Share Comment Star ◦◦◦

Comparison: APO-zero-unpaired -- KTO

APO paper: https://www.arxiv.org/abs/2408.06266

Karel D'Oosterlinck

Created on August 24 | Last edited on August 24

APO-zero-unpaired ablates the KL of KTO. It pushes desirable rewards above 0, undesirable rewards below 0. Not calculating the KL makes APO-zero-unpaired faster.

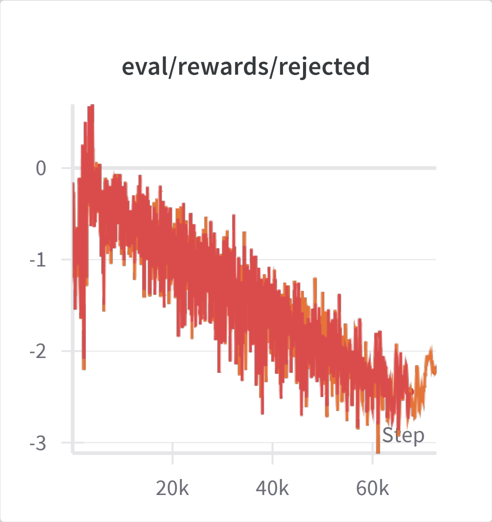
To study the different training dynamics this induces, we've run both losses on llama-3-8b-Instruct across the 4 datasets described in the APO and CLAIR paper: https://www.arxiv.org/abs/2408.06266. We will publish downstream results on these experiments soon.

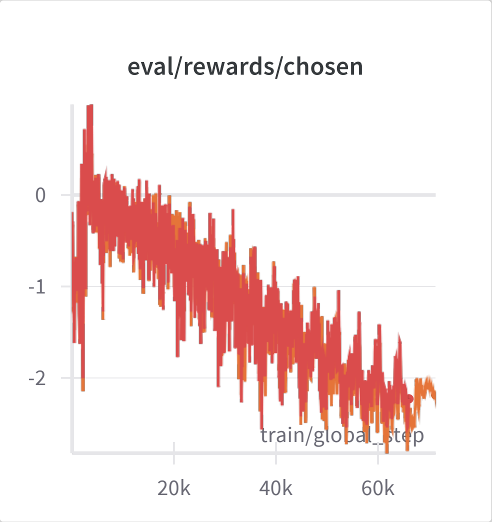
For now, let's focus on the conventional RLAIF (on-policy) dataset runs for both losses. Let's also consider dynamics on the CLAIR dataset.

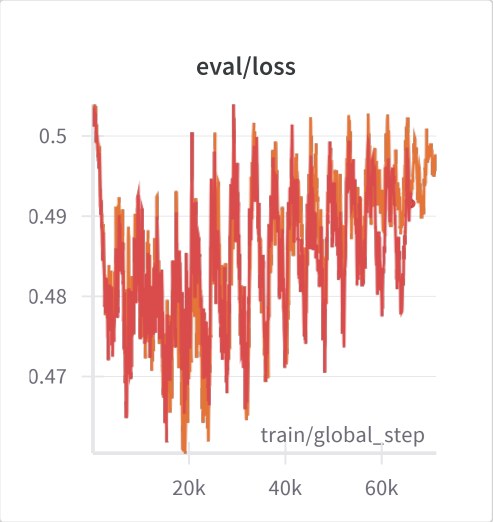
APO-zero-unpaired runs faster

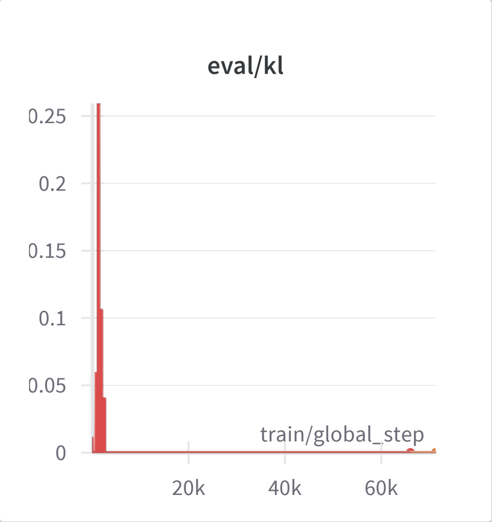
- APO-zero wall-clock time on the RLAIF dataset: 13h 49m 55s --> 830 minutes
- KTO wall-clock time on the RLAIF dataset: 19h 40m 10s --> 1180 minutes (~42% longer compare tc APO-zero)

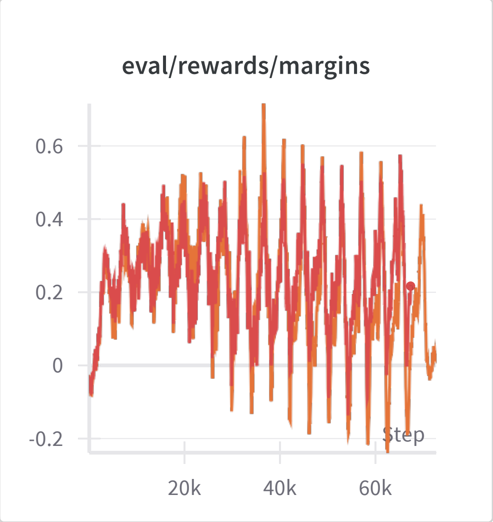
RLAIF training dynamics are very similar for both losses.


eval/rewards/rejected


eval/rewards/chosen

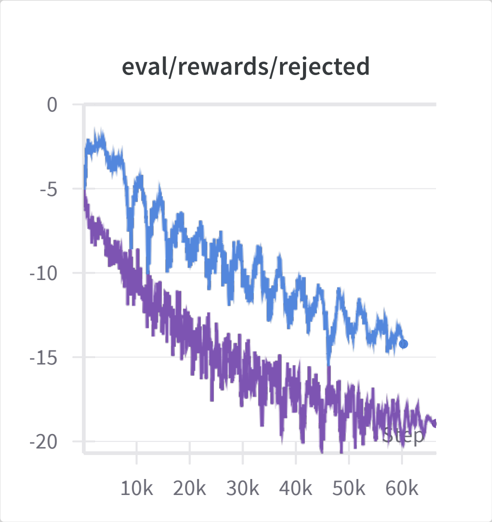

eval/loss

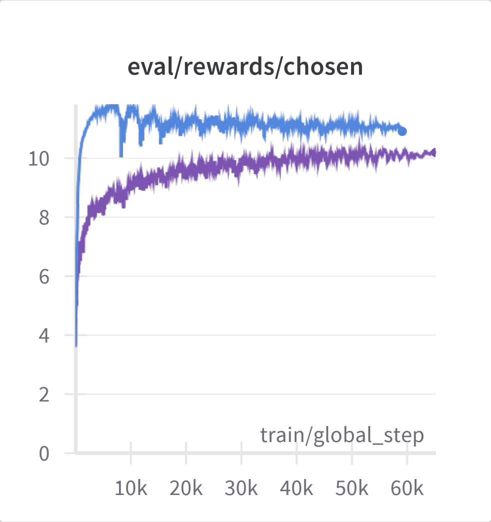

eval/kl

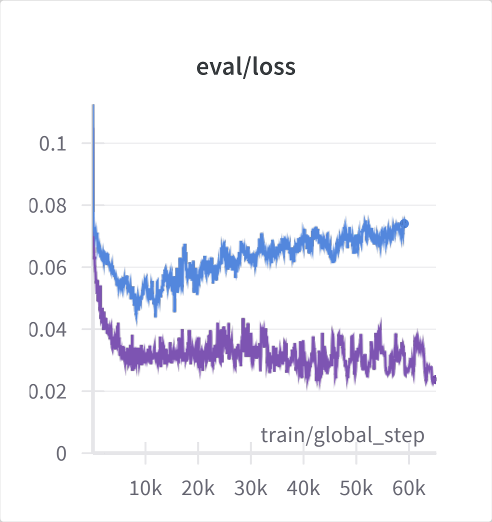

eval/rewards/margins

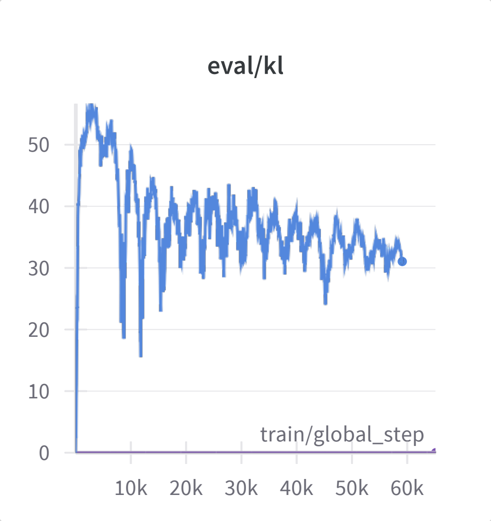
CLAIR training dynamics are different for APO-zero and KTO.

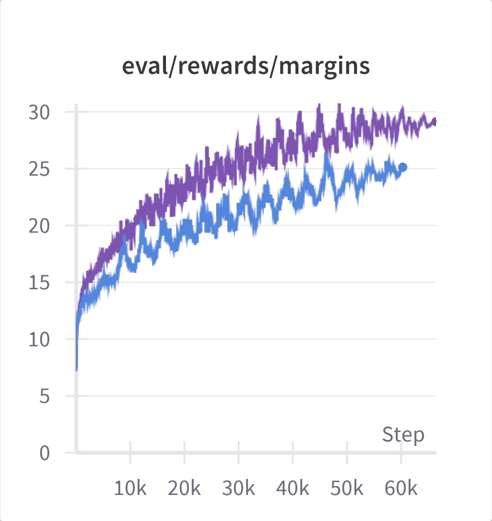
APO-zero and KTO differ on this dataset, due to the high KL for KTO here. Yet, the training dynamics of APO-zero are as intended: desirable rewards are smoothly pushed above 0, undesirable rewards are pushed below 0.


eval/rewards/rejected


eval/rewards/chosen


eval/loss


eval/kl


eval/rewards/margins

Created with ❤️ on Weights & Biases.

https://wandb.ai/contextual/apo-unpaired/reports/Comparison-APO-zero-unpaired-KTO---Vmlldzo5MTM4MjI0