Aligning Language Models with Human Preferences

MARTIM SANTOS, Instituto Superior Técnico.
ANDRÉ MARTINS and SWETA AGRAWAL, Supervisors.

Large language models (LLMs) are characterized by their remarkable ability to learn extensive world knowledge and generate human-like text across diverse applications. However, the generated text often contains misleading and toxic content, emphasizing the need to align LLMs with human values and preferences to ensure more useful and secure AI systems. A widely employed strategy in numerous prominent models, including OpenAI's GPT-3.5 and GPT-4, involves Reinforcement Learning from Human Feedback (RLHF). While this method has demonstrated impressive outcomes, RLHF's complexity, instability, and sensitivity to hyperparameters challenge its empirical success and usability across various real-life scenarios. Recent reinforcement learning-free (RL-free) approaches — such as DPO, CPO, SimPO, and SLiC — address these issues. In this study, we investigate whether the promising results of RL-free methods observed in larger models extend to small language models (SLMs). Focusing on machine translation and summarization, we assess the ability of these models to efficiently learn human preferences by evaluating the quality and human alignment of their outputs, as well as their capacity to avoid common biases. Specifically, we train three compact baseline models — TinyLlama 1.1B, Gemma-2 2B, and EuroLLM 1.7B — with several RL-free methods and compare their performance against baselines. By evaluating the effectiveness of RL-free methods on smaller LLMs, this work is the first to provide a comprehensive comparison of several feedback methods applied to state-of-the-art small language models (SLMs), contributing to the development of secure and accessible AI systems suitable for resource-constrained environments.

Additional Key Words and Phrases: Language Models · Fine-tuning · Human Preferences · Alignment · Machine Translation · Summarization

1 Introduction

Large Language Models (LLMs) like OpenAI's GPT series [OpenAI 2023], Google's PaLM [Anil et al. 2023], and Meta's LLaMA [Touvron et al. 2023] leverage vast neural networks and extensive datasets to master language intricacies, enabling superior performance in tasks like translation and summarization. Fine-tuning these models through Supervised Fine-Tuning (SFT) and Instruction Tuning has advanced their practical usability. However, challenges remain, such as aligning model outputs with nuanced human feedback and societal values, mitigating inherent biases, and aligning with human preferences.

Addressing these issues involves surpassing traditional SFT by incorporating human preferences to ensure the models' outputs are both high-quality and ethical. Reinforcement Learning from Human Feedback (RLHF) marks a significant departure from traditional methods in refining LLMs, introducing a systematic approach that incorporates human preferences and evaluations. RLHF, which includes training a reward model from human feedback and optimizing the LLM policy via techniques like Proximal Policy Optimization (PPO) [Schulman et al. 2017], aims to enhance output quality and human alignment.

Despite RLHF's benefits in refining LLMs, it faces several drawbacks such as increased complexity and hyperparameter sensitivity. Alternative approaches which do not require RL, such as DPO (Direct Preference Optimization) [Rafailov et al. 2023], RSO (Statistical Rejection Sampling Optimization) [Liu et al. 2023], SimPO (Simple Preference Optimization) [Meng et al. 2024], CPO (Contrastive Preference Optimization) [Xu et al. 2024], PRO (Preference-based Reinforcement Optimization) [Song et al. 2023], and SLiC (Sequence Likelihood Calibration) [Zhao et al. 2023] offer simpler and more direct methods of optimizing these models based on human preferences. These methods avoid the intermediate step of reward model training, aiming to maintain efficiency and robustness while demonstrating competitive performance with RLHF across various tasks.

1.1 Research Objectives

Despite significant advancements and the development of alternative methods, challenges persist in efficiently managing diverse preferences and ensuring effective exploration during training across various generation tasks and model scales. A critical research gap is the systematic comparison of Preference Optimization (PO) methods, especially their application to small language models (SLMs).

Systematic comparison of PO methods allow to understand the conditions under which different PO methods are effective. Furthermore, applying these methods to smaller language models is particularly important as these models often offer a balance between computational efficiency and performance capability. Small language models[1] (SLMs), despite their widespread adoption in modern smart devices, have received significantly less attention compared to their large language model (LLM) counterparts. Even with limited capacity, these models have demonstrated a competitive advantage over larger models in reasoning tasks [Fu et al. 2023; Magister et al. 2023]. Yet, their application in natural language generation tasks remains underexplored.

This research aims to be the first comprehensive comparison of feedback strategies for SLMs, exploring how to implement advanced training architectures based on human preferences effectively. This exploration is crucial in resource-constrained settings where smaller models are prevalent but underexplored.

1.2 Main Contributions

Building upon our objectives and findings, the main contributions of this work are:

(1) **Comprehensive Study of Small LMs:** Conduct a detailed study of RL-free approaches on small LMs, focusing on the machine translation and summarization tasks. This includes the application of SFT and Preference Training using commonly used Preference Optimization methods such as DPO, CPO, SimPO, and SLiC within the context of SLMs.
(2) **Evaluation of Benchmarks and Human Preferences:** Investigate how standard benchmarks for evaluating model

Authors' Contact Information: Martim Santos, Instituto Superior Técnico.; André Martins; Sweta Agrawal, Supervisors.

[1] The definition of "small" is subjective and relative. We consider the upper limit of 5B parameters as for the size of SLMs, as done in Lu et al. [2024]

performance in these tasks correlate with human preferences for utility and safety, particularly when applied to smaller models.

(3) **Comparative Analysis Across Model Scales:** Perform an in-depth analysis of the outcomes from small models and compare these results, not only against those from larger models but also against similarly sized baseline models without additional training. This comparison aims to evaluate the consistency and reproducibility of results across different model scales.

(4) **Bias Analysis in SLMs:** Analyze potential biases inherent in smaller models due to their limited size and constrained learning capabilities, discussing implications for their deployment and use in real-world applications.

2 Background and Related Work

2.1 Large Language Models

Language models consist of artificial neural networks designed to understand and generate humanlike language. A LLM is an advanced type of language model characterized by its numerous parameters. They are usually enormous neural networks that contain billions of parameters, enabling them to learn language patterns, syntax, and semantics on an extensive scale. LLMs achieve this through the use of massive datasets during training, coupled with powerful computational resources.

2.2 Pre-training

Pre-training is the foundational stage in the development and training process of LLMs, where the model learns from vast and diverse text data. This extensive pre-training on massive datasets enables LLMs to capture a broad spectrum of linguistic knowledge, making them versatile and capable of handling various downstream natural language processing tasks effectively.

2.3 Fine-tuning

Fine-tuning is an important process that adapts pre-trained models to specific tasks, leveraging the knowledge acquired during unsupervised pre-training. This approach enables efficient transfer learning, allowing models to specialize in particular domains or applications by exposing them to task-specific datasets and information.

2.3.1 Supervised Fine-Tuning. SFT involves refining a pre-trained language model using labeled data tailored for a specific task. This process aims to improve the model's performance on targeted objectives by leveraging task-specific labeled datasets containing input examples and their corresponding desired outputs. A specialized technique, **Instruction tuning**, tailors LLMs to perform specific tasks based on explicit instructions.

2.3.2 Fine-tuning Challenges. The efficacy of the fine-tuning process has some challenges, especially in intricate tasks such as machine translation and text summarization. This misalignment becomes particularly conspicuous when the task requires a nuanced perception of context and the synthesis of information. It necessitates methodologies that move beyond traditional fine-tuning objectives, seeking to align model outputs more closely with human expectations and preferences.

2.4 RLHF

RLHF marks a significant shift from traditional supervised learning methods by focusing on aligning language models with human preferences and values. RLHF uses a systematic approach that incorporates human feedback to address the subjective complexities of language tasks and to mitigate issues related to content toxicity. This method extends across various NLP domains, significantly enhancing scalability and adaptability.

The process begins with SFT where models are initially adjusted based on human-labeled responses. The subsequent phase involves generating response pairs from the SFT model and evaluating them using human preferences. These preferences train a reward model that, through a binary classification approach, refine the training of the final language model to produce more desirable outputs.

RLHF not only allows for the integration of nuanced human judgments but also involves optimizing a reward function to produce high-quality, ethical outputs. This function is adjusted to maintain diversity in responses and prevent the generation of irrelevant high-reward outputs. Typically, RLHF employs advanced reinforcement learning techniques such as Proximal Policy Optimization (PPO) to achieve these goals, ensuring the model remains closely aligned with human preferences while enhancing its practical utility and adaptability. RLHF formulates the following optimization problem:

$$\max_{\pi_\theta} \mathbb{E}_{x\sim D, y\sim \pi_\theta(y \mid x)} \Big[r_\theta(x, y) - \beta D_{KL}(\pi_\theta(y \mid x) \parallel \pi_{\text{ref}}(y \mid x)) \Big] \quad (1)$$

where β controls the deviation from the base reference policy π_{ref}, namely the initial SFT model π_{SFT}. Moreover, in practice, the language model policy π_θ is also initialized to π_{SFT}.

In conclusion, RLHF approaches offer two notable advantages compared to traditional SFT. Firstly, they leverage both positively and negatively labeled responses, leading to a more nuanced and informative learning process. Secondly, they can engage in self-bootstrapping to iteratively refine the model's responses, contributing to continuous improvement and adaptability.

Despite its advantages, RLHF introduces significant complexities compared to traditional supervised learning approaches. The need for meticulous hyperparameter tuning, multiple model loading for PPO training, and increased memory demands a complicate implementation and compromises scalability, making RLHF less stable and more resource-intensive.

2.5 Alternative Approaches to RLHF

In addressing these limitations, recent studies suggest novel and varied offline methodologies. Recently, offline methods such as **DPO** [Rafailov et al. 2023], **SimPO** [Meng et al. 2024], **CPO** [Xu et al. 2024], and **SLiC** [Liu et al. 2023] have emerged as attractive alternatives to RLHF, offering improvements in stability and scalability while maintaining competitive performance.

DPO [Rafailov et al. 2023] is proposed as a straightforward method for policy optimization using preferences *directly*. This method introduces a new parameterization of the reward model in RLHF, allowing for the extraction of the optimal policy in closed form

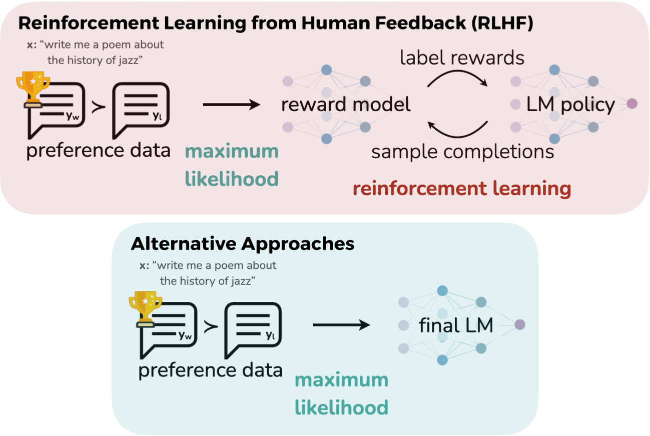

Fig. 1. **Alternative approaches optimizing for human preferences while avoiding reinforcement learning**. RLHF involve first training a reward model on human preferences for response pairs, then using reinforcement learning to maximize this reward. In contrast, the alternative approaches simplifies this process by directly optimizing a policy without the need for a reward model. Figure based on [Rafailov et al. 2023].

without an RL training loop. Ultimately, the policy objective becomes:

$$\begin{aligned}&\mathcal{L}_{\mathrm{DPO}}(\pi_\theta;\pi_{\mathrm{ref}})\\&=-\mathbb{E}_{(x,y_w,y_l)\sim D}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y_w\mid x)}{\pi_{\mathrm{ref}}(y_w\mid x)}-\beta\log\frac{\pi_\theta(y_l\mid x)}{\pi_{\mathrm{ref}}(y_l\mid x)}\right)\right].\end{aligned} \tag{2}$$

SimPO [Meng et al. 2024] refines DPO's approach by simplifying it yet showing more effectiveness. SimPO addresses two main DPO's drawbacks: (1) the requirement of a reference model π_{ref} during training incurs additional memory and computational costs, and (2) the discrepancy between the reward being optimized during training and the generation metric used for inference. SimPO employs an implicit reward formulation that directly aligns with the generation metric, eliminating the need for a reference model. Additionally, it introduces a target reward margin γ to help separating the winning and losing responses.

CPO [Xu et al. 2024] addresses the inherent limitations of traditional SFT by focusing on optimizing model outputs against a contrastive preference framework instead of merely minimizing differences from a gold-standard reference. It addresses DPO's memory- or speed- inefficiency be setting π_{ref} as a uniform prior U and obtain

$$\begin{aligned}&\mathcal{L}(\pi_\theta;\pi_{\mathrm{ref}}=U)\\&=-\mathbb{E}_{(x,y_w,y_l)\sim D}\Big[\log\sigma\Big(\beta\log\pi_\theta(y_w\mid x)-\beta\log\pi_\theta(y_l\mid x)\Big)\Big]\end{aligned} \tag{3}$$

SLiC [Liu et al. 2023] combines a rank calibration loss and a cross-entropy regularization loss, taking advantage of their simplicity and natural fit to pairwise human feedback data. The loss function is then defined as:

$$\begin{aligned}\mathcal{L}(\theta)=\max(0,\delta-\log\pi_\theta(y_w\mid x)+\log\pi_\theta(y_l\mid x))\\-\lambda\log\pi_\theta(y_{\mathrm{ref}}\mid x)\end{aligned} \tag{4}$$

3 Small Language Models (SLMs)

Recent developments in small language models (SLMs) have drawn substantial interest from both academia and industry, particularly due to their efficiency and lower computational requirements. Advances in SLMs, such as the TinyLlama [Zhang et al. 2024], Phi-2 [Microsoft 2024], Gemma-2 [Gemma Team 2024], and EuroLLM [Martins et al. 2024], demonstrate that these models can rival or even outperform larger models in various tasks. These models incorporate innovative changes in architecture and training approaches, enabling them to deliver exceptional performance for their size. Moreover, the focus on multilingual capabilities, as seen with EuroLLM, enhances their utility in diverse applications.

This trend reflects a shift in the focus of research and development efforts towards creating models that are not only powerful but also more sustainable and accessible for everyday use. Lu et al. [2024] provides a comprehensive survey of Small Language Models (SLMs), exploring their capabilities in various domains.

While SLMs show considerable promise, the application of preference alignment methods to these models is still an underexplored area that could potentially further enhance their performance and human alignment.

4 Tasks Definitions & Evaluation

4.1 Machine Translation (MT)

MT refers to the automated process of translating text information from one language to another. Central to MT is the development of models that not only translate but also correctly maintain the semantic content and syntactic structure of the original text. Datasets for machine translation typically include a pair of input-output examples that are translations of each other, i.e. parallel text. State-of-the-art datasets for MT are WMT23 [Kocmi et al. 2023], FLORES-200 [NLLB Team 2022], TICO-19 [Anastasopoulos et al. 2020], TowerBlocks [Alves et al. 2024a] and MT-Pref [Agrawal et al. 2024].

Evaluating MT systems predominantly relies on automatic metrics that compare generated translations to reference texts. Traditional metrics are BLEU [Papineni et al. 2002], METEOR [Banerjee and Lavie 2005], chrF [Popović 2015], BLEURT [Sellam et al. 2020], COMET [Rei et al. 2020] and XCOMET [Guerreiro et al. 2023].

4.2 Summarization

Summarization involves condensing a source text into a shorter form that retains the essential information and key ideas of the original content. The goal is to generate a concise summary $\hat{Y}$ from a given input sequence X, effectively capturing the key points while maintaining coherence and readability. Notable datasets include CNN/DailyMail [Hermann et al. 2015], Webis-TLDR-17 [Völske et al. 2017], Reddit TL;DR, and OpenAI TL;DR Human Feedback [Stiennon et al. 2020].

Evaluating summarization systems predominantly relies on automatic metrics that compare generated summaries to reference summaries. Traditional metrics are ROUGE [Lin 2004], METEOR [Banerjee and Lavie 2005], BERTScore [Zhang et al. 2020] and model-based metrics [Song et al. 2023] employing a reward model.

5 Challenges in Model Evaluation

When evaluating texts, various biases can affect the fairness and accuracy of assessments. Studies have identified biases such as position bias [Wang et al. 2023], where LLMs may favor responses based on their presentation order rather than content; self-enhancement bias [Zheng et al. 2023], where LLMs prefer responses they generated themselves; and length bias, where longer responses are favored irrespective of their quality. Addressing these biases is crucial for improving the reliability of LLM evaluations, with strategies like position swapping, multiple evidence calibration, and Human-in-the-Loop calibration proposed to mitigate their effects.

5.1 Verbosity Bias

Verbosity bias in LLMs is a notable challenge, particularly problematic in tasks requiring concise outputs like summarization. This bias leads LLMs to prefer longer responses even when brevity would suffice, producing impractically verbose outputs. Research on this issue, such as the studies by Zheng et al. [2023], Huang et al. [2024], and Saito et al. [2023], highlight varying susceptibility among models like GPT-4 and GPT-3.5 to verbosity, with impacts differing across tasks like creative writing and summarization.

Additionally, the influence of verbosity bias in preference optimization, particularly in smaller models, remains underexplored. Studies by Rafailov et al. [2023] and Meng et al. [2024] suggest that existing methods like DPO do not explicitly counteract verbosity, and SimPO achieves good performance without increasing response length.

6 Models and Datasets

6.1 Backbone Models

In our study, we utilize three SLM backbone models: TinyLlama 1.1B, Gemma-2 2B, and EuroLLM-1.7B, all noted for their efficiency and tailored to tasks requiring limited computational resources. TinyLlama 1.1B [Zhang et al. 2024], with its 1.1 billion parameters, performs remarkably in various downstream tasks, significantly outperforming other open-source language models of similar scale. Gemma-2 2B [Gemma Team 2024], is a cutting-edge, lightweight model developed by Google as part of the Gemma family. Finally, EuroLLM-1.7B [Martins et al. 2024] supports multiple European languages and is trained on a diverse corpus for tasks including machine translation.

6.2 Datasets

For machine translation, **MT-Pref** [Agrawal et al. 2024] dataset is employed during the training phase. This dataset includes 18,000 instances spanning 18 language directions sourced from diverse domains post-2022. In the evaluation phase, we utilize the **WMT23** General MT Data [Kocmi et al. 2023], and **FLORES-200** [NLLB Team 2022] datasets as these are standard benchmarks for reporting MT results.

For summarization, we employ the **Reddit TL;DR** dataset [Stiennon et al. 2020], derived from the Webis-TLDR-17 Corpus; and **OpenAI TL;DR Human Feedback** [Stiennon et al. 2020] which comprises 64,832 summary comparisons from the TL;DR dataset, providing human evaluative feedback. These summarization datasets are used for both training and evaluation.

7 Implemetation Details

This section outlines our comprehensive approach for training our models on the MT and summarization tasks, building on insights from literature analysis and architectural decisions. Focusing on prominent RL-free models have a competitive performance to RLHF, we train our backbone models with SFT and well-known Preference Optimization algorithms: DPO, CPO, SimPO, and SLiC.

7.1 Training Procedure

We leverage our own code base, inspired on Tower Alignment codebase from the DeepSpin project[2] that uses HuggingFace TRL [Leandro von Werra et al. 2020] alongside optimizations from DeepSpeed ZeRO3 [Rajbhandari et al. 2020] and FlashAttention-2 [Dao 2023].

Previous research on preference optimization algorithms primarily focuses on LLMs with 6B to 13B parameters, showing the effectiveness of DPO and SimPO in tasks such as summarization for certain hyperparameter configurations. However, despite the insights gained from these larger models, there is a noticeable gap in reported outcomes for smaller models. To bridge this gap, significant efforts were made to optimize hyperparameters for smaller models to ensure effective training for both machine translation and summarization tasks.

We customized scripts from the Alignment Handbook to fit our specific needs, adjusting several critical hyperparameters like learning rate, gradient accumulation steps, and batch size to suit the needs of smaller models. The adjustments were guided by systematically studying the outcomes of different configurations and exploring values that consistently improved results on smaller models. Our findings reveal that preference algorithms like DPO and CPO can be very unstable in smaller models, often leading to over-optimization that affects the generation of coherent and high-quality text. Interestingly, despite improvements in training and evaluation losses, performance metrics such as ROUGE and BLEU did not always reflect significant progress.

Through extensive testing of various hyperparameter settings, we identified configurations that stabilized performance across all models, ensuring a reliable comparison of RL-free preference optimization methods on SLMs. Ultimately, we fixed hyperparameters that balanced model output quality, training stability and the available computational resources.

7.2 Trained Models

We implement **DPO** [Rafailov et al. 2023], **CPO** [Xu et al. 2024], **SimPO** [Meng et al. 2024], and **SLiC** [Zhao et al. 2023] across both TinyLlama 1.1B and Gemma-2 2B models for machine translation and summarization tasks. Although SLiC was intended for both tasks, resource limitations prevented its implementation on the

[2] https://deep-spin.github.io/

Gemma-2 model for summarization. On EuroLLM model, all four implementations are executed. Additionally, we develop **DPO-γ**, a variant of DPO that integrates SimPO's target reward margin $\gamma > 0$ in the DPO objective, ensuring that the reward for the preferred response significantly exceeds that of the less favored response. This adjustment aims to better align the model with human preferences [Rafailov et al. 2023]. Similarly, **SLiC-DPO** combines DPO's reward reparameterization with SLiC's calibration loss, aiming to align the preference alignment ever further. While DPO-γ is implemented across all models and tasks, SLiC-DPO is also not implemented in Gemma-2 for the summarization task due to resource constraints.

Parameters. Standard parameters across all models include a learning rate of 1×10^{-7}, with $\beta = 0.1$ and $\gamma = 0.5$ for DPO-γ, and $\delta = 1$ for SLiC variants.

Prompt templates. Our models were fine-tuned using the `chatml` template [Casper et al. 2023] for machine translation. For summarization, we utilized the same prompt as Stiennon et al. [2020].

8 Experimental Results

8.1 Evaluation Setup

8.1.1 Generation. We utilized vLLM [Kwon et al. 2023] for its high-throughput capabilities and efficient memory management to generate translations and summaries finding that it offered superior performance and output quality compared to other methods. For MT and summarization generations we use greedy decoding.

8.1.2 Baselines. For machine translation, our study involves detailed evaluations of all our models against their basic or instruction-tuned forms and against open models such TowerBase 7B, TowerBase 13B and TowerInstruct 7B [Alves et al. 2024a]. In-context learning examples from a development set were used for the TowerBase models, while other MT models were evaluated in a zero-shot manner. For summarization, we compare our models against their backbone model and against Llama 7B fine-tuned with PRO [Song et al. 2023].

8.2 Machine Translation

We evaluate our fine-tuned MT models on the WMT23 test set [Kocmi et al. 2023](en↔{de,ru,zh}). Table 1 shows partially the obtained results. We also include the results on the FLORES-200 devtest set [NLLB Team 2022] (en↔{de,ru,zh,es,fr,pt,nl,it,ko}). These evaluations were done using Tower-Eval [Alves et al. 2024b]. We report system-level translation quality using chrF [Popović 2015], BLEU [Papineni et al. 2002], COMET [Rei et al. 2020] and XCOMET [Guerreiro et al. 2023]. In this section we provide some key insights we obtained in our detailed comparison.

Backbone models perform poorly prior to any fine-tuning. Both TinyLlama and Gemma2 exhibit suboptimal initial performances due to insufficient fine-tuning specific to their target tasks, and show very low scores for the evaluated metrics.

While these models provide a solid foundation for our study, we also explore EuroLLM-1.7B [Martins et al. 2024] — a recent state-of-the-art model specifically designed for machine translation — to strengthen our results. EuroLLM-1.7B-Instruct performance achieves significantly higher values compared to the other two backbone models and presents comparable performance against TowerBase models – 7B and 13B.

Models	en⟶xx		xx⟶en	
	chrF	COMET	chrF	COMET
	TinyLlama-1.1B			
TinyLlama-1.1B	18.38	48.84	24.98	48.51
| + SFT	34.01	68.74	48.43	76.97
| + DPO	35.75	69.21	50.06	77.34
| + DPO-γ	35.88	**69.24**	50.14	**77.37**
| + SimPO	**36.27**	67.09	**50.90**	76.37
| + SLiC-DPO	35.99	69.23	50.02	77.36
| + CPO	19.47	50.78	35.88	61.52
| + SLiC-CPO	12.47	38.56	28.64	53.37
	Gemma-2-2B-IT			
Gemma-2-2B-IT	27.38	55.27	41.90	72.18
| + SFT	48.07	81.28	56.94	81.78
| + DPO	48.53	**82.42**	57.47	**82.21**
| + DPO-γ	48.52	**82.42**	57.50	**82.21**
| + SimPO	**48.72**	81.37	**58.11**	81.78
| + SLiC-DPO	48.55	82.39	57.33	82.15
| + CPO	43.30	77.33	53.74	80.43
| + SLiC-CPO	42.01	76.75	53.57	80.39
	EuroLLM-1.7B-Instruct			
EuroLLM-1.7B-Instruct	49.42	82.46	56.40	81.48
| + SFT	50.48	82.64	57.07	81.68
| + DPO	50.81	82.96	57.25	**81.91**
| + DPO-γ	50.95	**82.99**	57.32	**81.91**
| + SimPO	**51.00**	82.92	**57.55**	81.91
| + SLiC-DPO	**51.00**	82.97	57.32	81.89
| + CPO	50.01	82.66	56.44	81.41
| + SLiC-CPO	49.67	82.64	56.52	81.45
| + DPO	49.37	82.60	56.67	**81.62**
| + DPO-γ	49.58	**82.65**	**56.70**	**81.62**
| + SimPO	49.27	82.44	55.21	80.93
| + SLiC-DPO	**49.64**	82.61	56.68	**81.62**
TowerBase-7B	51.65	82.76	56.84	81.03
TowerBase-13B	51.77	82.89	57.72	80.95
TowerInstruct-7B	52.25	84.28	59.78	82.77

Table 1. Machine translation model results on WMT23 dataset focusing on chrF and COMET metrics. Best ranked models for each metric are in bold. Dark blue boxes indicates that the improvement over the SFT baseline is considerable. The lesser improvements are highlighted in shallow blue boxes . Decreases in performance are marked with red boxes .

SFT increases both lexical and neural quality of the model translations. Applying SFT increases the performance of the backbone models. It enables the models to adhere to the designated chat prompts and generate more complete and accurate translations which, in turn, results in increasing both lexical and neural metrics. For EuroLLM-1.7B-Instruct we also report PO methods directly over the base model – without prior SFT – as their prior instruction tuning already makes it perform at a high level.

DPO methods consistently outperform the SFT model and other preference-based methods. Both DPO, DPO-γ, and SLiC-DPO demonstrate the best performance when considering the COMET [Rei et al. 2020] and XCOMET [Guerreiro et al. 2023] metrics compared to other preference optimization methods.

SFT outperforms CPO. While CPO enhances translation quality over the backbone model, the performance gains fall short compared to the ones achieved with SFT. SFT achieves higher COMET and XCOMET metrics than CPO for TinyLlama and Gemma. The disparity in performance, particularly in COMET and XCOMET metrics, ranges up to 14% for TinyLlama and 5% for Gemma, whereas in lexical metrics they can reach ~62%. We conclude that initial fine-tuning seems imperative for the model to grasp the essential characteristics of the expected desired translation outputs.

SFT+CPO outperforms SimPO but remains behind DPO methods. To further explore the gap between SFT and CPO, CPO was applied to the SFT models instead of the base model for TinyLlama and Gemma models. Although SLiC-DPO shows comparable results, it exhibits a marginal decline in COMET scores and a more noticeable reduction in XCOMET scores. DPO methods continue to deliver superior consistently better performance.

While SimPO achieves the highest chrF scores, its performance does not consistently outperform SFT models. SimPO is the only preference optimization method that degrades the performance of the SFT baseline for TinyLlama. Although there are visible gains in chrF and BLEU metrics, when we consider metrics like COMET and XCOMET, which focus more on semantic accuracy and contextual appropriateness, there is a decrease in performance. For Gemma-2, SimPO shows an improvement in chrF and a downgrade in performance in BLEU in both language directions. For EuroLLM, there's an enhancement across all metrics compared to the SFT baseline, except for BLEU, which sees a reduction.

Notably, SimPO is the only reference-free model applied on top of SFT. The absence of a reference model during training cuts down on memory and computational demands. However, this approach can cause performance fluctuations in SLMs, as evidenced in our results.

DPO-based models without pior SFT outperform EuroLLM-1.7B-Instruct base model. Both DPO, DPO-γ, and SLiC-DPO outperform EuroLLM-1.7B-Instruct base model when considering the COMET [Rei et al. 2020] and XCOMET [Guerreiro et al. 2023] metrics.

DPO-based models and SimPO on EuroLLM outperform TowerBase-7B and TowerBase-13B in COMET. While EuroLLM-1.7B-Instruct shows comparable performance to TowerBase-7B, when further improved through SFT and various preference optimization methods it surpasses both TowerBase models – 7B and 13B – in COMET metric.

Results on FLORES Dataset. We further evaluate our models on the FLORES dataset to assess their performance across different datasets. We summarize the key findings that replicate what was seen for WMT23: **1)** DPO methods consistently outperform the SFT model and other preference-based methods; **2)** while SimPO achieves the highest chrF scores in from English to the target language, it either maintains or degrades COMET values compared to SFT for the TinyLlama and Gemma; and **3)** DPO-based models without pior SFT outperform EuroLLM-1.7B-Instruct base model.

Remarkably on FLORES, **all our preference optimization models using EuroLLM-1.7B-Instruct show comparable COMET values against TowerBase-7B** and come close to TowerBase-13B and TowerInstruct 7B, all models of much superior size.

8.3 Summarization

We evaluate our fine-tuned models for summarization on TL;DR test set [Stiennon et al. 2020] and present the results in Table 2. We report summarization quality using the standard ROUGE metric [Lin 2004], by reporting the F1 scores for ROUGE-L and ROUGE-Lsum, METEOR [Banerjee and Lavie 2005], BERTScore [Zhang et al. 2020] and Reward, a model-based metric that replicates human preferences [Song et al. 2023]. Due to the extensive resources required to train Gemma-2 models with non reference-free approaches on the summarization dataset, we excluded both SLiC models from our evaluation.

We present the key findings for summarization in this section.

Models	ROUGE-L	METEOR	Reward
	TinyLlama-1.1B		
Tinyllama-1.1B	11.91	20.06	27.70
| + SFT	24.90	26.29	69.04
| + DPO	**24.99**	31.15	88.46
| + DPO-γ	24.64	**31.56**	**89.34**
| + SimPO	21.80	29.98	88.43
| + SLiC-DPO	24.61	31.40	89.32
| + CPO	21.82	25.74	64.92
| + SLiC-CPO	21.70	26.22	66.90
	Gemma-2-2B		
Gemma-2-2B	15.55	25.30	67.53
| + SFT	24.44	26.29	78.82
| + DPO	**25.99**	32.70	96.04
| + DPO-γ	25.80	**32.80**	**96.32**
| + SimPO	23.09	32.43	95.37
| + CPO	23.68	27.21	86.24
LLaMA-7B + PRO	33.63	–	89.71

Table 2. Summarization results on TL;DR dataset. Dark blue boxes indicates that the improvement over the SFT baseline is considerable. The lesser improvements are highlighted in shallow blue boxes. Decreases in performance are marked with red boxes.

As highlighted in Table 2, **Gemma-2 2B demonstrates a robust baseline, significantly outperforming TinyLlama-1.1B** in summarization tasks, with notable improvements in all metrics, particularly the Reward score which shows a ~240% increase over TinyLlama. This underscores Gemma-2's superior capability in generating high-quality summaries.

SFT significantly enhances the quality of summaries produced by both models. For instance, TinyLlama's Reward score jumps from 27.70 to 69.04 after applying SFT, marking a substantial improvement in summary quality.

CPO outperforms SFT on Gemma-2 but underperforms on TinyLlama. CPO significantly enhances the base model's performance, though results against SFT are mixed. Contrary to the trend observed in the machine translation task, CPO demonstrates a notable advantage over SFT on Gemma-2, particularly in the Reward metric.

DPO-based methods, including DPO, DPO-γ, SimPO, and SLiC-DPO, consistently outperform both SFT and CPO models across all metrics, particularly in terms of Reward. DPO stands out as the most effective method, surpassing SFT across all metrics for both models. DPO-γ also demonstrates strong performance, notably in the METEOR and Reward metrics, highlighting its effectiveness in aligning with human preferences in summarization.

In comparison to LLaMA-7B + PRO, the summaries produced by DPO, DPO-γ, and SimPO on Gemma-2 are preferred, although there is a decline in ROUGE-L scores.

8.4 Valuable Results and Takeaways

The study presents significant insights into the effectiveness of preference optimization (PO) methods on SLMs for the machine translation and summarization tasks. We highlight some key findings:

TinyLlama is unsuitable for MT. TinyLlama is inadequate for MT, in contrast, models like EuroLLM [Martins et al. 2024], ALMA [NLLB Team 2022], and Tower [Alves et al. 2024a], which undergo extensive fine-tuning, demonstrate superior performance.

SFT is Essential for SLMs SFT is essential across all models and tasks, consistently yielding substantial improvements over base models.

DPO outperformes CPO. Contrary to findings in larger models, namely 7B models [Agrawal et al. 2024; Saeidi et al. 2024], DPO tends to outperform CPO in smaller models for the summarization task.

PO in summarization significantly improves the quality of the summaries. Preference optimization markedly enhances summarization quality, particularly evident in metrics that align with human judgment.

9 Alignment with Human Preferences

Building on insights from Chen et al. [2024], we explore the effectiveness of preference learning algorithms by focusing on their *ranking accuracies* – a critical metric for assessing their alignment with human preferences. Earlier findings highlight a significant alignment gap, with these algorithms often achieving less than 60% accuracy on standard datasets. This finding motivates a thorough examination of the alignment of SLMs with human preferences.

We focus our study of human preferences primarily on machine translation, as constructing high-quality human preference datasets for other tasks, particularly summarization, remains costly and logistically challenging. We use the **HumanPreferences-BW** dataset [Agrawal et al. 2024], which includes 200 source instances sampled from the WMT23 English-German and Chinese-English test sets.

9.1 Accuracy Scores

Our initial evaluation involves calculating all our MT models' accuracy in preferring the chosen translation over the rejected one by analyzing their log probabilities for each translation. Table 3 shows the obtained results.

Model	**TinyLlama**	**Gemma-2**	**EuroLLM**
Base Model	0.53	0.56	0.6525
| + SFT	0.54	0.6275	0.615
| + DPO	0.5425	0.6325	0.6325
| + DPO-γ	0.5425	**0.6425**	0.64
| + SimPO	**0.5625**	0.6275	0.65
| + SLiC-DPO	0.545	0.6325	0.64
| + CPO	0.5325	0.6	0.6425
| + SLiC-CPO	0.53	0.61	0.64
| + DPO	–	–	0.6575
| + DPO-γ	–	–	0.6625
| + SimPO	–	–	**0.67**
| + SLiC-DPO	–	–	0.6625

Table 3. Model accuracy score comparison on HumanPreferences-BW. Dark blue boxes indicate a big improvement over the SFT baseline. Smaller improvements are highlighted in light blue boxes. Highest accuracy values for each model are in bold.

The analysis reveals that **SFT alone enhances the alignment of the TinyLlama and Gemma-2 models with human preferences**. Gemma-2 shows improvements across SFT, CPO, and SLiC-CPO methods. Conversely, for EuroLLM, SFT, CPO, and SLiC-CPO all reduce preference accuracy.

SimPO achieves the highest accuracy in TinyLlama and EuroLLM. SimPO, when paired with SFT, achieves the best performance outcomes for TinyLlama. Importantly, SimPO is not the model that generates higher quality responses. It decreases both COMET and XCOMET metrics compared to SFT in TinyLlama – Table 1. In contrast, SimPO alone excels in enhancing human preference alignment in EuroLLM without the need for prior SFT.

Results also show that **TinyLlama struggles with preference learning due to its size, while Gemma-2 shows the most improved alignment after SFT + PO methods applied, and EuroLLM effectively learns preferences with direct application of PO algorithms.**

9.2 Easy Preferences

Easy preferences are defined as scenarios where the model's probabilistic outputs align with human preferences. Specifically, these are instances where the domain-annotated preferred option is rated higher than the less preferred option both by the human evaluators and the base model. We assess how different models perform under these conditions, specifically focusing on their ability to consistently maintain these easy preferences across various PO strategies.

We evaluate the results of different models against a baseline also against the SFT model, when applicable. We present the results obtained in Tables 4 and 5, respectively.

Results show that **SFT reduces the retention of original preferences by approximately 14%** while **CPO preserve the most easy preferences**. Furthermore, applying preference optimization algorithms on top of SFT yields varied effects on the retention of easy preferences.

Specifically, **DPO reduces SFT correct preferences in TinyLlama.** When DPO is applied post-SFT on TinyLlama, Table 5 show that it leads to lowest accuracy of 0.935, lower than that achieved

Model	Accuracy	Avg. Chosen diff	Avg. Rejected diff	Avg. Model diff
		TinyLlama-1.1B		
Base Model	1.0 (212 samples)	–	–	–
| + SFT	0.863	+29.720	+29.702	+0.018
| + DPO	0.873	+27.061	+27.194	-0.132
| + DPO-γ	0.854	+28.768	+27.848	+0.920
| + SimPO	0.811	+21.416	+18.649	**+2.766**
| + SLiC-DPO	0.853	+28.843	+27.947	+0.897
| + CPO	**0.991**	+0.445	+0.429	+0.016
| + SLiC-CPO	0.929	+17.311	+17.163	+0.148
		Gemma-2 2B IT		
Base Model	1.0 (224 samples)	–	–	–
| + SFT	0.862	+41.322	+43.221	-1.899
| + DPO	0.830	+38.290	+37.949	+0.341
| + DPO-γ	0.830	+38.010	+37.535	+0.474
| + SimPO	0.763	+23.514	+18.0125	**+5.501**
| + SLiC-DPO	0.835	+39.250	+39.400	-0.150
| + CPO	0.884	+32.404	+32.888	-0.484
| + SLiC-CPO	**0.888**	+31.325	+31.723	-0.398
		EuroLLM-1.7B-Instruct		
Base Model	1.0 (261 samples)	–	–	–
| + SFT	0.877	+10.273	+10.569	-0.295
| + DPO	0.900	+10.446	+9.834	+0.612
| + DPO-γ	0.912	+10.474	+9.797	+0.676
| + SimPO	0.916	+10.458	+9.259	**+1.199**
| + SLiC-DPO	0.904	+0.107	-0.753	+0.861
| + CPO	**0.963**	+8.661	+8.775	-0.114
| + SLiC-CPO	0.961	+8.379	+8.580	-0.201
| + DPO	0.973	-2.306	-3.661	+1.356
| + DPO-γ	0.973	-2.524	-3.955	+1.431
| + SimPO	0.939	-20.072	-21.857	**+1.785**
| + SLiC-DPO	**0.977**	-2.389	-3.901	+1.512

Table 4. Easy Preferences Accuracy and details. Higher values for accuracy and average model difference are in bold.

Model	Accuracy	Avg. Chosen diff	Avg. Rejected diff	Avg. Model diff
		TinyLlama-1.1B + SFT (216 samples)		
+ DPO	0.935	-2.672	-2.011	-0.661
+ DPO-γ	0.949	-0.689	-1.929	+1.240
+ SimPO	0.894	-7.576	-11.381	**+3.805**
+ SLiC-DPO	**0.954**	-0.620	-1.832	+1.211
		Gemma-2 2B IT + SFT (251 samples)		
+ DPO	0.928	-2.595	-5.680	+3.085
+ DPO-γ	0.932	-2.840	-6.108	+3.268
+ SimPO	0.832	-17.273	-26.727	**+9.453**
+ SLiC-DPO	**0.952**	-1.703	-4.141	+2.438

Table 5. Easy Preferences accuracies compared to the SFT baseline.

with DPO-γ (0.949) and SLiC-DPO (0.954). Additionally, the average model difference decreases 0.661, suggesting that the model becomes less effective at distinguishing between chosen and rejected responses. This indicates that DPO may lead to the unlearning of certain preferences acquired during SFT.

Finally, **EuroLLM does not benefit from SFT in terms of retaining easy preferences** as shown in Table 4.

9.3 Hard Preferences

Hard preferences refer to scenarios where the preferences explicitly contradict the model's predictions. These are cases where the translation annotated as 'chosen' by human labelers is preferred over another, yet the model inversely ranks these choices, preferring the 'rejected' translation over the 'chosen' one.

We evaluate the results of different models against the baseline and also against the SFT model, when applicable. Tables 6 and 7 show the obtained results, respectively.

Building on the findings of Chen et al. [2024], we observe that **PO algorithms rarely flip preference rankings**.

Interestingly, **SimPO corrects the most *hard* preferences** for every baseline, across all three backbone models.

DPO-γ outperforms DPO and SLiC-DPO learning *hard* preferences. There is a consistent ~2% increase in learning *hard* preferences using the DPO variant incorporating γ. This improvement persists even when directly fine-tuning the EuroLLM-1.7B-Instruct with preference optimization methods.

Model Enhancements	TinyLlama-1.1B (188 samples)	Gemma-2 2B IT (176 samples)	EuroLLM-1.7B-Instruct (139 samples)
Base Model	0.0	0.0	0.0
| + SFT	0.176	0.330	0.122
| + DPO	0.170	0.381	0.129
| + DPO-γ	0.191	0.403	0.129
| + SimPO	**0.282**	**0.449**	**0.151**
| + SLiC-DPO	0.197	0.375	0.144
| + CPO	0.016	0.381	0.043
| + SLiC-CPO	0.080	0.256	0.036
| + DPO	–	–	0.065
| + DPO-γ	–	–	0.079
| + SimPO	–	–	**0.165**
| + SLiC-DPO	–	–	0.072

Table 6. Hard Preferences Accuracy and details. Higher values for accuracy and average model difference are in bold.

9.4 Valuable Results and Takeways

Our findings highlight that **SimPO excels in aligning difficult human preferences in translations**, particularly outperforming other methods in TinyLlama and Gemma models. **DPO-γ improves preference learning by introducing a reward margin** γ, ensuring a clearer distinction between preferred and non-preferred outputs. **CPO effectively retains easy preferences but struggles with correcting hard preferences**, especially in TinyLlama and EuroLLM.

Notably, **Gemma-2 shows the most significant improvement in aligning with human preferences** when trained with SFT + DPO-γ.

10 Length Bias

Length bias in large language models (LLMs) is a significant issue, affecting not only generative models but also the benchmarks used to estimate response quality. Studies have shown that evaluators often favor models that produce longer outputs, despite the potential for these outputs to dilute important details with unnecessary content. Building upon research with larger models, our work investigates the effects of length bias on the SLMs trained.

10.1 Length Comparison

We evaluate the output summaries generated by each model for the TL;DR test set. For TinyLlama, we extend this analysis to the the OpenAI TL;DR Human Feedback (Summarize) test set.

Our key findings are:

SFT shows the shortest lengthy summaries in average. While base models are overly verbose, when SFT is applied the average size of the summaries produced significantly decreases, for both models.

CPO produces the shortest summaries on average among the PO methods but shows the highest variance. Among the preference optimization methods, CPO produces the least verbose summaries, however, it also exhibits the greatest variance for both datasets. Results for Summarize are shown in Figure 2.

Model Enhancements	**TinyLlama-1.1B** (184 samples)	**Gemma-2 2B IT** (149 samples)	**EuroLLM-1.7B-Instruct** (154 samples)
SFT	0.0	0.0	0.0
| + DPO	0.082	0.134	0.071
| + DPO-γ	0.065	0.154	0.097
| + SimPO	**0.174**	**0.275**	**0.117**
| + SLiC-DPO	0.065	0.094	0.084
DPO	–	–	0.227
DPO-γ	–	–	0.240
SimPO	–	–	**0.279**
SLiC-DPO	–	–	0.234

Table 7. Hard Preferences accuracies compared to the SFT baseline.

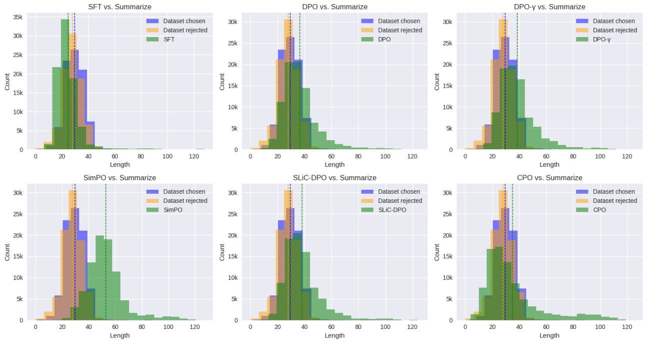

Fig. 2. Distribution of response lengths for TinyLlama models in Summarize dataset. Average lengths are marked by the dashed line.

DPO-based methods increase the length of summaries. While improving the quality of the generated summaries, DPO-based methods also result in an increase in the average length of the summaries compared to the SFT model and the reference summaries.

DPO-γ produces longer summaries compared to DPO. While the introduction of the γ parameter improves DPO-γ's performance on the Reward metric, it also results in generating longer summaries across both datasets.

SimPO generates the longest summaries. While for larger models SimPO is presented as a DPO alternative that presents minimal length exploitation [Meng et al. 2024], for smaller models it is responsible for producing, in average, the most verbose summaries apart from those produced by the base models. Particularly, there is a shift in the distribution of SimPO's output lengths towards higher values, with the density peak occurring at a greater length and a wider spread of lengths overall. Few summaries match the length of dataset's chosen summaries.

10.2 Length vs. Reward Bias

LLMs are often influenced by the verbosity of the outputs, favoring longer and more verbose texts, even if they are less succinct or of lower quality. Saito et al. [2023] highlights discrepancy between answers of LLM and those of humans.

With that, the training results insights we obtained, and both performance and length results in mind, we present a comprehensive analysis of the impact of length in the Reward metric utilized.

Our work show that:

DPO-based methods lengthy summaries tend to have high Reward. In DPO-based models, there is a noticeable trend where longer summaries generally receive higher rewards.

CPO generate longer and low reward outputs. The CPO model reflects a spread reward distribution similar to that of SFT. It displays significant variability in rewards, especially with longer sequences where outcomes can have either very low or very high reward values.

SimPO lengthy summaries are not necessarily good. Compared to SFT, SimPO generates summaries with higher reward, shifting the density of the rewards more towards zones with higher rewards. For longer outputs, there is not a necessary correlation between length and reward.

We also verify that **DPO's summaries quality are very sensitive to training learning rate** as well as **SimPO produces longer but similarly preferred responses compared to DPO across all learning rates**.

11 Conclusions

This thesis provides the first comprehensive comparison of feedback methods on state-of-the-art small language models (SLMs), thoroughly investigating RL-free approaches in machine translation and summarization tasks. Through detailed analysis using SFT and various Preference Optimization methods like DPO, CPO, SimPO, and SLiC, the study evaluates SLMs against larger models and similar-sized baselines, focusing on alignment with human preferences and the inherent challenges and biases due to their limited size and learning capacity. The findings enhance our understanding of optimizing SLMs to improve quality and human preference alignment, increasing their practical utility.

Despite its insights, this work notes significant limitations, including the absence of human expert evaluations and lack of significance testing on the obtained results. Future work should extend these foundational findings by incorporating expert evaluations, tailoring hyperparameters to specific PO approaches and tasks, and expanding the range of RL-free methods explored to further enhance the performance and preference alignment of SLMs.

References

Sweta Agrawal, José G. C. de Souza, Ricardo Rei, António Farinhas, Gonçalo Faria, Patrick Fernandes, Nuno M Guerreiro, and Andre Martins. 2024. Modeling User Preferences with Automatic Metrics: Creating a High-Quality Preference Dataset for Machine Translation. arXiv:2410.07779 [cs.CL] https://arxiv.org/abs/2410.07779

Duarte M. Alves, José Pombal, Nuno M. Guerreiro, Pedro H. Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G. C. de Souza, and André F. T. Martins. 2024a. Tower: An Open Multilingual Large Language Model for Translation-Related Tasks. arXiv:2402.17733 [cs.CL]

Duarte M. Alves, José Pombal, Nuno M. Guerreiro, Pedro H. Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G. C. de Souza, and André F. T. Martins. 2024b. Tower: An Open Multilingual Large Language Model for Translation-Related Tasks. arXiv:2402.17733 [cs.CL] https://arxiv.org/abs/2402.17733

Antonios Anastasopoulos, Alessandro Cattelan, Zi-Yi Dou, Marcello Federico, Christian Federmann, Dmitriy Genzel, Franscisco Guzmán, Junjie Hu, Macduff Hughes, Philipp Koehn, Rosie Lazar, Will Lewis, Graham Neubig, Mengmeng Niu, Alp Öktem, Eric Paquin, Grace Tang, and Sylwia Tur. 2020. TICO-19: the Translation Initiative for COvid-19. In *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020*. Association for Computational Linguistics, Online. https://doi.org/10.18653/v1/2020.nlpcovid19-2.5

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403* (2023).

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Association for Computational Linguistics, Ann Arbor, Michigan, 65–72. https://aclanthology.org/W05-0909

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. 2023. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217* (2023).

Angelica Chen, Sadhika Malladi, Lily H. Zhang, Xinyi Chen, Qiuyi Zhang, Rajesh Ranganath, and Kyunghyun Cho. 2024. Preference Learning Algorithms Do Not Learn Preference Rankings. arXiv:2405.19534 [cs.LG] https://arxiv.org/abs/2405.19534

Tri Dao. 2023. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691* (2023).

Yao Fu, Hao Peng, Litu Ou, Ashish Sabharwal, and Tushar Khot. 2023. Specializing Smaller Language Models towards Multi-Step Reasoning. arXiv:2301.12726 [cs.CL] https://arxiv.org/abs/2301.12726

Gemma Team. 2024. Gemma 2: Improving Open Language Models at a Practical Size. arXiv:2408.00118 [cs.CL] https://arxiv.org/abs/2408.00118

Nuno M. Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André F. T. Martins. 2023. xCOMET: Transparent Machine Translation Evaluation through Fine-grained Error Detection. arXiv:2310.10482 [cs.CL] https://arxiv.org/abs/2310.10482

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching Machines to Read and Comprehend. arXiv:1506.03340 [cs.CL] https://arxiv.org/abs/1506.03340

Kung-Hsiang Huang, Philippe Laban, Alexander R. Fabbri, Prafulla Kumar Choubey, Shafiq Joty, Caiming Xiong, and Chien-Sheng Wu. 2024. Embrace Divergence for Richer Insights: A Multi-document Summarization Benchmark and a Case Study on Summarizing Diverse Information from News Articles. arXiv:2309.09369 [cs.CL] https://arxiv.org/abs/2309.09369

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Philipp Koehn, Benjamin Marie, Christof Monz, Makoto Morishita, Kenton Murray, Makoto Nagata, Toshiaki Nakazawa, Martin Popel, Maja Popović, and Mariya Shmatova. 2023. Findings of the 2023 Conference on Machine Translation (WMT23): LLMs Are Here but Not Quite There Yet. In *Proceedings of the Eighth Conference on Machine Translation*, Philipp Koehn, Barry Haddow, Tom Kocmi, and Christof Monz (Eds.). Association for Computational Linguistics, Singapore, 1–42. https://doi.org/10.18653/v1/2023.wmt-1.1

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

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

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*. Association for Computational Linguistics, Barcelona, Spain, 74–81. https://aclanthology.org/W04-1013

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

Zhenyan Lu, Xiang Li, Dongqi Cai, Rongjie Yi, Fangming Liu, Xiwen Zhang, Nicholas D. Lane, and Mengwei Xu. 2024. Small Language Models: Survey, Measurements, and Insights. arXiv:2409.15790 [cs.CL] https://arxiv.org/abs/2409.15790

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2023. Teaching Small Language Models to Reason. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (Eds.). Association for Computational Linguistics, Toronto, Canada, 1773–1781. https://doi.org/10.18653/v1/2023.acl-short.151

Pedro Henrique Martins, Patrick Fernandes, João Alves, Nuno M. Guerreiro, Ricardo Rei, Duarte M. Alves, José Pombal, Amin Farajian, Manuel Faysse, Mateusz Klimaszewski, Pierre Colombo, Barry Haddow, José G. C. de Souza, Alexandra Birch, and André F. T. Martins. 2024. EuroLLM: Multilingual Language Models for Europe. arXiv:2409.16235 [cs.CL] https://arxiv.org/abs/2409.16235

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. SimPO: Simple Preference Optimization with a Reference-Free Reward. arXiv:2405.14734 [cs.CL] https://arxiv.org/abs/2405.14734

Research Microsoft. 2024. Phi-2: The Surprising Power of Small Language Models. https://www.microsoft.com/en-us/research/blog/phi-2-the-surprising-power-of-small-language-models/.

NLLB Team. 2022. No Language Left Behind: Scaling Human-Centered Machine Translation. arXiv:2207.04672 [cs.CL] https://arxiv.org/abs/2207.04672

OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774 [cs.CL]

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, 311–318. https://doi.org/10.3115/1073083.1073135

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Lisbon, Portugal, 392–395. https://doi.org/10.18653/v1/W15-3049

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

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: Memory optimizations toward training trillion parameter models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE, 1–16.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A Neural Framework for MT Evaluation. arXiv:2009.09025 [cs.CL] https://arxiv.org/abs/2009.09025

Amir Saeidi, Shivanshu Verma, and Chitta Baral. 2024. Insights into Alignment: Evaluating DPO and its Variants Across Multiple Tasks. arXiv:2404.14723 [cs.CL] https://arxiv.org/abs/2404.14723

Keita Saito, Akifumi Wachi, Koki Wataoka, and Youhei Akimoto. 2023. Verbosity Bias in Preference Labeling by Large Language Models. arXiv:2310.10076 [cs.CL] https://arxiv.org/abs/2310.10076

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

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning Robust Metrics for Text Generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (Eds.). Association for Computational Linguistics, Online, 7881–7892. https://doi.org/10.18653/v1/2020.acl-main.704

Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. 2023. Preference ranking optimization for human alignment. *arXiv preprint arXiv:2306.17492* (2023).

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

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

Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. 2017. TL;DR: Mining Reddit to Learn Automatic Summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, Lu Wang, Jackie Chi Kit Cheung, Giuseppe Carenini, and Fei Liu (Eds.). Association for Computational Linguistics, Copenhagen, Denmark, 59–63. https://doi.org/10.18653/v1/W17-4508

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023. Large Language Models are not Fair Evaluators. arXiv:2305.17926 [cs.CL] https://arxiv.org/abs/2305.17926

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024. Contrastive Preference Optimization: Pushing the Boundaries of LLM Performance in Machine Translation. arXiv:2401.08417 [cs.CL] https://arxiv.org/abs/2401.08417

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. 2024. TinyLlama: An Open-Source Small Language Model. arXiv:2401.02385 [cs.CL] https://arxiv.org/abs/2401.02385

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. arXiv:1904.09675 [cs.CL] https://arxiv.org/abs/1904.09675

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

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. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. arXiv:2306.05685 [cs.CL] https://arxiv.org/abs/2306.05685