresponding human-generated responses (y) (Liu et al., 2024). Alternatively, alignment can be pursued via preference optimization, using a human preference dataset that differentiates between a better response (y_w) and a worse one (y_l) for the same prompt (x): $\mathcal{D} = \{(x^{(i)}, y_w^{(i)}, y_l^{(i)})\}_{i=1}^{N}$.

To this day, RLHF (Ouyang et al., 2022) remains the most popular technique used in state-of-the-art LLMs like GPT-4 (OpenAI, 2023), Claude (Bai et al., 2022b), Bard (Google, 2023), and Llama 2-Chat (Touvron et al., 2023). Different implementations of RLHF can vary in terms of data collection, training processes, and choice of RL algorithms. Typically, RLHF (Ouyang et al., 2022) involves three main steps: (1) collecting feedback, (2) training a RM based on that feedback, and (3) optimising the LLMs using RL techniques, such as Proximal Policy Optimization (PPO) Schulman et al. (2017). Since RLHF was first introduced, several advancements have been made, including fine-grained reward systems (Bai et al., 2022b; Wu et al., 2023b; Dong et al., 2023b; Wang et al., 2023c; 2024c), or replaced the original PPO algorithm with other RL techniques (Wu et al., 2023a).

An alternative to RLHF is DPO (Rafailov et al., 2023), which employs an offline RL approach to optimize language models based on preference data, without the need for a separate reward model. While DPO can be used independently, it is often complementary to other training methods such as SFT or statistical rejection sampling, to further improve human alignment based on a RM (Zhao et al., 2023; Liu et al., 2024; Dubey et al., 2024). Statistical rejection sampling, also called best-of-N or top-k-over-N (Bai et al., 2022b; Touvron et al., 2023; Dubey et al., 2024) is another widely used technique. Moreover, certain methods perform human alignment without RL to avoid instabilities, and fine-tune the model on filtered samples by a RM, or other sources (Dong et al., 2023a; Yuan et al., 2023).

A significant challenge in all human alignment techniques is data acquisition (Casper et al., 2023). This includes problems such as evaluator misalignment, supervision difficulties, and feedback quality issues (Casper et al., 2023). However, as AI systems continue to improve, LLMs are increasingly employed for tasks that were traditionally handled by humans, such as data annotation and generation. Unlike human feedback, AI-generated feedback offers better scalability, enabling faster and more cost-effective data collection. For example, RLAIF, introduced by Bai et al. (2022b), is a promising approach that trains reward models based on preferences generated by off-the-shelf LLMs. Variations of RLAIF have been explored in several studies (Lee et al., 2023; Jiao, 2023; Cui et al., 2024; Li et al., 2024; Yang et al., 2024). In the context of self-generating instructions, approaches like Self-Instruct (Wang et al., 2023a), Self-Refine (Madaan et al., 2023), and Self-Alignment (Li et al., 2023) demonstrate how models can autonomously generate datasets based on their learned human preferences.

Different alignment methods like RLHF and RLAIF rely on the RM to incorporate the human feedback. The RM learns to predict human preference based on labeled examples, serving as a proxy for human judgment later. Therefore, the success of language model alignment relies heavily on the quality of the underlying reward model (Pace et al., 2024), which in turn dictates the behaviour of the resultant chatbot (Shen et al., 2023). Even in LLM inference, methods like best-of-N sampling use the RM to evaluate model outputs (Cui et al., 2024). RM has also become crucial for generating

[2] Before the process of human alignment, these models are referred to as "pre-trained" LLMs.

Table 1: Eye-tracking (ET) features computed per word.

Feature	Definition
First Fixation Duration (FFD)	Time spent on the initial fixation
Go-Past Time (GPT)	Cumulative fixation time before moving to the right
Total Reading Time (TRT)	Overall time spent fixating on a word
Number of Fixations (nFix)	Number of fixations on each word
Proportion of participand (fixProp)	Proportion of participants that fixated on the word

synthetic data for preference alignment. In recent RLAIF methods, reward modeling has expanded beyond its traditional role and is now used to generate artificial feedback.

2.1.1 Reward modeling

In the original implementation (Ouyang et al., 2022), the goal of RM training is to train a classifier that predicts the probability of human preference p^* between two completions (Equation 1), modelled by a Bradley-Terry model (Bradley & Terry, 1952). The typical setup involves showing two completions, with preferences being measured using win-loss-tie outcomes or a Likert scale to capture the strength of preference (Bai et al., 2022a). The data is processed into prompt-chosen-rejected trios, where the chosen completion, y_w, is preferred over the rejected one, y_l, forming the basis for training (Ouyang et al., 2022).

$$p^*(y_w \succ y_l \mid x) = \frac{\exp(r^*(x, y_w))}{\exp(r^*(x, y_w)) + \exp(r^*(x, y_l))}. \tag{1}$$

2.2 Eye-Tracking

Eye-tracking (ET) systems monitor oculomotor behavior, such as eye movements and fixations, offering valuable insights into visual attention, information processing, and expands our understanding of reading and language comprehension. (Zhang & Hollenstein, 2024). Specifically, ET data often include fixations – pauses in eye movement to focus on specific areas (Mathias et al., 2022); saccades – rapid movements between two points (McGuire & Tomuro, 2021); scanpaths – sequences of fixations that reveal saccades and regressions (Yang & Hollenstein, 2023); and other temporal and spatial gaze behavior features (Zhang & Hollenstein, 2024). Incorporating ET data into NLP tasks often involves the use of several features listed in Table 1.

While several publicly available datasets such as ZUCO (Hollenstein et al., 2020b), ZUCO2 (Hollenstein et al., 2018), PROVO (Luke & Christianson, 2018), ETSA-I (Mishra et al., 2016), ETSA-II (Mishra et al., 2018), GECO (Cop et al., 2017), GECO-MT (Colman et al., 2022) are widely used in ET research, obtaining real ET data for NLP tasks remains a challenge. This is primarily due to the cost and precision requirements of ET equipment, the unavailability of gaze data during inference, as well as privacy concerns (Khurana et al., 2023). To address these challenges, two main approaches have been proposed. The first involves integrating ET data into the model during training through methods like Multi-task learning (MTL), which eliminates the need for ET data during inference (Mishra et al., 2018; Klerke et al., 2016; Ren & Xiong, 2023; Yu et al., 2024; Deng et al., 2024). The second approach involves techniques that directly predict users' gaze behaviour (Deng et al., 2024; 2023a; Zhang & Hollenstein, 2024; Wang et al., 2024a), creating synthetic ET data for any given text or stimulus (Deng et al., 2023b; Bolliger et al., 2023; Khurana et al., 2023; Li & Rudzicz, 2021; Hollenstein et al., 2021; 2022).

3 Related Work

Reward Modelling. The most popular approach to reward modeling follows the framework introduced by Ouyang et al. (2022). Several studies have examined alternative versions for refining RMs. For instance, Bai et al. (2022b) proposed more fine-grained reward structures, evaluating helpfulness and harmlessness separately. Other approaches have explored different reward modelling strategies (Wu et al., 2023b; Dong et al., 2023b; Wang et al., 2023c). Another line of research has focused

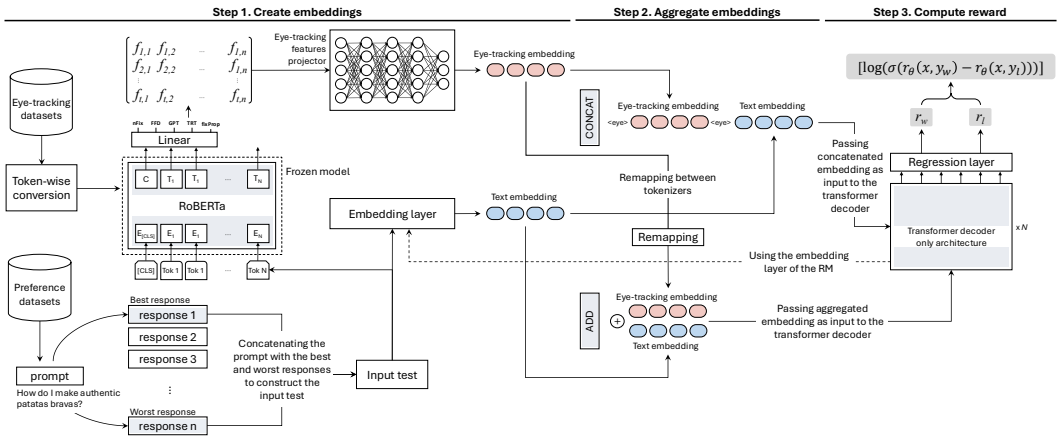


Figure 2: Overview of the **GazeReward** framework, incorporating eye-tracking features into the reward model. The architecture is illustrated in the figure using the second ET prediction model, but it would be identical if the first one were used instead (see subsection 5.1)

on Process Based Reward Models (PRMs) (Lightman et al., 2024; Uesato et al., 2022) which differ from conventional RMs by predicting the correctness of intermediate steps, rather than solely evaluating final outputs. Other studies implement data augmentation techniques (Shen et al., 2023), or cross-attention mechanisms between encoded input text and candidate pairs (Jiang et al., 2023b). Moreover, some works have leveraged synthetic preference data for reward modelling (Cui et al., 2024; Jiao, 2023). Wu et al. (2024b) built upon the LLM-as-a-Judge framework Zheng et al. (2023) by introducing LLM-as-a-Meta-Judge, which evaluates the model's judgments to generate preference pairs that enhance its decision-making capabilities. Finally, Pace et al. (2024) incorporated a self-training approach to improve reward model training. However, to date, no research has explored the integration of ET or other implicit feedback signals into RM.

Eye-tracking in Natural Language Processing. Several studies have investigated the use of ET data for a variety of NLP tasks, such as named entity recognition (Hollenstein & Zhang, 2019; Ren & Xiong, 2023; Yu et al., 2024; Hollenstein et al., 2019), text comprehension (Ahn et al., 2020; Reich et al., 2022; Sood et al., 2020b), language modelling (Huang et al., 2023; Huang & Hollenstein, 2023; Deng et al., 2023b), and question answering (Zhang & Hollenstein, 2024; Wang et al., 2024a). Other applications include code comprehension (Alakmeh et al., 2024), code summarization (Zhang et al., 2024) and hallucination detection (Maharaj et al., 2023). Eye-tracking has also been applied to sentiment analysis and sarcasm detection tasks (Mishra et al., 2016; 2017; 2018; Barrett et al., 2018; Huang et al., 2023; Khurana et al., 2023; Hollenstein et al., 2019; Yang & Hollenstein, 2023; Kiegeland et al., 2024; Deng et al., 2023a; Mathias et al., 2018; McGuire & Tomuro, 2021). The most relevant work to our approach is by Kiegeland et al. (2024), which introduced a dataset generation method using ET signals for DPO, building on the controlled sentiment generation framework proposed by Deng et al. (2023a); Yang & Hollenstein (2023). While this study has contributed to the first steps towards integrating ET for human alignment in LLMs, it is task- and dataset-specific, often relying on ranking criteria that underutilize the potential of ET feedback. In contrast, our approach presents a more general framework by directly incorporating implicit feedback into the RM, rather than limiting its application to dataset creation.

4 GazeReward: Reward modeling with ET Feedback

In this section, we discuss the proposed framework for augmenting the RM using implicit feedback derived from ET signals (Figure 2). Initially, we generate the ET features (subsection 4.1). For this step, we consider two state-of-the-art ET prediction models. Next, we combine the ET features with the text (subsection 4.2), producing different types of combined embeddings, and finally pass them as input into the RM to obtain the reward for the prompt and its corresponding response (subsection 4.3).

Table 2: Overview of different corpora used in the study to train the reward model.

Corpus	Train set	Val. set	Test set	Lang.	Reference
OASST1	6567	1160	416	EN*	Köpf et al. (2023)
HelpSteer2	5938	1049	364	EN	Wang et al. (2024c)

4.1 EYE-TRACKING FEATURES GENERATION

As discussed insubsection 2.2, obtaining organic ET features for NLP applications presents several challenges. In this work, we consider an approach inspired by RLAIF research, where feedback is artificially generated from pre-trained LLMs and, in particular, from ET prediction models. Specifically, we incorporate the output of two different ET prediction models (Li & Rudzicz, 2021; Huang & Hollenstein, 2023) and evaluate the impact of different set of features. As input to these models, we pass the same text as we do in the RM: a combination of prompt x and response y. The output is a set of ET features, denoted as f_{et}, for each token $f_{et} = \{f_1, f_2, \ldots, f_w\} \in \mathbb{R}^{w \times f}$, where w represents the number of tokens in the tokenizer used by the ET prediction model, and f is the number of features. Depending on the specific model, between one and five synthetic features $f_{et} = \{f_1, f_2, \ldots, f_w\} \in \mathbb{R}^{w \times f}$ are generated per token for the input text.

4.2 RM AUGMENTATION USING EYE-TRACKING FEATURES

We implement two different approaches for incorporating ET features into the RM, as shown in Figure 2. In the first approach, **GazeConcat**, we concatenate the ET embeddings with the text embeddings. In the second approach, **GazeAdd**, we add the ET embeddings to the text embeddings. Furthermore, we concatenate the prompt and the response to be evaluated and pass them through the pre-trained embedding layer of , to generate the embeddings $H = \{h_1, h_2, \ldots, h_n\} \in \mathbb{R}^{n \times d}$, where n is the number of tokens in the tokenizer used by the RM and d is the model embedding size.

To project these features to the model embedding size (d), we use a Multilayer Perceptron (MLP) ET feature projector $fp()$. The $fp()$ consists of two linear layers, two dropout layers, two Layer Normalization layers, and `ReLU` activation, designed for stable, non-linear ET feature representation and overfitting prevention. The model's input dimension dynamically adjusts to accommodate the number of features used during training. The ET features projector can be formulated as $emb_{ETF} = fp(f_{et}) \in \mathbb{R}^{w \times d}$ (Figure 2). This formula describes the projection of ETF features (f_{et}) through a function fp, resulting in an embedding matrix emb_{ETF} with dimensions $w \times d$, where w represents the number of time steps and d the embedding dimension.

GazeConcat: The ET embedding, denoted as emb_{ETF}, is concatenated with the text embedding H to form the input for the RM. To distinguish between the two modalities, we introduce two special tokens: *⟨eye⟩* and *⟨/eye⟩*, which flag the start and end of the ET embedding, respectively (Figure 2). These special tokens are randomly initialized as one-dimensional vectors and added to the embedding layer or the RM model for training. The final input is structured as: $(emb(<eye>) \circ emb_{ETF} \circ emb(</eye>) \circ H)$. The same process is applied to the attentions masks.

GazeAdd: The input to the RM consists of the ET embedding emb_{ETF} and the text embedding H, which are added in an elementwise fashion: $(emb_{ETF} + H)$. The two ET prediction models use different tokenizers, which also differ from those used by the base models in the RM. As a result, the number of tokens n in the input for the RM and the number of tokens w generated by the ET prediction model may not match. To address this embedding alignment issue, and have the same dimension, we remap the ET features from the w-token space to the n-token space used by each base model in the RM. Further implementation details can be found in Appendix A.1.3.

4.3 REWARD MODEL

The RM's architecture and hyperparameters are identical to those of the pretrained LLM, except that the classification head used for next-token prediction is replaced with a regression head that outputs a scalar reward (Touvron et al., 2023). This scalar reward indicates the quality of the model generation, corresponding to the predicted score for the final reply in a conversation. Differences in

these rewards represent the log-odds that one response is preferred over another. The loss function is defined in Equation 2, where y_w refers to the preferred response in a pair of completions y_w and y_l. The dataset D consists of human comparisons, where $r_\theta(x, y_w), r_\theta(x, y_l)$ represents the RM θ scalar outputs for the preferred and less preferred completions, respectively Ouyang et al. (2022).

$$\text{loss}(\theta) = -E_{(x,y_w,y_l)\sim D}\left[\log\left(\sigma\left(r_\theta(x, y_w) - r_\theta(x, y_l)\right)\right)\right] \tag{2}$$

In the proposed method, we augment the traditional RM, which uses text input (a combination of the prompt x and response y), by incorporating (artificial) implicit feedback through ET features generated from the same text. These ET features provide valuable information for capturing human preferences, thereby improving the RM's performance.

5 Experiments

5.1 Experimental setup

Datasets. For our experiments, we use the OpenAssistant Conversations dataset's (OASST1) (Köpf et al., 2023) and HelpSteer2 (Wang et al., 2024c) (Table 2). OASST1 is a human-generated, human-annotated, assistant-style conversation, created through global crowdsourcing and widely used for human alignment tasks (Köpf et al., 2023; Dettmers et al., 2023; Wu et al., 2024b). We filtered all non-English text, as the ET prediction models were exclusively trained on English data. Among the different responses in the dataset, we selected the two most distinct responses to compare the chosen and the rejected responses (Wang et al., 2024b). HelpSteer2 is a more recent, English-only dataset that has been used in studies such as Wang et al. (2024b;c). The dataset provides annotations for five response attributes: helpfulness, correctness, coherence, complexity, and verbosity. To transform it into a preference dataset, we designate the response with the higher helpfulness score as the chosen response and the other as the rejected response, following a method similar to that used in DPO training (Wang et al., 2024c) (see Appendix A.1.1 for more details about the datasets).

Dataset Preparation. To tune LLMs for human-AI interaction, we need to define a chat dialogue protocol that allows the model to understand human instructions and rate them. To this end, we adopt a chat protocol that utilizes special header and termination tokens, similar to the format used in Llama 3. For example, in the case of the Llama 3 8B model, the concatenation of a prompt and its corresponding response would follow this structure: *<im_start>user* Example Prompt *<im_end>* *<im_start>assistant* Example Response *<im_end>* (see Appendix A.1.1 for more details).

Models. As RM base models we use the pretrained checkpoint of Hugging Face (Appendix A.1.4) for Llama 3 8B, Llama 3 8B-instruct (Dubey et al., 2024) and Mistral 7B (Jiang et al., 2023a).

ET prediction models. In our analyses, we utilise two state-of-the-art ET prediction models, both pre-trained to predict ET features and kept frozen in our implementation. The input to these models is the same text used for the RM, with minimal modifications (Appendix A.1.2). The first model (Huang & Hollenstein, 2023), consists of a T5 embedding layer (Raffel et al., 2020), a two-layer BiLSTM (Hochreiter & Schmidhuber, 1997), and a one-hidden-layer MLP. This model was trained on the Dundee, GECO (Cop et al., 2017), ZuCo1 (Hollenstein et al., 2018), and ZuCo2 (Hollenstein et al., 2020a) datasets, and predicts total reading time (TRT) per token (Figure 3). The second model (Li & Rudzicz, 2021), is based on RoBERTa (Liu et al., 2019) with a regression head on each token. This head is a linear layer that outputs five features: FFD, fixProp, GPT, TRT, and nFix (Table 1). The model is initialized with pre-trained weights and fine-tuned on the ZuCo1 (Hollenstein et al., 2018), ZuCo2 (Hollenstein et al., 2020a) and PROVO (Luke & Christianson, 2018) datasets. Since RoBERTa's maximum sequence length is 512 tokens and our input sequences are longer, we employ a sliding window approach. The input is split into 512-token segments with a 50-token overlap, and the results are combined using a linear weighted approach. Further details on these models and their integration into our framework are provided in Appendix A.1.2.

Baseline models. To evaluate the improvement in accuracy for a RM that incorporates implicit feedback, and specifically ET signals, we compare the same RM with and without the ET embeddings. For each dataset and model, we train and evaluate all combinations of integrating and combining ET features, and then compare them against a RM trained on the same base model and dataset but without implicit feedback.

Table 3: Reward modeling accuracy (%) for OASST1 dataset. The highest results are in bold and the second highest are underlined.

		Llama-3-8B-Instruct		**Llama-3-8B**		**Mistral-7B**	
	baseline	64.6	diff(%)	58.6	diff(%)	62.0	diff(%)
GazeConcat	$fcomb_1$	68.5	6.1%	68.5	17.0%	63.9	3.1%
	$fcomb_{2.5}$	70.7	9.4%	70.4	20.3%	71.2	14.7%
	$fcomb_{2.2}$	68.5	6.1%	**73.1**	**24.8%**	**72.6**	**17.1%**
GazeAdd	$fcomb_1$	**70.9**	**9.8%**	71.2	21.5%	-	
	$fcomb_{2.5}$	51.9	-19.6%	54.3	-7.2%	-	
	$fcomb_{2.2}$	70.0	8.3%	69.0	17.8%	-	

Table 4: Reward modeling accuracy (%) for Helpsteer2 dataset. The highest results are in bold and the second highest are underlined.

		Llama-3-8B-Instruct		**Llama-3-8B**		**Mistral-7B**	
	baseline	54.3	diff(%)	53.2	diff(%)	51.25	diff(%)
GazeConcat	$fcomb_1$	62.8	15.7%	59.6	12.0%	**63.3**	**23.6%**
	$fcomb_{2.5}$	59.9	10.3%	53.3	0.2%	53.0	3.5%
	$fcomb_{2.2}$	59.3	9.2%	53.3	0.2%	57.1	11.5%
GazeAdd	$fcomb_1$	**63.2**	**16.3%**	**64.6**	**21.3%**	-	
	$fcomb_{2.5}$	59.9	10.3%	58.8	10.5%	-	
	$fcomb_{2.2}$	59.1	8.7%	59.6	12.1%	-	

Evaluation metrics. Performance is determined by measuring the model's ability to predict the better reponse from pairs of replies with different ranks. Accuracy is calculated as the percentage of cases where the reward score for the preferred response is higher than that of the less preferred response, based on a held-out dataset. This method follows similar approaches found in Touvron et al. (2023); Yuan et al. (2023); Köpf et al. (2023); Cui et al. (2024). We use the test split proposed by the authors for each dataset (Table 2). We also conduct a complementary evaluation on RewardBench (Lambert et al., 2024), a benchmark dataset created for evaluating performance and safety features of RM's (see Appendix B for more details).

Training procedure. In our implementetion, the ET prediction model remains frozen while the ET features projector is trained (Figure 2). For the RM, we fine-tune the open-source models previously introduced with QLoRA (Dettmers et al., 2023) a Parameter-Efficient Fine-Tuning (PEFT) method based on Low-Rank Adaptation (LoRA) (Hu et al., 2021), with other memory optimization techniques. We follow the training process for the RM as outlined in Touvron et al. (2023); Ouyang et al. (2022). We independently train each model on its respective dataset for two epochs, as detailed in Wang et al. (2024c). For hyperparameter tuning, we reserve 15% of each dataset for validation (shown in Table 2). The best-performing checkpoints are selected based on the lowest validation loss and used for performance evaluation. We perform a grid search to determine the optimal batch size and testing values of {8, 16, 32}. The AdamW optimizer (Loshchilov & Hutter, 2019) is used, with the Learning Rate (LR) is tuned over the range {1, 5, 10, 50}e-6, following the values reported in Touvron et al. (2023); Cui et al. (2024); Wang et al. (2024c). Additionally, we evaluate different LR schedulers: constant, linear, and cosine with a minimum LR. Further hyperparameter values and implementation details for both the RM and the ET projector can be found in Appendix C.

5.2 Results

The results of our experiments on the OASST and HelpSteer datasets, covering all possible combinations of ET features, models, and inclusion methods, as shown in Table 3 and Table 4 respectively. For the Mistral model, results for the **GazeAdd** method are unavailable due to the inability to map features between the ET prediction model's tokenizer and the reward model's tokenizer (details in Appendix A.1.3). In what follows, we discuss the main findings.

Effect of Model Initialization. We evaluate the impact of model initialization on performance. Open-access LLMs typically come in two forms: a pre-trained version without human alignment and a final version that has undergone alignment with human feedback. Since we lack access to intermediate checkpoints, we experiment with both pre-trained models (Mistral 7B and Llama 3) and models that are already human-aligned (Llama 3 Instruct). Our goal is to confirm that our method is effective for RM initialized with both pre-trained and human-aligned checkpoints. When comparing accuracy improvements relative to the baseline (without ET features), all models show considerable gains from incorporating implicit feedback. Notably, the Llama 3 8B and Mistral 7B models, which had no prior alignment, demonstrate over 20% performance improvement from the incorporation of ET features, indicating that unaligned models benefit more from implicit feedback.

Inclusion method. The results shown Table 3 and Table 4 indicate that both **GazeConcat** methods and **GazeAdd** introduce a substantial performance improvements to the RM. Across both datasets, concatenating embeddings (**GazeConcat**) delivers more consistent results. Incorporating ET information through specialized separator embeddings allows the model to process both text and ET features more robustly. However, in the HelpSteer dataset (Table 4), directly adding ET information to the text embeddings (**GazeAdd**) results in the greatest improvement over the baseline.

Eye-tracking (ET) feature importance. Different ET features capture distinct aspects of reading behaviour and information processing, influencing model performance uniquely (Zhang & Hollenstein, 2024). Here, we examine how model performance varies when incorporating three different feature combinations generated by two different ET prediction models: $fcomb_1$ – TRT generated by the first ET prediction model; $fcomb_{2.5}$ – five features (nFix, FFD, GPT, TRT, fixProp) generated by the second ET prediction model; and $fcomb_{2.2}$ – TRT and FFD generated by the second ET prediction model. TRT and FFD are widely used in ET research (Huang et al., 2023; Huang & Hollenstein, 2023; Zhang & Hollenstein, 2024; Maharaj et al., 2023; Wang et al., 2022), and they have been shown to correlate with attention scores from pre-trained transformer models (Wang et al., 2024a; Bensemann et al., 2022; Sood et al., 2020a). Other studies have used gradient-based saliency methods to explore similar correlations (Hollenstein & Beinborn, 2021; Wu et al., 2024a). When comparing results, we observe that the RM benefits from implicit feedback regardless of the ET feature combination or ET prediction model used. Specifically, in most cases, $fcomb_1$ yields the best results, particularly with the **GazeAdd** method. For **GazeConcat**, $fcomb_{2.2}$ performs best for the OASST1 dataset, while $fcomb_1$ excels in HelpSteer2. We attribute the superior performance of $fcomb_1$ to how the ET prediction model generating the fixations was trained, including the data and preprocessing methods used (see Appendix A.1.2). Moreover, when comparing $fcomb_{2.2}$ and $fcomb_{2.5}$ – both generated by the same model – only in one case does integrating nFix, GPT, and fixProp improves performance. In some instances, using $fcomb_{2.5}$ results in worse performance than the baseline, confirming findings provided by prior studies, which suggest that features related to reading time, such as FFD and TRT, contribute most to performance gains.

RewardBench. As a side contribution, we evaluate our best performing models (trained on the OAAST1 dataset) on RewardBench. This evaluation is not intended to directly compare our method with larger, more resource-intensive RM, but rather to show that through the integration of multi-modal signals like ET features we can significantly enhance the performance of RM models. The results shown in Table 5 demonstrate consistent improvements as previously observed, with gains exceeding 40% for the Mistral model – a notable gain considering that the base RM is the same. We note that the performance of the baseline models is impacted by RM trained on base models with less than 9B parameters and on relatively small datasets (see details in Appendix B).

6 Discussion

In this work, we introduced a novel framework for integrating implicit feedback into the Reward Model, a key component for aligning LLMs and generating synthetic data for further alignment. We validated our approach using widely-adopted, open-source models such as Llama 3 and Mistral, for initializing the RM. By employing two different models to generate ET features, our results show that incorporating implicit feedback consistently improves the RM's ability to predict user preferences, regardless of the model used and without the need to reach large parameter counts or train on massive datasets. Additionally, our method leverages ET features generated by models, making it fully scalable and applicable to various human alignment methods, including those that

Table 5: Reward modeling accuracy (%) evaluating on RewardBench dataset. All models are trained on OASST1 dataset. The highest results are in bold and the second highest are underlined.

		Llama-3-8B-Instruct		**Llama-3-8B**		**Mistral-7B**	
	baseline	46.9	diff(%)	50.9	diff(%)	41.2	diff(%)
GazeConcat	$fcomb_1$	57.8	23.1%	<u>58.4</u>	<u>14.5%</u>	59.9	45.4%
	$fcomb_{2.5}$	**58.4**	**24.4%**	58.1	14.1%	<u>60.3</u>	<u>46.4%</u>
	$fcomb_{2.2}$	<u>58.1</u>	<u>23.8%</u>	**58.5**	**14.8%**	**60.5**	**46.9%**
GazeAdd	$fcomb_1$	56.5	20.3%	56.6	11.2%	-	
	$fcomb_{2.5}$	54.9	16.9%	53.8	5.6%	-	
	$fcomb_{2.2}$	55.4	17.9%	52.5	3.1%	-	

involve artificially generated datasets. This work advances the ongoing discussion on optimizing AI alignment with human values and shows the potential of multimodal signals for NLP research enhancing current methods.

6.1 Limitations & Future Work

Data. A limitation of our study is that both ET prediction models were trained on a relatively small datasets (Appendix A.1.2) that are not tailored to our tasks. Future work could benefit from directly collecting ET data specifically for LLM-generated responses, to offer insights into human reading comprehension and information processing of prompts, which could further improve model performance. Additionally, since the ET prediction models used in our experiments were trained on English corpora, the method's generalizability to other languages requires further investigation. Moreover, we explored two methodologies for integrating ET features into the RM, but other approaches could prove more effective. For instance, ET features could be used to modify the RM's attention mask, as suggested by Zhang & Hollenstein (2024). Regarding dataset selection, both models used, Mistral 7B (Jiang et al., 2023a) and Llama 3 (Dubey et al., 2024), were fine-tuned on publicly available data, though specific details on the datasets are limited. Therefore, we cannot discount the possibility that the datasets we used may have been part of the models' pretraining, particularly for Llama 3 7B Instruct, which has already undergone human alignment. However, as we compare against baselines using the same model checkpoints, any potential effects would be consistent across both conditions. Finally, it is important to note that although the ET prediction models remain fixed during training, our solution has a slightly higher number of parameters than the baseline.

Training. The scaling trends for the RM (Touvron et al., 2023) show that larger models or models trained on massive datasets perform better. A promising direction would be to validate our framework on larger models, without relying on PEFT methods, and on larger datasets. We are confident that, despite the considerable computational costs this may entail, our framework is capable of scaling effectively. Another direction is integrating the proposed RM into an alignment method like RLHF, or applying it in rejection sampling to generate synthetic preference datasets, ensuring that accuracy gains in the RM translate to improvements in the final LLMs.

Reproducibility Statement

All the code necessary to reproduce this work will be released in a GitHub repository once published. Both datasets used are publicly available (Köpf et al., 2023; Wang et al., 2024c). Additionally, both ET prediction models have been trained with public datasets (Cop et al., 2017; Hollenstein et al., 2018; 2020a).

Impact Statement

Since our research uses only synthetic ET data, there are no privacy concerns or need for large-scale experiments involving human subjects. We should also raise attention to the limitations of human feedback and ET prediction models bias, that inevitably reflect aspects of their training data.

REFERENCES

Seoyoung Ahn, Conor Kelton, Aruna Balasubramanian, and Greg Zelinsky. Towards Predicting Reading Comprehension From Gaze Behavior. In *ACM Symposium on Eye Tracking Research and Applications*, ETRA '20 Short Papers, pp. 1–5, New York, NY, USA, June 2020. Association for Computing Machinery. ISBN 978-1-4503-7134-6. doi: 10.1145/3379156.3391335. URL https://dl.acm.org/doi/10.1145/3379156.3391335.

Tarek Alakmeh, David Reich, Lena Jäger, and Thomas Fritz. Predicting code comprehension: a novel approach to align human gaze with code using deep neural networks. In *Alakmeh, Tarek; Reich, David; Jäger, Lena; Fritz, Thomas (2024). Predicting code comprehension: a novel approach to align human gaze with code using deep neural networks. In: ACM International Conference on the Foundations of Software Engineering, Porto de Galinhas, Brazil, 17 July 2024 - 19 July 2024.*, Porto de Galinhas, Brazil, July 2024. University of Zurich. doi: 10.1145/3660795. URL https://www.zora.uzh.ch/id/eprint/260042/. Issue: 1 Number: 1.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete Problems in AI Safety, July 2016. URL http://arxiv.org/abs/1606.06565. arXiv:1606.06565 [cs].

Anthropic. Claude 3: Introducing the next generation of Claude, April 2024. URL https://www.anthropic.com/news/claude-3-family.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback, April 2022a. URL http://arxiv.org/abs/2204.05862. arXiv:2204.05862 [cs].

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional AI: Harmlessness from AI Feedback, December 2022b. URL http://arxiv.org/abs/2212.08073. arXiv:2212.08073 [cs].

Maria Barrett, Joachim Bingel, Nora Hollenstein, Marek Rei, and Anders Søgaard. Sequence Classification with Human Attention. In Anna Korhonen and Ivan Titov (eds.), *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pp. 302–312, Brussels, Belgium, October 2018. Association for Computational Linguistics. doi: 10.18653/v1/K18-1030. URL https://aclanthology.org/K18-1030.

Joshua Bensemann, Alex Peng, Diana Benavides-Prado, Yang Chen, Neset Tan, Paul Michael Corballis, Patricia Riddle, and Michael Witbrock. Eye Gaze and Self-attention: How Humans and Transformers Attend Words in Sentences. In Emmanuele Chersoni, Nora Hollenstein, Cassandra Jacobs, Yohei Oseki, Laurent Prévot, and Enrico Santus (eds.), *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pp. 75–87, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.cmcl-1.9. URL https://aclanthology.org/2022.cmcl-1.9.

Lena Bolliger, David Reich, Patrick Haller, Deborah Jakobi, Paul Prasse, and Lena Jäger. ScanDL: A Diffusion Model for Generating Synthetic Scanpaths on Texts. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 15513–15538, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.960. URL https://aclanthology.org/2023.emnlp-main.960.

Ralph Allan Bradley and Milton E. Terry. Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons. *Biometrika*, 39(3/4):324–345, 1952. ISSN 0006-3444. doi: 10.2307/2334029. URL https://www.jstor.org/stable/2334029. Publisher: [Oxford University Press, Biometrika Trust].

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Wang, Samuel Marks, Charbel-Raphaël 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 Bıyık, Anca Dragan, David Krueger, Dorsa Sadigh, and Dylan Hadfield-Menell. Open Problems and Fundamental Limitations of Reinforcement Learning from Human Feedback, July 2023. URL http://arxiv.org/abs/2307.15217. arXiv:2307.15217 [cs].

Toon Colman, Margot Fonteyne, Joke Daems, Nicolas Dirix, and Lieve Macken. GECO-MT: The Ghent Eye-tracking Corpus of Machine Translation. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Jan Odijk, and Stelios Piperidis (eds.), *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pp. 29–38, Marseille, France, June 2022. European Language Resources Association. URL https://aclanthology.org/2022.lrec-1.4.

Uschi Cop, Nicolas Dirix, Denis Drieghe, and Wouter Duyck. Presenting GECO: An eyetracking corpus of monolingual and bilingual sentence reading. *Behavior Research Methods*, 49(2):602–615, April 2017. ISSN 1554-3528. doi: 10.3758/s13428-016-0734-0. URL https://doi.org/10.3758/s13428-016-0734-0.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. UltraFeedback: Boosting Language Models with Scaled AI Feedback, July 2024. URL http://arxiv.org/abs/2310.01377. arXiv:2310.01377 [cs].

Shuwen Deng, Paul Prasse, David Reich, Tobias Scheffer, and Lena Jäger. Pre-Trained Language Models Augmented with Synthetic Scanpaths for Natural Language Understanding. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6500–6507, Singapore, December 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.400. URL https://aclanthology.org/2023.emnlp-main.400.

Shuwen Deng, David R. Reich, Paul Prasse, Patrick Haller, Tobias Scheffer, and Lena A. Jäger. Eyettention: An Attention-based Dual-Sequence Model for Predicting Human Scanpaths during Reading. *Proc. ACM Hum.-Comput. Interact.*, 7(ETRA), May 2023b. doi: 10.1145/3591131. URL https://doi.org/10.1145/3591131. Place: New York, NY, USA Publisher: Association for Computing Machinery.

Shuwen Deng, Paul Prasse, David Reich, Tobias Scheffer, and Lena Jäger. Fine-Tuning Pre-Trained Language Models with Gaze Supervision. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 217–224, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.acl-short.21.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient Finetuning of Quantized LLMs. November 2023. URL https://openreview.net/forum?id=OUIFPHEgJU&referrer=%5Bthe%20profile%20of%20Ari%20Holtzman%5D(%2Fprofile%3Fid%3D~Ari_Holtzman1).

Hanze Dong, Wei Xiong, Deepanshu Goyal, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. RAFT: Reward rAnked FineTuning for Generative Foundation Model Alignment, May 2023a. URL http://arxiv.org/abs/2304.06767. arXiv:2304.06767 [cs, stat].

Yi Dong, Zhilin Wang, Makesh Narsimhan Sreedhar, Xianchao Wu, and Oleksii Kuchaiev. SteerLM: Attribute Conditioned SFT as an (User-Steerable) Alternative to RLHF, October 2023b. URL http://arxiv.org/abs/2310.05344. arXiv:2310.05344 [cs].

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang,

Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vítor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. The Llama 3 Herd of Models, August 2024. URL http://arxiv.org/abs/2407.21783. arXiv:2407.21783 [cs].

Google. Google Bard - Herramienta de IA Generativa y Bot Conversacional, 2023. URL https://bard.google.com.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9 (8):1735–1780, November 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

Nora Hollenstein and Lisa Beinborn. Relative Importance in Sentence Processing. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pp. 141–150, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-short.19. URL https://aclanthology.org/2021.acl-short.19.

Nora Hollenstein and Ce Zhang. Entity Recognition at First Sight: Improving NER with Eye Movement Information. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 1–10, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1001. URL https://aclanthology.org/N19-1001.

Nora Hollenstein, Jonathan Rotsztejn, Marius Troendle, Andreas Pedroni, Ce Zhang, and Nicolas Langer. ZuCo, a simultaneous EEG and eye-tracking resource for natural sentence reading. *Scientific Data*, 5(1):180291, December 2018. ISSN 2052-4463. doi: 10.1038/sdata.2018.291. URL

https://www.nature.com/articles/sdata2018291. Publisher: Nature Publishing Group.

Nora Hollenstein, Maria Barrett, Marius Troendle, Francesco Bigiolli, Nicolas Langer, and Ce Zhang. Advancing NLP with Cognitive Language Processing Signals, April 2019. URL http://arxiv.org/abs/1904.02682. arXiv:1904.02682 [cs].

Nora Hollenstein, Maria Barrett, and Lisa Beinborn. Towards Best Practices for Leveraging Human Language Processing Signals for Natural Language Processing. In Emmanuele Chersoni, Barry Devereux, and Chu-Ren Huang (eds.), *Proceedings of the Second Workshop on Linguistic and Neurocognitive Resources*, pp. 15–27, Marseille, France, May 2020a. European Language Resources Association. ISBN 979-10-95546-52-8. URL https://aclanthology.org/2020.lincr-1.3.

Nora Hollenstein, Marius Troendle, Ce Zhang, and Nicolas Langer. ZuCo 2.0: A Dataset of Physiological Recordings During Natural Reading and Annotation. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis (eds.), *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pp. 138–146, Marseille, France, May 2020b. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.18.

Nora Hollenstein, Emmanuele Chersoni, Cassandra L. Jacobs, Yohei Oseki, Laurent Prévot, and Enrico Santus. CMCL 2021 Shared Task on Eye-Tracking Prediction. In Emmanuele Chersoni, Nora Hollenstein, Cassandra Jacobs, Yohei Oseki, Laurent Prévot, and Enrico Santus (eds.), *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pp. 72–78, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.cmcl-1.7. URL https://aclanthology.org/2021.cmcl-1.7.

Nora Hollenstein, Emmanuele Chersoni, Cassandra Jacobs, Yohei Oseki, Laurent Prévot, and Enrico Santus. CMCL 2022 Shared Task on Multilingual and Crosslingual Prediction of Human Reading Behavior. In Emmanuele Chersoni, Nora Hollenstein, Cassandra Jacobs, Yohei Oseki, Laurent Prévot, and Enrico Santus (eds.), *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pp. 121–129, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.cmcl-1.14. URL https://aclanthology.org/2022.cmcl-1.14.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models, October 2021. URL http://arxiv.org/abs/2106.09685. arXiv:2106.09685 [cs].

Xinting Huang and Nora Hollenstein. Long-Range Language Modeling with Selective Cache. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 4838–4858, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.321. URL https://aclanthology.org/2023.findings-emnlp.321.

Xinting Huang, Jiajing Wan, Ioannis Kritikos, and Nora Hollenstein. Longer Fixations, More Computation: Gaze-Guided Recurrent Neural Networks, October 2023. URL http://arxiv.org/abs/2311.00159. arXiv:2311.00159 [cs].

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7B, October 2023a. URL http://arxiv.org/abs/2310.06825. arXiv:2310.06825 [cs].

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. LLM-Blender: Ensembling Large Language Models with Pairwise Ranking and Generative Fusion, June 2023b. URL http://arxiv.org/abs/2306.02561. arXiv:2306.02561 [cs].

Tianhao Wu * Hanlin Zhu and Jiantao Banghua Zhu * Jiao, Evan Frick *. Starling-7B: Increasing LLM Helpfulness & Harmlessness with RLAIF, November 2023. URL https://starling.cs.berkeley.edu.

Alan Kennedy, Joël Pynte, Wayne Murray, and Shirley-Anne Paul. Frequency and predictability effects in the Dundee Corpus: An eye movement analysis. *Quarterly journal of experimental psychology (2006)*, 66, March 2012. doi: 10.1080/17470218.2012.676054.

Varun Khurana, Yaman Kumar, Nora Hollenstein, Rajesh Kumar, and Balaji Krishnamurthy. Synthesizing Human Gaze Feedback for Improved NLP Performance. pp. 1895–1908, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.139. URL https://aclanthology.org/2023.eacl-main.139.

Samuel Kiegeland, David Robert Reich, Ryan Cotterell, Lena Ann Jäger, and Ethan Wilcox. The Pupil Becomes the Master: Eye-Tracking Feedback for Tuning LLMs. July 2024. URL https://openreview.net/forum?id=8oLUcBgKua.

Chris L. Kleinke. Gaze and eye contact: A research review. *Psychological Bulletin*, 100(1):78–100, 1986. ISSN 1939-1455. doi: 10.1037/0033-2909.100.1.78. Place: US Publisher: American Psychological Association.

Sigrid Klerke, Yoav Goldberg, and Anders Søgaard. Improving sentence compression by learning to predict gaze. In Kevin Knight, Ani Nenkova, and Owen Rambow (eds.), *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1528–1533, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1179. URL https://aclanthology.org/N16-1179.

Kristina Krasich, Robert McManus, Stephen Hutt, Myrthe Faber, Sidney K. D'Mello, and James R. Brockmole. Gaze-based signatures of mind wandering during real-world scene processing. *Journal of Experimental Psychology. General*, 147(8):1111–1124, August 2018. ISSN 1939-2222. doi: 10.1037/xge0000411.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi-Rui Tam, Keith Stevens, Abdullah Barhoum, Nguyen Minh Duc, Oliver Stanley, Richárd Nagyfi, Shahul ES, Sameer Suri, David Glushkov, Arnav Dantuluri, Andrew Maguire, Christoph Schuhmann, Huu Nguyen, and Alexander Mattick. OpenAssistant Conversations – Democratizing Large Language Model Alignment, April 2023. URL http://arxiv.org/abs/2304.07327. arXiv:2304.07327 [cs].

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, L. J. Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. RewardBench: Evaluating Reward Models for Language Modeling, March 2024. URL http://arxiv.org/abs/2403.13787. arXiv:2403.13787 [cs].

M. F. Land and S. Furneaux. The knowledge base of the oculomotor system. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 352(1358):1231–1239, August 1997. ISSN 0962-8436. doi: 10.1098/rstb.1997.0105.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Lu, Thomas Mesnard, Colton Bishop, Victor Carbune, and Abhinav Rastogi. RLAIF: Scaling Reinforcement Learning from Human Feedback with AI Feedback, September 2023. URL http://arxiv.org/abs/2309.00267. arXiv:2309.00267 [cs].

Ang Li, Qiugen Xiao, Peng Cao, Jian Tang, Yi Yuan, Zijie Zhao, Xiaoyuan Chen, Liang Zhang, Xiangyang Li, Kaitong Yang, Weidong Guo, Yukang Gan, Xu Yu, Daniell Wang, and Ying Shan. HRLAIF: Improvements in Helpfulness and Harmlessness in Open-domain Reinforcement Learning From AI Feedback, March 2024. URL http://arxiv.org/abs/2403.08309. arXiv:2403.08309 [cs].

Bai Li and Frank Rudzicz. TorontoCL at CMCL 2021 Shared Task: RoBERTa with Multi-Stage Fine-Tuning for Eye-Tracking Prediction. In Emmanuele Chersoni, Nora Hollenstein, Cassandra Jacobs, Yohei Oseki, Laurent Prévot, and Enrico Santus (eds.), *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pp. 85–89, Online, June 2021.

Association for Computational Linguistics. doi: 10.18653/v1/2021.cmcl-1.9. URL https://aclanthology.org/2021.cmcl-1.9.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Luke Zettlemoyer, Omer Levy, Jason Weston, and Mike Lewis. Self-Alignment with Instruction Backtranslation, August 2023. URL http://arxiv.org/abs/2308.06259. arXiv:2308.06259 [cs].

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. LET'S VERIFY STEP BY STEP. 2024.

Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J. Liu, and Jialu Liu. Statistical Rejection Sampling Improves Preference Optimization, January 2024. URL http://arxiv.org/abs/2309.06657. arXiv:2309.06657 [cs].

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach, July 2019. URL http://arxiv.org/abs/1907.11692. arXiv:1907.11692 [cs].

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization, January 2019. URL http://arxiv.org/abs/1711.05101. arXiv:1711.05101 [cs, math].

Steven G. Luke and Kiel Christianson. The Provo Corpus: A large eye-tracking corpus with predictability norms. *Behavior Research Methods*, 50(2):826–833, April 2018. ISSN 1554-3528. doi: 10.3758/s13428-017-0908-4. URL https://doi.org/10.3758/s13428-017-0908-4.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-Refine: Iterative Refinement with Self-Feedback, May 2023. URL http://arxiv.org/abs/2303.17651. arXiv:2303.17651 [cs].

Kishan Maharaj, Ashita Saxena, Raja Kumar, Abhijit Mishra, and Pushpak Bhattacharyya. Eyes Show the Way: Modelling Gaze Behaviour for Hallucination Detection. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 11424–11438, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.764. URL https://aclanthology.org/2023.findings-emnlp.764.

Sandeep Mathias, Diptesh Kanojia, Kevin Patel, Samarth Agrawal, Abhijit Mishra, and Pushpak Bhattacharyya. Eyes are the Windows to the Soul: Predicting the Rating of Text Quality Using Gaze Behaviour. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2352–2362, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1219. URL https://aclanthology.org/P18-1219.

Sandeep Mathias, Diptesh Kanojia, Abhijit Mishra, and Pushpak Bhattacharyya. A Survey on Using Gaze Behaviour for Natural Language Processing, January 2022. URL http://arxiv.org/abs/2112.15471. arXiv:2112.15471 [cs].

Erik S. McGuire and Noriko Tomuro. Sentiment Analysis with Cognitive Attention Supervision. *Proceedings of the Canadian Conference on Artificial Intelligence*, June 2021. doi: 10.21428/594757db.90170c50. URL https://caiac.pubpub.org/pub/kzxft4i8.

Abhijit Mishra, Diptesh Kanojia, and Pushpak Bhattacharyya. Predicting Readers' Sarcasm Understandability by Modeling Gaze Behavior. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1), March 2016. ISSN 2374-3468. doi: 10.1609/aaai.v30i1.9884. URL https://ojs.aaai.org/index.php/AAAI/article/view/9884. Number: 1.

Abhijit Mishra, Diptesh Kanojia, Seema Nagar, Kuntal Dey, and Pushpak Bhattacharyya. Leveraging Cognitive Features for Sentiment Analysis, January 2017. URL http://arxiv.org/abs/1701.05581. arXiv:1701.05581 [cs].

Abhijit Mishra, Srikanth Tamilselvam, Riddhiman Dasgupta, Seema Nagar, and Kuntal Dey. Cognition-Cognizant Sentiment Analysis With Multitask Subjectivity Summarization Based on Annotators' Gaze Behavior. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, April 2018. doi: 10.1609/aaai.v32i1.12068. URL https://ojs.aaai.org/index.php/AAAI/article/view/12068. ISSN: 2374-3468, 2159-5399 Issue: 1 Journal Abbreviation: AAAI.

OpenAI. GPT-4 Technical Report, March 2023. URL http://arxiv.org/abs/2303.08774. arXiv:2303.08774 [cs].

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, March 2022. URL http://arxiv.org/abs/2203.02155. arXiv:2203.02155 [cs].

Alizée Pace, Jonathan Mallinson, Eric Malmi, Sebastian Krause, and Aliaksei Severyn. West-of-N: Synthetic Preference Generation for Improved Reward Modeling, January 2024. URL http://arxiv.org/abs/2401.12086. arXiv:2401.12086 [cs].

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct Preference Optimization: Your Language Model is Secretly a Reward Model, December 2023. URL http://arxiv.org/abs/2305.18290. arXiv:2305.18290 [cs].

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, July 2020. URL http://arxiv.org/abs/1910.10683. arXiv:1910.10683 [cs, stat].

David Robert Reich, Paul Prasse, Chiara Tschirner, Patrick Haller, Frank Goldhammer, and Lena A. Jäger. Inferring Native and Non-Native Human Reading Comprehension and Subjective Text Difficulty from Scanpaths in Reading. In *2022 Symposium on Eye Tracking Research and Applications*, ETRA '22, pp. 1–8, New York, NY, USA, June 2022. Association for Computing Machinery. ISBN 978-1-4503-9252-5. doi: 10.1145/3517031.3529639. URL https://doi.org/10.1145/3517031.3529639.

Yuqi Ren and Deyi Xiong. CogAlign: Learning to Align Textual Neural Representations to Cognitive Language Processing Signals, November 2023. URL http://arxiv.org/abs/2106.05544. arXiv:2106.05544 [cs].

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms, August 2017. URL http://arxiv.org/abs/1707.06347. arXiv:1707.06347 [cs].

Lingfeng Shen, Sihao Chen, Linfeng Song, Lifeng Jin, Baolin Peng, Haitao Mi, Daniel Khashabi, and Dong Yu. The Trickle-down Impact of Reward Inconsistency on RLHF. October 2023. URL https://openreview.net/forum?id=MeHmwCDifc.

Ekta Sood, Simon Tannert, Diego Frassinelli, Andreas Bulling, and Ngoc Thang Vu. Interpreting Attention Models with Human Visual Attention in Machine Reading Comprehension. In Raquel Fernández and Tal Linzen (eds.), *Proceedings of the 24th Conference on Computational Natural Language Learning*, pp. 12–25, Online, November 2020a. Association for Computational Linguistics. doi: 10.18653/v1/2020.conll-1.2. URL https://aclanthology.org/2020.conll-1.2.

Ekta Sood, Simon Tannert, Philipp Mueller, and Andreas Bulling. Improving Natural Language Processing Tasks with Human Gaze-Guided Neural Attention. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6327–6341. Curran Associates, Inc., 2020b. URL https://proceedings.neurips.cc/paper/2020/hash/460191c72f67e90150a093b4585e7eb4-Abstract.html.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, Soroosh Mariooryad, Yifan Ding, Xinyang Geng, Fred Alcober, Roy Frostig, Mark Omernick, Lexi Walker, Cosmin Paduraru, Christina Sorokin, Andrea Tacchetti, Colin Gaffney, Samira Daruki, Olcan Sercinoglu, Zach Gleicher, Juliette Love, Paul Voigtlaender, Rohan Jain, Gabriela Surita, Kareem Mohamed, Rory Blevins, Junwhan Ahn, Tao Zhu, Kornraphop Kawintiranon, Orhan Firat, Yiming Gu, Yujing Zhang, Matthew Rahtz, Manaal Faruqui, Natalie Clay, Justin Gilmer, J. D. Co-Reyes, Ivo Penchev, Rui Zhu, Nobuyuki Morioka, Kevin Hui, Krishna Haridasan, Victor Campos, Mahdis Mahdieh, Mandy Guo, Samer Hassan, Kevin Kilgour, Arpi Vezer, Heng-Tze Cheng, Raoul de Liedekerke, Siddharth Goyal, Paul Barham, D. J. Strouse, Seb Noury, Jonas Adler, Mukund Sundararajan, Sharad Vikram, Dmitry Lepikhin, Michela Paganini, Xavier Garcia, Fan Yang, Dasha Valter, Maja Trebacz, Kiran Vodrahalli, Chulayuth Asawaroengchai, Roman Ring, Norbert Kalb, Livio Baldini Soares, Siddhartha Brahma, David Steiner, Tianhe Yu, Fabian Mentzer, Antoine He, Lucas Gonzalez, Bibo Xu, Raphael Lopez Kaufman, Laurent El Shafey, Junhyuk Oh, Tom Hennigan, George van den Driessche, Seth Odoom, Mario Lucic, Becca Roelofs, Sid Lall, Amit Marathe, Betty Chan, Santiago Ontanon, Luheng He, Denis Teplyashin, Jonathan Lai, Phil Crone, Bogdan Damoc, Lewis Ho, Sebastian Riedel, Karel Lenc, Chih-Kuan Yeh, Aakanksha Chowdhery, Yang Xu, Mehran Kazemi, Ehsan Amid, Anastasia Petrushkina, Kevin Swersky, Ali Khodaei, Gowoon Chen, Chris Larkin, Mario Pinto, Geng Yan, Adria Puigdomenech Badia, Piyush Patil, Steven Hansen, Dave Orr, Sebastien M. R. Arnold, Jordan Grimstad, Andrew Dai, Sholto Douglas, Rishika Sinha, Vikas Yadav, Xi Chen, Elena Gribovskaya, Jacob Austin, Jeffrey Zhao, Kaushal Patel, Paul Komarek, Sophia Austin, Sebastian Borgeaud, Linda Friso, Abhimanyu Goyal, Ben Caine, Kris Cao, Da-Woon Chung, Matthew Lamm, Gabe Barth-Maron, Thais Kagohara, Kate Olszewska, Mia Chen, Kaushik Shivakumar, Rishabh Agarwal, Harshal Godhia, Ravi Rajwar, Javier Snaider, Xerxes Dotiwalla, Yuan Liu, Aditya Barua, Victor Ungureanu, Yuan Zhang, Bat-Orgil Batsaikhan, Mateo Wirth, James Qin, Ivo Danihelka, Tulsee Doshi, Martin Chadwick, Jilin Chen, Sanil Jain, Quoc Le, Arjun Kar, Madhu Gurumurthy, Cheng Li, Ruoxin Sang, Fangyu Liu, Lampros Lamprou, Rich Munoz, Nathan Lintz, Harsh Mehta, Heidi Howard, Malcolm Reynolds, Lora Aroyo, Quan Wang, Lorenzo Blanco, Albin Cassirer, Jordan Griffith, Dipanjan Das, Stephan Lee, Jakub Sygnowski, Zach Fisher, James Besley, Richard Powell, Zafarali Ahmed, Dominik Paulus, David Reitter, Zalan Borsos, Rishabh Joshi, Aedan Pope, Steven Hand, Vittorio Selo, Vihan Jain, Nikhil Sethi, Megha Goel, Takaki Makino, Rhys May, Zhen Yang, Johan Schalkwyk, Christina Butterfield, Anja Hauth, Alex Goldin, Will Hawkins, Evan Senter, Sergey Brin, Oliver Woodman, Marvin Ritter, Eric Noland, Minh Giang, Vijay Bolina, Lisa Lee, Tim Blyth, Ian Mackinnon, Machel Reid, Obaid Sarvana, David Silver, Alexander Chen, Lily Wang, Loren Maggiore, Oscar Chang, Nithya Attaluri, Gregory Thornton, Chung-Cheng Chiu, Oskar Bunyan, Nir Levine, Timothy Chung, Evgenii Eltyshev, Xiance Si, Timothy Lillicrap, Demetra Brady, Vaibhav Aggarwal, Boxi Wu, Yuanzhong Xu, Ross McIlroy, Kartikeya Badola, Paramjit Sandhu, Erica Moreira, Wojciech Stokowiec, Ross Hemsley, Dong Li, Alex Tudor, Pranav Shyam, Elahe Rahimtoroghi, Salem Haykal, Pablo Sprechmann, Xiang Zhou, Diana Mincu, Yujia Li, Ravi Addanki, Kalpesh Krishna, Xiao Wu, Alexandre Frechette, Matan Eyal, Allan Dafoe, Dave Lacey, Jay Whang, Thi Avrahami, Ye Zhang, Emanuel Taropa, Hanzhao Lin, Daniel Toyama, Eliza Rutherford, Motoki Sano, HyunJeong Choe, Alex Tomala, Chalence Safranek-Shrader, Nora Kassner, Mantas Pajarskas, Matt Harvey, Sean Sechrist, Meire Fortunato, Christina Lyu, Gamaleldin Elsayed, Chenkai Kuang, James Lottes, Eric Chu, Chao Jia, Chih-Wei Chen, Peter Humphreys, Kate Baumli, Connie Tao, Rajkumar Samuel, Cicero Nogueira dos Santos, Anders Andreassen, Nemanja Rakićević, Dominik Grewe, Aviral Kumar, Stephanie Winkler, Jonathan Caton, Andrew Brock, Sid Dalmia, Hannah Sheahan, Iain Barr, Yingjie Miao, Paul Natsev, Jacob Devlin, Feryal Behbahani, Flavien Prost, Yanhua Sun, Artiom Myaskovsky, Thanumalayan Sankaranarayana Pillai, Dan Hurt, Angeliki Lazaridou, Xi Xiong, Ce Zheng, Fabio Pardo, Xiaowei Li, Dan Horgan, Joe Stanton, Moran Ambar, Fei Xia, Alejandro Lince, Mingqiu Wang, Basil Mustafa, Albert Webson, Hyo Lee, Rohan Anil, Martin Wicke, Timothy Dozat, Abhishek Sinha, Enrique Piqueras, Elahe Dabir, Shyam Upadhyay, Anudhyan Boral, Lisa Anne Hendricks, Corey Fry, Josip Djolonga, Yi Su, Jake Walker, Jane Labanowski, Ronny Huang, Vedant Misra, Jeremy Chen, R. J. Skerry-Ryan, Avi Singh, Shruti Rijhwani, Dian Yu, Alex Castro-Ros, Beer Changpinyo, Romina Datta, Sumit Bagri, Arnar Mar Hrafnkelsson, Marcello Maggioni, Daniel Zheng, Yury Sulsky, Shaobo Hou, Tom Le Paine, Antoine Yang, Jason Riesa, Dominika Rogozinska, Dror Marcus, Dalia El Badawy, Qiao Zhang, Luyu Wang, Helen Miller, Jeremy Greer, Lars Lowe Sjos, Azade Nova, Heiga Zen, Rahma Chaabouni, Mihaela Rosca, Jiepu Jiang, Charlie Chen, Ruibo Liu, Tara

Sainath, Maxim Krikun, Alex Polozov, Jean-Baptiste Lespiau, Josh Newlan, Zeyncep Cankara, Soo Kwak, Yunhan Xu, Phil Chen, Andy Coenen, Clemens Meyer, Katerina Tsihlas, Ada Ma, Juraj Gottweis, Jinwei Xing, Chenjie Gu, Jin Miao, Christian Frank, Zeynep Cankara, Sanjay Ganapathy, Ishita Dasgupta, Steph Hughes-Fitt, Heng Chen, David Reid, Keran Rong, Hongmin Fan, Joost van Amersfoort, Vincent Zhuang, Aaron Cohen, Shixiang Shane Gu, Anhad Mohananey, Anastasija Ilic, Taylor Tobin, John Wieting, Anna Bortsova, Phoebe Thacker, Emma Wang, Emily Caveness, Justin Chiu, Eren Sezener, Alex Kaskasoli, Steven Baker, Katie Millican, Mohamed Elhawaty, Kostas Aisopos, Carl Lebsack, Nathan Byrd, Hanjun Dai, Wenhao Jia, Matthew Wiethoff, Elnaz Davoodi, Albert Weston, Lakshman Yagati, Arun Ahuja, Isabel Gao, Golan Pundak, Susan Zhang, Michael Azzam, Khe Chai Sim, Sergi Caelles, James Keeling, Abhanshu Sharma, Andy Swing, YaGuang Li, Chenxi Liu, Carrie Grimes Bostock, Yamini Bansal, Zachary Nado, Ankesh Anand, Josh Lipschultz, Abhijit Karmarkar, Lev Proleev, Abe Ittycheriah, Soheil Hassas Yeganeh, George Polovets, Aleksandra Faust, Jiao Sun, Alban Rrustemi, Pen Li, Rakesh Shivanna, Jeremiah Liu, Chris Welty, Federico Lebron, Anirudh Baddepudi, Sebastian Krause, Emilio Parisotto, Radu Soricut, Zheng Xu, Dawn Bloxwich, Melvin Johnson, Behnam Neyshabur, Justin Mao-Jones, Renshen Wang, Vinay Ramasesh, Zaheer Abbas, Arthur Guez, Constant Segal, Duc Dung Nguyen, James Svensson, Le Hou, Sarah York, Kieran Milan, Sophie Bridgers, Wiktor Gworek, Marco Tagliasacchi, James Lee-Thorp, Michael Chang, Alexey Guseynov, Ale Jakse Hartman, Michael Kwong, Ruizhe Zhao, Sheleem Kashem, Elizabeth Cole, Antoine Miech, Richard Tanburn, Mary Phuong, Filip Pavetic, Sebastien Cevey, Ramona Comanescu, Richard Ives, Sherry Yang, Cosmo Du, Bo Li, Zizhao Zhang, Mariko Iinuma, Clara Huiyi Hu, Aurko Roy, Shaan Bijwadia, Zhenkai Zhu, Danilo Martins, Rachel Saputro, Anita Gergely, Steven Zheng, Dawei Jia, Ioannis Antonoglou, Adam Sadovsky, Shane Gu, Yingying Bi, Alek Andreev, Sina Samangooei, Mina Khan, Tomas Kocisky, Angelos Filos, Chintu Kumar, Colton Bishop, Adams Yu, Sarah Hodkinson, Sid Mittal, Premal Shah, Alexandre Moufarek, Yong Cheng, Adam Bloniarz, Jaehoon Lee, Pedram Pejman, Paul Michel, Stephen Spencer, Vladimir Feinberg, Xuehan Xiong, Nikolay Savinov, Charlotte Smith, Siamak Shakeri, Dustin Tran, Mary Chesus, Bernd Bohnet, George Tucker, Tamara von Glehn, Carrie Muir, Yiran Mao, Hideto Kazawa, Ambrose Slone, Kedar Soparkar, Disha Shrivastava, James Cobon-Kerr, Michael Sharman, Jay Pavagadhi, Carlos Araya, Karolis Misiunas, Nimesh Ghelani, Michael Laskin, David Barker, Qiujia Li, Anton Briukhov, Neil Houlsby, Mia Glaese, Balaji Lakshminarayanan, Nathan Schucher, Yunhao Tang, Eli Collins, Hyeontaek Lim, Fangxiaoyu Feng, Adria Recasens, Guangda Lai, Alberto Magni, Nicola De Cao, Aditya Siddhant, Zoe Ashwood, Jordi Orbay, Mostafa Dehghani, Jenny Brennan, Yifan He, Kelvin Xu, Yang Gao, Carl Saroufim, James Molloy, Xinyi Wu, Seb Arnold, Solomon Chang, Julian Schrittwieser, Elena Buchatskaya, Soroush Radpour, Martin Polacek, Skye Giordano, Ankur Bapna, Simon Tokumine, Vincent Hellendoorn, Thibault Sottiaux, Sarah Cogan, Aliaksei Severyn, Mohammad Saleh, Shantanu Thakoor, Laurent Shefey, Siyuan Qiao, Meenu Gaba, Shuo-yiin Chang, Craig Swanson, Biao Zhang, Benjamin Lee, Paul Kishan Rubenstein, Gan Song, Tom Kwiatkowski, Anna Koop, Ajay Kannan, David Kao, Parker Schuh, Axel Stjerngren, Golnaz Ghiasi, Gena Gibson, Luke Vilnis, Ye Yuan, Felipe Tiengo Ferreira, Aishwarya Kamath, Ted Klimenko, Ken Franko, Kefan Xiao, Indro Bhattacharya, Miteyan Patel, Rui Wang, Alex Morris, Robin Strudel, Vivek Sharma, Peter Choy, Sayed Hadi Hashemi, Jessica Landon, Mara Finkelstein, Priya Jhakra, Justin Frye, Megan Barnes, Matthew Mauger, Dennis Daun, Khuslen Baatarsukh, Matthew Tung, Wael Farhan, Henryk Michalewski, Fabio Viola, Felix de Chaumont Quitry, Charline Le Lan, Tom Hudson, Qingze Wang, Felix Fischer, Ivy Zheng, Elspeth White, Anca Dragan, Jean-baptiste Alayrac, Eric Ni, Alexander Pritzel, Adam Iwanicki, Michael Isard, Anna Bulanova, Lukas Zilka, Ethan Dyer, Devendra Sachan, Srivatsan Srinivasan, Hannah Muckenhirn, Honglong Cai, Amol Mandhane, Mukarram Tariq, Jack W. Rae, Gary Wang, Kareem Ayoub, Nicholas FitzGerald, Yao Zhao, Woohyun Han, Chris Alberti, Dan Garrette, Kashyap Krishnakumar, Mai Gimenez, Anselm Levskaya, Daniel Sohn, Josip Matak, Inaki Iturrate, Michael B. Chang, Jackie Xiang, Yuan Cao, Nishant Ranka, Geoff Brown, Adrian Hutter, Vahab Mirrokni, Nanxin Chen, Kaisheng Yao, Zoltan Egyed, Francois Galilee, Tyler Liechty, Praveen Kallakuri, Evan Palmer, Sanjay Ghemawat, Jasmine Liu, David Tao, Chloe Thornton, Tim Green, Mimi Jasarevic, Sharon Lin, Victor Cotruta, Yi-Xuan Tan, Noah Fiedel, Hongkun Yu, Ed Chi, Alexander Neitz, Jens Heitkaemper, Anu Sinha, Denny Zhou, Yi Sun, Charbel Kaed, Brice Hulse, Swaroop Mishra, Maria Georgaki, Sneha Kudugunta, Clement Farabet, Izhak Shafran, Daniel Vlasic, Anton Tsitsulin, Rajagopal Ananthanarayanan, Alen Carin, Guolong Su, Pei Sun, Shashank V, Gabriel Carvajal, Josef Broder, Iulia Comsa, Alena Repina, William Wong, Warren Weilun Chen, Peter Hawkins, Egor Filonov, Lucia Loher, Christoph

Hirnschall, Weiyi Wang, Jingchen Ye, Andrea Burns, Hardie Cate, Diana Gage Wright, Federico Piccinini, Lei Zhang, Chu-Cheng Lin, Ionel Gog, Yana Kulizhskaya, Ashwin Sreevatsa, Shuang Song, Luis C. Cobo, Anand Iyer, Chetan Tekur, Guillermo Garrido, Zhuyun Xiao, Rupert Kemp, Huaixiu Steven Zheng, Hui Li, Ananth Agarwal, Christel Ngani, Kati Goshvadi, Rebeca Santamaria-Fernandez, Wojciech Fica, Xinyun Chen, Chris Gorgolewski, Sean Sun, Roopal Garg, Xinyu Ye, S. M. Ali Eslami, Nan Hua, Jon Simon, Pratik Joshi, Yelin Kim, Ian Tenney, Sahitya Potluri, Lam Nguyen Thiet, Quan Yuan, Florian Luisier, Alexandra Chronopoulou, Salvatore Scellato, Praveen Srinivasan, Minmin Chen, Vinod Koverkathu, Valentin Dalibard, Yaming Xu, Brennan Saeta, Keith Anderson, Thibault Sellam, Nick Fernando, Fantine Huot, Junehyuk Jung, Mani Varadarajan, Michael Quinn, Amit Raul, Maigo Le, Ruslan Habalov, Jon Clark, Komal Jalan, Kalesha Bullard, Achintya Singhal, Thang Luong, Boyu Wang, Sujeevan Rajayogam, Julian Eisenschlos, Johnson Jia, Daniel Finchelstein, Alex Yakubovich, Daniel Balle, Michael Fink, Sameer Agarwal, Jing Li, Dj Dvijotham, Shalini Pal, Kai Kang, Jaclyn Konzelmann, Jennifer Beattie, Olivier Dousse, Diane Wu, Remi Crocker, Chen Elkind, Siddhartha Reddy Jonnalagadda, Jong Lee, Dan Holtmann-Rice, Krystal Kallarackal, Rosanne Liu, Denis Vnukov, Neera Vats, Luca Invernizzi, Mohsen Jafari, Huanjie Zhou, Lilly Taylor, Jennifer Prendki, Marcus Wu, Tom Eccles, Tianqi Liu, Kavya Kopparapu, Francoise Beaufays, Christof Angermueller, Andreea Marzoca, Shourya Sarcar, Hilal Dib, Jeff Stanway, Frank Perbet, Nejc Trdin, Rachel Sterneck, Andrey Khorlin, Dinghua Li, Xihui Wu, Sonam Goenka, David Madras, Sasha Goldshtein, Willi Gierke, Tong Zhou, Yaxin Liu, Yannie Liang, Anais White, Yunjie Li, Shreya Singh, Sanaz Bahargam, Mark Epstein, Sujoy Basu, Li Lao, Adnan Ozturel, Carl Crous, Alex Zhai, Han Lu, Zora Tung, Neeraj Gaur, Alanna Walton, Lucas Dixon, Ming Zhang, Amir Globerson, Grant Uy, Andrew Bolt, Olivia Wiles, Milad Nasr, Ilia Shumailov, Marco Selvi, Francesco Piccinno, Ricardo Aguilar, Sara McCarthy, Misha Khalman, Mrinal Shukla, Vlado Galic, John Carpenter, Kevin Villela, Haibin Zhang, Harry Richardson, James Martens, Matko Bosnjak, Shreyas Rammohan Belle, Jeff Seibert, Mahmoud Alnahlawi, Brian McWilliams, Sankalp Singh, Annie Louis, Wen Ding, Dan Popovici, Lenin Simicich, Laura Knight, Pulkit Mehta, Nishesh Gupta, Chongyang Shi, Saaber Fatehi, Jovana Mitrovic, Alex Grills, Joseph Pagadora, Dessie Petrova, Danielle Eisenbud, Zhishuai Zhang, Damion Yates, Bhavishya Mittal, Nilesh Tripuraneni, Yannis Assael, Thomas Brovelli, Prateek Jain, Mihajlo Velimirovic, Canfer Akbulut, Jiaqi Mu, Wolfgang Macherey, Ravin Kumar, Jun Xu, Haroon Qureshi, Gheorghe Comanici, Jeremy Wiesner, Zhitao Gong, Anton Ruddock, Matthias Bauer, Nick Felt, Anirudh GP, Anurag Arnab, Dustin Zelle, Jonas Rothfuss, Bill Rosgen, Ashish Shenoy, Bryan Seybold, Xinjian Li, Jayaram Mudigonda, Goker Erdogan, Jiawei Xia, Jiri Simsa, Andrea Michi, Yi Yao, Christopher Yew, Steven Kan, Isaac Caswell, Carey Radebaugh, Andre Elisseeff, Pedro Valenzuela, Kay McKinney, Kim Paterson, Albert Cui, Eri Latorre-Chimoto, Solomon Kim, William Zeng, Ken Durden, Priya Ponnapalli, Tiberiu Sosea, Christopher A. Choquette-Choo, James Manyika, Brona Robenek, Harsha Vashisht, Sebastien Pereira, Hoi Lam, Marko Velic, Denese Owusu-Afriyie, Katherine Lee, Tolga Bolukbasi, Alicia Parrish, Shawn Lu, Jane Park, Balaji Venkatraman, Alice Talbert, Lambert Rosique, Yuchung Cheng, Andrei Sozanschi, Adam Paszke, Praveen Kumar, Jessica Austin, Lu Li, Khalid Salama, Wooyeol Kim, Nandita Dukkipati, Anthony Baryshnikov, Christos Kaplanis, XiangHai Sheng, Yuri Chervonyi, Caglar Unlu, Diego de Las Casas, Harry Askham, Kathryn Tunyasuvunakool, Felix Gimeno, Siim Poder, Chester Kwak, Matt Miecnikowski, Vahab Mirrokni, Alek Dimitriev, Aaron Parisi, Dangyi Liu, Tomy Tsai, Toby Shevlane, Christina Kouridi, Drew Garmon, Adrian Goedeckemeyer, Adam R. Brown, Anitha Vijayakumar, Ali Elqursh, Sadegh Jazayeri, Jin Huang, Sara Mc Carthy, Jay Hoover, Lucy Kim, Sandeep Kumar, Wei Chen, Courtney Biles, Garrett Bingham, Evan Rosen, Lisa Wang, Qijun Tan, David Engel, Francesco Pongetti, Dario de Cesare, Dongseong Hwang, Lily Yu, Jennifer Pullman, Srini Narayanan, Kyle Levin, Siddharth Gopal, Megan Li, Asaf Aharoni, Trieu Trinh, Jessica Lo, Norman Casagrande, Roopali Vij, Loic Matthey, Bramandia Ramadhana, Austin Matthews, C. J. Carey, Matthew Johnson, Kremena Goranova, Rohin Shah, Shereen Ashraf, Kingshuk Dasgupta, Rasmus Larsen, Yicheng Wang, Manish Reddy Vuyyuru, Chong Jiang, Joana Ijazi, Kazuki Osawa, Celine Smith, Ramya Sree Boppana, Taylan Bilal, Yuma Koizumi, Ying Xu, Yasemin Altun, Nir Shabat, Ben Bariach, Alex Korchemniy, Kiam Choo, Olaf Ronneberger, Chimezie Iwuanyanwu, Shubin Zhao, David Soergel, Cho-Jui Hsieh, Irene Cai, Shariq Iqbal, Martin Sundermeyer, Zhe Chen, Elie Bursztein, Chaitanya Malaviya, Fadi Biadsy, Prakash Shroff, Inderjit Dhillon, Tejasi Latkar, Chris Dyer, Hannah Forbes, Massimo Nicosia, Vitaly Nikolaev, Somer Greene, Marin Georgiev, Pidong Wang, Nina Martin, Hanie Sedghi, John Zhang, Praseem Banzal, Doug Fritz, Vikram Rao, Xuezhi Wang, Jiageng Zhang, Viorica Patraucean, Dayou Du, Igor

Mordatch, Ivan Jurin, Lewis Liu, Ayush Dubey, Abhi Mohan, Janek Nowakowski, Vlad-Doru Ion, Nan Wei, Reiko Tojo, Maria Abi Raad, Drew A. Hudson, Vaishakh Keshava, Shubham Agrawal, Kevin Ramirez, Zhichun Wu, Hoang Nguyen, Ji Liu, Madhavi Sewak, Bryce Petrini, DongHyun Choi, Ivan Philips, Ziyue Wang, Ioana Bica, Ankush Garg, Jarek Wilkiewicz, Priyanka Agrawal, Xiaowei Li, Danhao Guo, Emily Xue, Naseer Shaik, Andrew Leach, Sadh MNM Khan, Julia Wiesinger, Sammy Jerome, Abhishek Chakladar, Alek Wenjiao Wang, Tina Ornduff, Folake Abu, Alireza Ghaffarkhah, Marcus Wainwright, Mario Cortes, Frederick Liu, Joshua Maynez, Andreas Terzis, Pouya Samangouei, Riham Mansour, Tomasz Kepa, François-Xavier Aubet, Anton Algymr, Dan Banica, Agoston Weisz, Andras Orban, Alexandre Senges, Ewa Andrejczuk, Mark Geller, Niccolo Dal Santo, Valentin Anklin, Majd Al Merey, Martin Baeuml, Trevor Strohman, Junwen Bai, Slav Petrov, Yonghui Wu, Demis Hassabis, Koray Kavukcuoglu, Jeffrey Dean, and Oriol Vinyals. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context, August 2024. URL http://arxiv.org/abs/2403.05530. arXiv:2403.05530 [cs].

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open Foundation and Fine-Tuned Chat Models, July 2023. URL http://arxiv.org/abs/2307.09288. arXiv:2307.09288 [cs].

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback, November 2022. URL http://arxiv.org/abs/2211.14275. arXiv:2211.14275 [cs].

Bingbing Wang, Bin Liang, Jiachen Du, Min Yang, and Ruifeng Xu. SEMGraph: Incorporating Sentiment Knowledge and Eye Movement into Graph Model for Sentiment Analysis. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 7521–7531, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.510. URL https://aclanthology.org/2022.emnlp-main.510.

Bingbing Wang, Bin Liang, Lanjun Zhou, and Ruifeng Xu. Gaze-infused BERT: Do human gaze signals help pre-trained language models? *Neural Computing and Applications*, April 2024a. ISSN 0941-0643, 1433-3058. doi: 10.1007/s00521-024-09725-8. URL https://link.springer.com/10.1007/s00521-024-09725-8.

Tianlu Wang, Ilia Kulikov, Olga Golovneva, Ping Yu, Weizhe Yuan, Jane Dwivedi-Yu, Richard Yuanzhe Pang, Maryam Fazel-Zarandi, Jason Weston, and Xian Li. Self-Taught Evaluators, August 2024b. URL http://arxiv.org/abs/2408.02666. arXiv:2408.02666 [cs].

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-Instruct: Aligning Language Models with Self-Generated Instructions, May 2023a. URL http://arxiv.org/abs/2212.10560. arXiv:2212.10560 [cs].

Yufei Wang, Wanjun Zhong, Liangyou Li, Fei Mi, Xingshan Zeng, Wenyong Huang, Lifeng Shang, Xin Jiang, and Qun Liu. Aligning Large Language Models with Human: A Survey, July 2023b. URL http://arxiv.org/abs/2307.12966. arXiv:2307.12966 [cs].

Zhilin Wang, Yi Dong, Jiaqi Zeng, Virginia Adams, Makesh Narsimhan Sreedhar, Daniel Egert, Olivier Delalleau, Jane Polak Scowcroft, Neel Kant, Aidan Swope, and Oleksii Kuchaiev.

HelpSteer: Multi-attribute Helpfulness Dataset for SteerLM, November 2023c. URL http://arxiv.org/abs/2311.09528. arXiv:2311.09528 [cs].

Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy J. Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. HelpSteer2: Open-source dataset for training top-performing reward models, June 2024c. URL http://arxiv.org/abs/2406.08673. arXiv:2406.08673 [cs].

Guojun Wu, Lena Bolliger, David Reich, and Lena Jäger. An Eye Opener Regarding Task-Based Text Gradient Saliency. In Tatsuki Kuribayashi, Giulia Rambelli, Ece Takmaz, Philipp Wicke, and Yohei Oseki (eds.), *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pp. 255–263, Bangkok, Thailand, August 2024a. Association for Computational Linguistics. URL https://aclanthology.org/2024.cmcl-1.22.

Tianhao Wu, Banghua Zhu, Ruoyu Zhang, Zhaojin Wen, Kannan Ramchandran, and Jiantao Jiao. Pairwise Proximal Policy Optimization: Harnessing Relative Feedback for LLM Alignment, October 2023a. URL http://arxiv.org/abs/2310.00212. arXiv:2310.00212 [cs].

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason Weston, and Sainbayar Sukhbaatar. Meta-Rewarding Language Models: Self-Improving Alignment with LLM-as-a-Meta-Judge, July 2024b. URL http://arxiv.org/abs/2407.19594. arXiv:2407.19594 [cs].

Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A. Smith, Mari Ostendorf, and Hannaneh Hajishirzi. Fine-Grained Human Feedback Gives Better Rewards for Language Model Training, October 2023b. URL http://arxiv.org/abs/2306.01693. arXiv:2306.01693 [cs].

Duo Yang and Nora Hollenstein. PLM-AS: Pre-trained Language Models Augmented with Scanpaths for Sentiment Classification. *Proceedings of the Northern Lights Deep Learning Workshop*, 4, January 2023. ISSN 2703-6928. doi: 10.7557/18.6797. URL https://septentrio.uit.no/index.php/nldl/article/view/6797.

Jingfeng Yang, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Bing Yin, and Xia Hu. Harnessing the Power of LLMs in Practice: A Survey on ChatGPT and Beyond, April 2023. URL http://arxiv.org/abs/2304.13712. arXiv:2304.13712 [cs].

Kevin Yang, Dan Klein, Asli Celikyilmaz, Nanyun Peng, and Yuandong Tian. RLCD: Reinforcement Learning from Contrastive Distillation for Language Model Alignment, March 2024. URL http://arxiv.org/abs/2307.12950. arXiv:2307.12950 [cs].

Jie Yu, Wenya Kong, and Fangfang Liu. CeER: A Nested Name Entity Recognition Model Incorporating Gaze Feature. In Wenjie Zhang, Anthony Tung, Zhonglong Zheng, Zhengyi Yang, Xiaoyang Wang, and Hongjie Guo (eds.), *Web and Big Data*, pp. 32–45, Singapore, 2024. Springer Nature Singapore. ISBN 978-981-9772-32-2.

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. RRHF: Rank Responses to Align Language Models with Human Feedback without tears, May 2023. URL http://arxiv.org/abs/2304.05302. arXiv:2304.05302 [cs].

Leran Zhang and Nora Hollenstein. Eye-Tracking Features Masking Transformer Attention in Question-Answering Tasks. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue (eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 7057–7070, Torino, Italy, May 2024. ELRA and ICCL. URL https://aclanthology.org/2024.lrec-main.619.

Yifan Zhang, Jiliang Li, Zachary Karas, Aakash Bansal, Toby Jia-Jun Li, Collin McMillan, Kevin Leach, and Yu Huang. EyeTrans: Merging Human and Machine Attention for Neural Code Summarization, February 2024. URL http://arxiv.org/abs/2402.14096. arXiv:2402.14096 [cs].

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J. Liu. SLiC-HF: Sequence Likelihood Calibration with Human Feedback, May 2023. URL http://arxiv.org/abs/2305.10425. arXiv:2305.10425 [cs].

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena, December 2023. URL http://arxiv.org/abs/2306.05685. arXiv:2306.05685 [cs].

A APPENDIX

A.1 IMPLEMENTATION DETAILS

This section provides further details on the implementation of our method. Subsection A.1.1 provides more details on the datasets used and their preprocessing steps. In subsection A.1.2, further information is given about the models used for generating ET features, along with the specific preprocessing required for each. Subsection A.1.3 explains the process of mapping the fixations from the tokenizer used by the generation model to the tokenizer used by the Reward Model. Subsection A.1.4 give more detials on the checkpoints used fot the RM backbone models. Finally, additional implementation details are discussed in subsection A.1.5.

A.1.1 DATASET PROCESSING

In this subsection, we provide more details about the datasets used and the preprocessing to train the RM. We use two different datasets: OpenAssistant Conversations dataset's (OASST1) [3] (Köpf et al., 2023) and HelpSteer2 [4] (Wang et al., 2024c).

OASST1. A human-generated, human-annotated assistant-style conversation corpus consisting of 161,443 messages in 35 different languages, resulting in over 10,000 complete and fully annotated conversation trees. The basic data structure is a Conversation Tree (CT), with nodes representing written messages in a conversation. A CT's root node represents an initial prompt, given by the prompter. The data was collected using a web-app interface as a product of a worldwide crowdsourcing effort involving over 13,500 volunteers, dividing the collection of each tree into five separate steps: prompting, labelling prompts, adding reply messages as prompter or assistant, labelling replies, and ranking assistant replies.

HelpSteer2. A CC-BY-4.0-licensed open-source helpfulness dataset, designed to train state-of-the-art RM consisting on 10,000 response pairs. It collects prompts mainly from ShareGPT [5], focusing on user inputs and filtering out non-English and programming-related prompts for quality. The prompts are clustered into topics and sampled based on complexity to ensure diversity. Multi-turn prompts are generated using an in-house model, with responses sourced from various internal models and human annotators. For each response, they annotate five attributes (helpfulness, correctness, coherence, complexity, and verbosity) on a Likert-5 scale involving multiple annotators for each response, ensuring high-quality ratings across five attributes.

Conversation format and dataset preparation.
To fine-tune LLMs for human-AI interaction, we need to define a chat protocol. We use a multi-message chat setup with a special header and termination tokens, similar to the one in Llama 3 Dubey et al. (2024). The header tokens differentiate the turns between the user and the system. For this, we use the *apply_chat_template*[6] feature from *FastTokenizers* in the *transformers* library.

The tokenizer used by the Meta-Llama-3-8B-Instruct model already incorporates this chat format since this model has already undergone human alignment. Therefore, we use this format in our experiments. For the other two models, we employ the default chat format provided by their respective tokenizers. We add new tokens in the embeddings layer for these chat formats and we train them as part of our process. Below, we provide an example of the template for each model.

[3] https://huggingface.co/datasets/OpenAssistant/oasst1
[4] https://huggingface.co/datasets/nvidia/HelpSteer2
[5] https://huggingface.co/datasets/RyokoAI/ShareGPT52K
[6] https://huggingface.co/docs/transformers/main/en/chat_templating

- **Meta-Llama-3-8B-Instruct:** *<begin_of_text><start_header_id>user<end_header_id>* Example Prompt *<eot_id><start_header_id>assistant<end_header_id>* Example Response *<eot_id>*
- **Meta-Llama-3-8B:** *<im_start>user* Example Prompt *<im_end>* *<im_start>assistant* Example Response *<im_end>*
- **Mistral-7B:** *<s>[INST]* Example Prompt *[/INST]* Example Response *</s>*

A.1.2 EYE-TRACKER FEATURES GENERATION MODELS

Special tokens are removed from the text before it is tokenized with the corresponding tokenizer used for the ET generator model. This is done to ensure that special tokens related to the chat format are not included in the input and are not assigned ET features to them, since these tokens are just for the RM to understand the chat format.

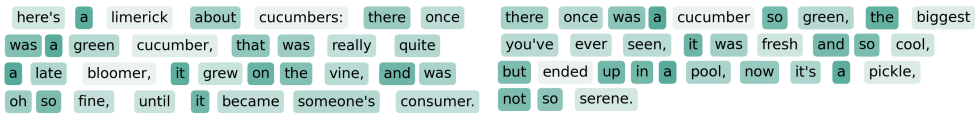
First model: Model presented in Huang & Hollenstein (2023). The code for the model along with the weights is publicly available, so we used the pre-trained checkpoint and we adapted their code for our implementation. This model was trained on several eye-tracking datasets, including Dundee (Kennedy et al., 2012), GECO (Cop et al., 2017), ZuCo1 (Hollenstein et al., 2018), ZuCo2 (Hollenstein et al., 2020a). More detailed information about this datasets is presented in Table 7. The best model achieves an mean squared error (MSE) of 4.02 on a randomly held-out test set (25% of all data). This model has 17.5M pararemetes and remains frozen during training. In Figure 3 we show an example of the synthetic total reading time (TRT) generated for the chosen and rejected response to a prompt.

For training this model, since the fixation duration is distributed differently across corpora, the authors normalize the fixation duration for each corpus, by dividing it by the mean duration of the corpus. Moreover, they map the duration values to discrete space $[1, 2, , K]$. Using K-quantiles, the fixation values were partitioned into K subsets of nearly equal sizes, and each value was assigned to the index of the corresponding subset. The model is then trained in a multi-task setting, computing the mean and variance of the fixation duration. This quartile-based processing is used in other works (Huang et al., 2023) that use ET data to improve performance in NLP tasks, and we believe it is part of the reason why we obtained better results with this combination when training the RM. The authors proposed a method specifically for converting word-level TRT to token-level fixation data during model training. Initially, the TRT of a word is assigned to its characters, then a small number is assigned to the last character of the word (mainly to give small values to punctuation). After tokenizing the word the span of each subword is obtained, and the maximum value in each span is taken as the final token-level fixation data.

To use this model in our setup, we need to reverse this conversion process and recompute the features from token-level back to word-level, allowing us to remap the features to a different tokenizer. This is done by summing the orignal features for all tokens corresponding to a word and then distributing them across the tokens mapped to the same word in the other tokenizer. More details of this conversion are explained in Appendix A.1.3 and an example of the process in Table 8.

Second model: Model presented in Li & Rudzicz (2021). The code and training data are also publicly available, so we trained it following their original methodology and we adapted their code to incorporate it into our implementation. The model was trained using the ZuCo 1 (Hollenstein et al., 2018) and ZuCo 2 (Hollenstein et al., 2020a) and PROVO (Luke & Christianson, 2018) datasets. For the ZuCo datasets, 800 sentences (15.7 tokens) were provided as training data and 191 sentences (3.5k tokens) were held out for evaluation. Information about the datasets used to train these models is in Table 7. The model is based on RoBERTa (Liu et al., 2019) with a regression head on each token. This head is a linear layer that outputs five features: FFD, fixProp, GPT, TRT, and nFix (Table 1). mean absolute error (MAE) for each feature is presented in Table 6. This model has 125M parameters and remains frozen during training.

In this generative model, the conversion of word-level features to token-level features during training is done by assigning the features of a word to the first token and it is assumed that the rest of the tokens of this word do not have features. We reversed this process similarly during inference by forcing the predictions for tokens that are not the first in a word to be zero. Since the maximum sequence length for RoBERTa is 512 and we are dealing with longer sequences, we implemented a



TRT per word in chosen response. TRT per word in rejected response.

Figure 3: TRT generated by first model (Huang & Hollenstein, 2023) of the chosen and rejected response to prompt 'Create a limerick about cucumbers'. Deeper colour represents longer fixation.

sliding window approach. We split the input into sequences of 512 tokens with a 50-token overlap. After processing, we combine the results using a linear combination.

Table 6: MAE performance of the model reported in Li & Rudzicz (2021).

nFix	FFD	GPT	TRT	fixProp	All (Dev)
3.984	0.713	2.424	1.556	10.781	3.892

Table 7: Overview of different corpora used to train the ET features generator models.

Corpus	Lang.	Sents.	Tokens	Subjects	Reference
Dundee	EN	2367	58598	20	Kennedy et al. (2012)
Provo	EN	189	2659	84	Luke & Christianson (2018)
ZuCo 1	EN	300	6588	12	Hollenstein et al. (2018)
ZuCo 2	EN	349	6828	18	Hollenstein et al. (2020b)
Geco	EN*	2449	-	23	Cop et al. (2017)

A.1.3 Mapping ET features between different tokenizers

Both ET features generator models used are based on different tokenizers, which are also different from the tokenizers employed by the based models used as RM. As a result, the number of tokens n in the input for the reward model and the number of tokens w for the ET features may not be the same. For **GazeAdd**, to be able to combine elementwise the ET feature embedding and the text embedding, they must have the same temporal dimensions. Therefore, we need to map the ET features per token from the ET tokenizer to the tokens of the RM tokenizer. Specifically, we convert our $f_{et} \in \mathbb{R}^{w \times f}$ (w is the number of tokens, and f is the number of features) to $f_{et}^{mapped} \in \mathbb{R}^{n \times f}$ where n is the number of tokens in the RM input. For that. we perform a mapping between the two tokenizers to obtain the *mapped features* f_{et}^{mapped}.

To map tokens generated by two different tokenizers, we use our *EyeTrackPy* python library that will be publicly released. First, we perform an initial mapping of tokens to the words they belong to in each tokenizer with some properties of *FastTokenizers* from the *transformers* library [7]. Then, we map words from one tokenizer to the words in the other and finally, we assume that the combination of the tokens that are mapped to a word in one tokenizer correspond to the tokens that are mapped to the word that is mapped to the initial word in the other tokenizer. Each row in Table 8, refers to a step in this process.

For each predictor, we reverse the method used to convert word-level features into token-level features (more details in Appendix A.1.2) but passing from tokens in the first one, to tokens in the second tokenizer. For example, if for the first ET features predictor models tokens t_1, t_2 are mapped to tokens t_1, t_2, t_3 in another second tokenizer, the values sum for all the tokens in the first list and distribute them equally across all the tokens in the second list: being t_1 (1s TRT) and t_2 (2s TRT) each of t_1, t_2, t_3 are assigned a TRT of $(1 + 2)/3 = 1s$. In Table 8 is represented a example of this process where row TRT(1) are the finatl TRT mapped for the fist ET predictor and TRT(2) for the

[7] https://huggingface.co/docs/transformers/main_classes/tokenizer

second one. Finally, because special chat tokens were removed when generating the ET features, we assign value 0 for all features in this tokens. At the time of publishing this work, some of the tokenizer functionalities needed for alignment between tokenizers were not available in Mistral 7B.

Table 8: Example of mapping TRT between two different tokenizers. TRT (1) represents the process used for the first ET predictor, and TRT(2) for the second ET predictor.

	Tokenizer 1	**Tokenizer 2**
Words	astrophotography	astrophotography
Tokens str	['_Astro', 'photo', 'graphy']	[$\dot{C}$, 'Ast', 'roph', 'ot', 'ography']
Tokens idx	[22, 23, 24]	[23, 24, 25, 26, 271]
Tokens IDs	[15001, 17720, 16369]	[198, 62152, 22761, 354, 5814]
TRT (1)	[11.23, 11.49, 10.16]	[6.58, 6.58, 6.58, 6.58, 6.58]
TRT (2)	[24.53, 0, 0]	[24.53, 0, 0, 0, 0]

A.1.4 MODELS

As RM base models we use the pretrained checkpoint of Hugging Face for Llama 3 8B [8] (Dubey et al., 2024), Llama 3 8B-instruct [9](Dubey et al., 2024) and Mistral 7B [10] (Jiang et al., 2023a).

A.1.5 TRAINING DATAILS

During training, the ET features predictor model remains frozen, while the ET features projector is normally trained without adapters (Figure 2). The RM is fine-tuned on top of the open-source models using QLoRA (Dettmers et al., 2023) based on Low-Rank Adaptation (LoRA) (Hu et al., 2021), which fine-tunes select dense layers by optimizing low-rank decomposition matrices representing weight changes, instead of directly updating pre-trained weights. QLoRA introduces memory optimization techniques such as the 4-bit NormalFloat (NF4), a novel data type, to improve performance without increasing memory usage. Following Dettmers et al. (2023) we use hyperparameters: r=8, alpha=32, and dropout=0.1.

We also fine-tune the RM embedding layer, since we are adding new tokens for the chat format and special separators tokens in our **RewardConcat** method (section 4). Also, the last layer added to the RM for the scalar reward is trained from scratch without adapters. Our implementation is based in *pytorch* and we use *transformers* [11] from Hugging Face.

Hardware. We trained the models on servers equipped with 2x Intel Xeon Platinum 8470 CPUs, 1TB of RAM, and either 2x NVIDIA H100 (80GB) or 4x NVIDIA A100 (80GB) GPUs. We always train using only GPU at a time per each model and training times were between 20 and 50 hours depending mainly on the number of steps between model evaluations.

B REWARD BENCHMARK

As we described in section 6, a future direction would be to train a Reward Model on a larger model with more data. It has been proven the scaling trends for the reward model; More data and a larger-size model generally improve accuracy (Touvron et al., 2023). Nevertheless, as a complement to our results, we evaluated our trained models on the dataset with the best results, OAAST1, in this RewardBench, a benchmark for Reward Models. RewardBench, proposed in Lambert et al. (2024), is a benchmark designed to evaluate the performance and safety of reward models. It consists of a set of datasets intended for measuring how reward models perform on challenging prompts across chat, reasoning, and safety domains, using a trio structure of prompt-chosen-rejected pairs. It comprises 2985 diverse tasks, each sample is formatted as a prompt with a manual or human-verified chosen and rejected completion. Due to its diversity of tasks (4 categories and 23 sub-categories) this

[8] https://huggingface.co/meta-llama/Meta-Llama-3-8B
[9] meta-llama/Meta-Llama-3-8B-Instruct
[10] https://huggingface.co/mistralai/Mistral-7B-v0.3
[11] https://huggingface.co/docs/transformers/index

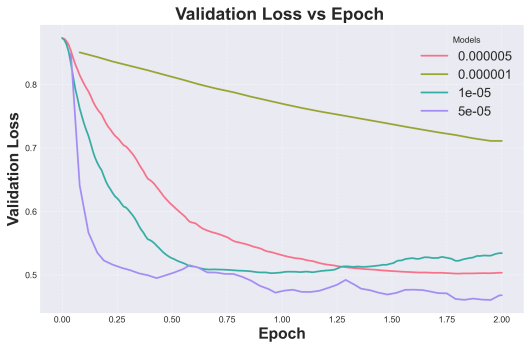


Figure 4: Validation loss with different LR on ConcatReward, batch size 8, features: $f1$ and Meta-Llama-3-8B-Instruct base model

benchmark minimizes the likelihood of overfitting. Task accuracy is calculated based on whether the chosen response receives a higher reward than the rejected response. We directly evaluate their open dataset reward-bench [12]

C HYPER PARAMETER TUNING

We performed hyperparameter tuning for the **GazeConcat** method and the baseline, and we replicated them in the **GazeAdd** method, as testing all combinations is computationally very expensive. For each dataset, 15% is reserved for validation to perform hyperparameter tuning. The best-performing checkpoints are selected based on the lowest validation loss and are subsequently used for performance evaluation in the test split. We trained for two epochs, as described in Wang et al. (2024c) and in line with trends observed in Touvron et al. (2023) where they found that training longer can lead to over-fitting. We perform a grid search for the optimal batch size, testing {8, 16, 32} values. AdamW optimizer (Loshchilov & Hutter, 2019) is used and the learning rate is tuned within the range of {1, 5, 10, 50}e-6, inspired by the values reported in Touvron et al. (2023); Cui et al. (2024); Wang et al. (2024c). Additionally, we explored different scheduler configuration, comparing constant, linear, and cosine with a minimum learning rate.

In general, the parameter that most affected the validation results was the learning rate. For the others, we ended up choosing values that worked well across all combinations. We achieved better results in both the baseline and the models concatenating the ET features with a learning rate of 0.0005 (Figure 4). In Figure 6a and Figure 6b, the validation loss and learning rate with different schedulers are shown. For lower learning rates, such as 0.00001, the scheduler had little effect. However, with higher learning rates, using a scheduler helped to mitigate overfiting. In Figure 7a and Figure 7b, it represents validation loss and learning rates with a higher learning rate of 0.0005. We opted to use this 0.00005 learning rate for all experiments, employing a cosine learning rate scheduler with a minimum learning rate of 0.7, in line with other studies such as Wang et al. (2024c); Touvron et al. (2023). We did not find a significant effect of training batch size on validation accuracy, but we opted for a value of 8, which often (especially with high learning rates and without a scheduler) was the one that tended to overfit the least (Figure 5).

ET features projector The PyTorch architecture of our ET features projector is shown below. $num_features$ varies between 1, 5, and 2 (depending on the configuration used $fcomb_1$, $fcomb_{2.5}$, $fcomb_{2.2}$ subsection 5.2). p_1, p_2 are dropout values. Finally, after testing different combinations, we used 0.1 and 0.3. This model has 0.53M parameters.

[12] `https://huggingface.co/datasets/allenai/reward-bench`

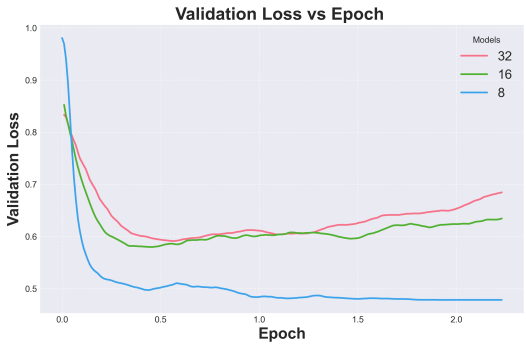
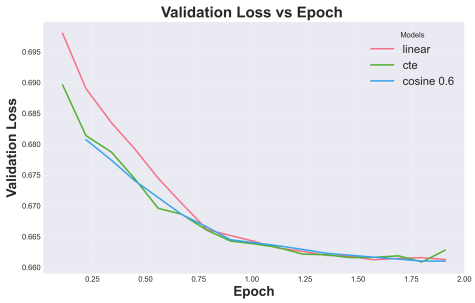
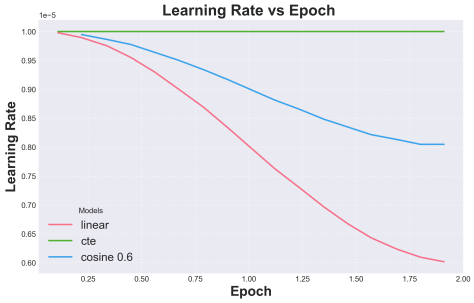


Figure 5: Validation loss with different batch size, learning rate: 5e-5, features: $f1$ and Meta-Llama-3-8B-Instruct base model

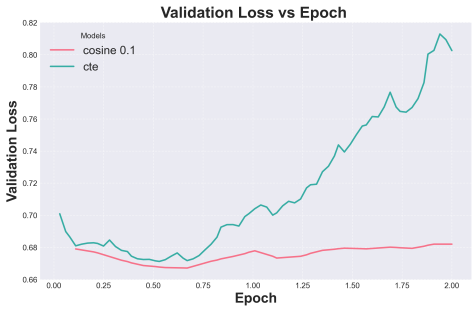


(a) Validation Loss Comparison for different LR Schedulers

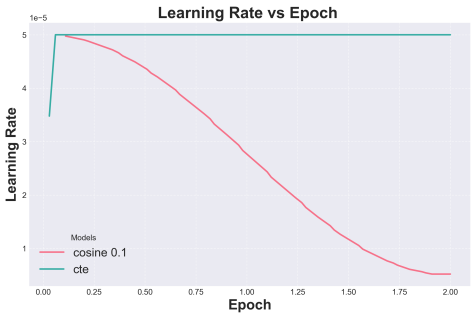


(b) Learning rate comparison for different LR Schedulers

Figure 6: ConcatReward, LR: 0.00001, features: $f1$ and Meta-Llama-3-8B-Instruct base model



(a) Validation Loss Comparison for different LR Schedulers



(b) Learning rate comparison for different LR Schedulers

Figure 7: ConcatReward, LR: 0.00005, features: $f1$ and Meta-Llama-3-8B-Instruct base model

Listing 1: PyTorch architecture of our gaze features projector

```
self.fixations_embedding_projector = nn.Sequential(
    nn.Linear(num_features, 128),
    nn.LayerNorm(128),
    nn.ReLU(),
    nn.Dropout(p=p_1),
    nn.Linear(128, hidden_size),
    nn.Dropout(p=p_2),
)
```