Conditional Language Policy: A General Framework for Steerable Multi-Objective Finetuning

Kaiwen Wang, Rahul Kidambi, Ryan Sullivan, Alekh Agarwal, Christoph Dann, Andrea Michi, Marco Gelmi, Yunxuan Li, Raghav Gupta, Avinava Dubey, Alexandre Ramé, Johan Ferret, Geoffrey Cideron, Le Hou, Hongkun Yu, Amr Ahmed, Aranyak Mehta, Léonard Hussenot, Olivier Bachem, Edouard Leurent
Google

Abstract

Reward-based finetuning is crucial for aligning language policies with intended behaviors (*e.g.*, creativity and safety). A key challenge is to develop steerable language models that trade-off multiple (conflicting) objectives in a flexible and efficient manner. This paper presents Conditional Language Policy (CLP), a general framework for finetuning language models on multiple objectives. Building on techniques from multi-task training and parameter-efficient finetuning, CLP learn steerable models that effectively trade-off conflicting objectives at *inference time*. Notably, this does not require training or maintaining multiple models to achieve different trade-offs between the objectives. Through extensive experiments and ablations on two summarization datasets, we show that CLP learns steerable language models that outperform and Pareto-dominate the existing approaches for multi-objective finetuning.

1 Introduction

In modern applications, reinforcement learning (RL) finetuning is often a multi-objective problem due to the diversity of human preferences (*e.g.*, factuality vs. creativity) and of applications (*e.g.*, summarization, coding, dialog). Since standard RL algorithms can only maximize a *scalar* reward function, it is common practice to linearly combine the multiple, often conflicting objectives with carefully tuned *weightings* that represent the relative importance of each reward [3, 1]. However, this does not reflect the plurality of use-cases or lead to a steerable model [31].

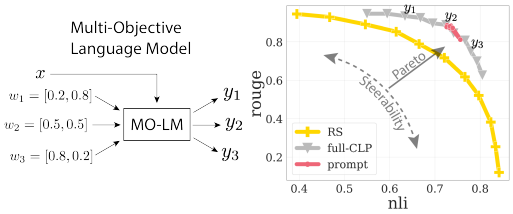


Figure 1: (Left) For prompt x, a multi-objective LM can output y_1, y_2, y_3 for different weightings w_1, w_2, w_3 of two rewards r_1 and r_2, such that the response y_i for weighting w_i is preferred under the weighted reward $w_i[1]r_1 + w_i[2]r_2$. (Right) Pareto-fronts when using the rewards NLI and Rouge (App. A.1.2). Rewarded Soups (RS) [23] is *Pareto-dominated* by both full-CLP (*this paper*) and prompting (say, [15]), but full-CLP is more appealing for its steerability, evidenced by its wider Pareto-front. Pareto-dominance (pushing out the front) and steerability (stretching out the front) are both key desiderata for MOFT.

To address the limitations of single-objective finetuning (SOFT), multi-objective finetuning (MOFT) is a new paradigm that learns a multi-objective LM (see Fig. 1 left), which can be steered at inference to generate outputs over the continuum of reward weightings *without any retraining* [23]. Specifically, a multi-objective LM takes a desired reward weighting at inference time and outputs a response that maximizes this weighted combination of rewards.

MOFT has been explored via prompt-based approaches [15, 11, 35] and parameter-based approaches [23, 15]. Prompt-based approaches finetune a LM that is steered by simply including the reward

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

weightings into the prompt. However, prompt-based MOFT is *sub-optimal in steerability* as we show in our experiments and can be sensitive to how weightings are encoded in the prompt. An alternative parameter-based approach, Rewarded Soups (RS) [23], (a) *independently* trains one LM per reward function, and (b) interpolates parameters from separate LMs with reward weightings to perform conditional generation at inference-time. Perhaps surprisingly, this *zero-shot* approach can effectively trade-off multiple rewards by relying on the linear mode connectivity [8]. However, we find that zero-shot MOFT is sub-optimal on weightings not seen during training.

This paper presents Conditional Language Policy (CLP), a general MOFT *framework* that employs parameter-space conditioning and multi-task training [6]. Using parameter-conditioning from RS, CLP is *consistently more steerable* than purely prompt-based approaches. Moreover, by finetuning on a diverse set of reward weightings, CLP produces *higher quality responses* than zero-shot approaches like RS while having comparable or better steerability. We conduct a systematic set of experiments, observing that CLP both Pareto-dominates RS and is more steerable than prompt-based MOFT (see Fig. 1 right). CLP robustly maintains the above benefits across many experimental conditions, including various choice of rewards and model sizes. We also conduct an automated evaluation with Gemini 1.0 Ultra [33] that further supports that CLP is more steerable and generates higher quality responses than existing baselines.

In summary, our contributions are:

1. We propose CLP, a general framework for MOFT that learns multi-objective LMs through multi-task learning and parameter efficient model averaging (Sec. 3).
2. We extensively evaluate CLP on summarization and show it robustly improves existing approaches in *both* output quality and steerability, across many experimental conditions and automated evaluator evaluations (Sec. 4).
3. We theoretically prove that logit mixing, a special case of CLP, is near-optimal under a coverage condition. We also provide an example instance where zero-shot (*i.e.*, without multi-task training) methods fail while CLP succeeds (App. C).

2 Problem Setup

Let $\pi_{\text{ref}}(y \mid x)$ be a base policy with parameters θ_{ref}, where x is the input prompt and y is the output generation. In **single-objective finetuning** (SOFT), there is a fixed reward function $R(x, y)$ and the goal is to maximize the expected reward without drifting too far from π_{ref}. Formally, SOFT learns a policy $\pi_\theta(\cdot)$ to maximize the following value:

$$V_{\alpha,R}(\pi) := \mathbb{E}_{x\sim\mathcal{D},y\sim\pi(x)}[(1-\alpha)R(x,y) - \alpha\,\mathrm{KL}(\pi(\cdot \mid x) \parallel \pi_{\text{ref}}(\cdot \mid x))], \tag{1}$$

where $\alpha \in (0, 1)$ is a fixed weighting for the reverse KL w.r.t. π_{ref}.[1] The main issue with SOFT is that both the reward R and KL weighting are fixed; thus, it is not possible to offer near-optimal behavior on other reward and KL weightings without retraining multiple times.

Multi-objective finetuning (MOFT) fixes the above issues by learning a *conditional* policy that generates y based not only on the prompt x, but also the α and R specified at test-time *without retraining*. We formulate MOFT using multi-objective RL [25] and consider a vectorial reward $\boldsymbol{R}(s, a) \in \mathbb{R}^m$ whose linear scalarizations $\{w^\top \boldsymbol{R}(\cdot) : w \in \Delta_m\}$ capture all possible trade-offs at test-time, where Δ_m is the $(m-1)$-simplex. The goal is to learn parameters ϕ such that, for all *weightings* $\alpha \in [\alpha_{\min}, 1]$ and $w \in \Delta_m$, the *conditioned* policy $\pi_\phi(\cdot; \alpha, w)$ maximizes $V_{\alpha,w^\top \boldsymbol{R}}$, the objective with KL-regularizer α and reward function $w^\top \boldsymbol{R}(\cdot)$. We frame MOFT as multi-task training over the weighting distribution $\mathcal{Q}$, and aim to maximize:

$$V_{\text{moft}}(\phi) = \mathbb{E}_{(\alpha,w)\sim\mathcal{Q}}[V_{\alpha,w^\top \boldsymbol{R}}(\pi_\phi(\cdot; \alpha, w))]. \tag{2}$$

MOFT Desiderata – Pareto-dominance & steerability. A multi-objective LM π *Pareto-dominates* another π' if $V_{\alpha,w^\top \boldsymbol{R}}(\pi(\cdot; \alpha, w)) \geq V_{\alpha,w^\top \boldsymbol{R}}(\pi'(\cdot; \alpha, w))$ for all values of α, w that one cares about. New MOFT algorithms should ideally Pareto-dominate existing baselines to ensure that generation quality is improved along all axes. *Steerability* is another important goal for MOFT algorithms. In Fig. 1, full-CLP and prompting both satisfy the first goal of Pareto-dominance, but full-CLP is desirable since its Pareto-curve has much better spread, *i.e.*, it is more steerable.

[1] For distributions $P \ll Q$, the KL-divergence is defined as $\mathrm{KL}(P \parallel Q) := \mathbb{E}_P[\log \frac{\mathrm{d}P}{\mathrm{d}Q}]$.

Algorithm 1 CLP: Conditional Language Policy

Input: Weightings sampler $\mathcal{Q}$, number of training steps T, learning rates $\{\eta_t\}_{t=1}^{T}$.

1: Init CLP: $\phi_0 = (\theta_{\text{ref}}[\mathcal{S}^C], \{\theta_{\text{ref}}[\mathcal{S}]\}_{i\in[m]})$.
2: **for** t in $0 \dots T-1$ **do**
3: Sample prompt $x_t \sim \mathcal{D}$ and KL & reward weightings $(\alpha_t, w_t) \sim \mathcal{Q}$.
4: Get conditioned policy $\pi_t(\cdot; \alpha_t, w_t)$ from running Alg. 2 with α_t, w_t, ϕ_t.
5: Conditioned generation $y_t \sim \pi_t(x_t; \alpha_t, w_t)$ (same computation as single LM).
6: Objective: $r_t \leftarrow (1-\alpha_t) \cdot w_t^\top \boldsymbol{R}(x_t, y_t) - \alpha_t \cdot \text{KL}(\pi_t(\cdot \mid x_t; \alpha_t, w_t) \parallel \pi_{\text{ref}}(\cdot \mid x_t))$.
7: Update CLP parameters: $\phi_{t+1} \leftarrow \phi_t + \eta_t \cdot g_t$ where $g_t = r_t \cdot \nabla_{\phi_t} \log \pi_t(y_t \mid x_t; \alpha_t, w_t)$.
8: **end for**
9: **Output:** CLP parameters ϕ_T.

Algorithm 2 Conditioning Mechanism

Fixed: KL-mixing function $f_{\text{mix}}(\cdot)$, index of conditioning params $\mathcal{S}$, CONDPROMPT = FALSE.

Input: KL & reward weightings (α, w), CLP parameters $\phi = (\theta_{\mathcal{S}^C}, \{\theta_{\mathcal{S}}^{(i)}\}_{i\in[m]})$.

1: Set $\beta = f_{\text{mix}}(\alpha)$ and compute:

$$\theta_{\mathcal{S}}^{\alpha,w} \leftarrow (1-\beta) \textstyle\sum_{i=1}^{m} w[i] \cdot \theta_{\mathcal{S}}^{(i)} + \beta \cdot \theta_{\text{ref}}[\mathcal{S}]. \tag{3}$$

2: Combine params $\theta^{\alpha,w} = \theta_{\mathcal{S}}^{\alpha,w} \oplus \theta_{\mathcal{S}^C}$.
3: **Return:** Policy that concatenates weights to input $\{x \mapsto \pi_{\theta^{\alpha,w}}(\texttt{CONCAT}([w, x]))\}$ if COND-PROMPT. Else, return policy $\pi_{\theta^{\alpha,w}}$

Notation. θ refers to LM parameters, while ϕ refers to CLP parameters (different structure from θ), which can be conditioned on (α, w) to produce a conditioned LM parameter $\theta^{\alpha,w}$ (same structure as θ). $\theta[\mathcal{S}]$ or $\theta_{\mathcal{S}}$ refer to the subset of parameters indexed by $\mathcal{S}$. We use $\oplus$ to combine disjoint parameter subsets, *i.e.*, $\theta = \theta[\mathcal{S}^C] \oplus \theta[\mathcal{S}]$. We assume θ and θ_{ref} have the same structure.

3 Conditional Language Policy (CLP)

We describe the CLP algorithm in Alg. 1, illustrated in Fig. 12), where each training round $t = 1, 2, \dots, T$ consists of three steps. First, we sample a prompt x_t and reward & KL weightings $w_t, \alpha_t \sim \mathcal{Q}$ for this round (Line 3). Second, we condition CLP on weightings (α_t, w_t) and sample generations $y_t \sim \pi_t(x_t; \alpha_t, w_t)$ (Line 5). Third, we compute the conditioned objective (Line 6) and update the CLP parameters with policy gradient (Line 7). The policy optimization step uses gradient ascent to maximize the objective in Eq. (2) and can be implemented by any standard approaches such as REINFORCE [39, 2], PPO [28], and also DPO [20] when given preference data.

We also describe the mechanism for computing conditional policies $\pi(\cdot; \alpha, w)$ in Alg. 2. Let $\mathcal{S}$ denote an index set on the LM parameters that we wish to use for parameter-conditioning [23], *e.g.*, attention weights, and its choice can trade-off steerability and memory cost of CLP. Then, we maintain (1) m sets of conditioned parameters $\{\theta_{\mathcal{S}}^{(i)}\}_{i\in[m]}$ indexed by $\mathcal{S}$, and (2) one set of unconditioned parameters $\theta_{\mathcal{S}^C}$. To condition the $\mathcal{S}$-part on (α, w), we linearly combine the m conditioning parameters with weightings w, and we then combine the result with $\theta_{\text{ref}}[\mathcal{S}]$ weighted by $f_{\text{mix}}(\alpha)$ (Eq. (3)). Then, we concatenate the conditioned part $\theta_{\mathcal{S}}^{\alpha,w}$ with the unconditioned part to obtain the full LM parameters $\theta^{\alpha,w} = \theta_{\mathcal{S}}^{\alpha,w} \oplus \theta_{\mathcal{S}^C}$. The parameter count of CLP is thus $\mathcal{O}(m|\mathcal{S}| + |\mathcal{S}^C|)$. We note that inference with CLP only requires one forward pass through the LM and the above parameter-averaging cost is amortized since it is done only once at the beginning.

The choice of $\mathcal{S}$ influences both the steerability and memory usage of CLP. On one extreme, the most steerable and high parameter count choice is to condition on all LM parameters and we call this full-CLP, *i.e.*, $\mathcal{S}_{\text{full}} = \{\text{indices of all LM parameters}\}$. This instance is inspired by model soups [41]. On the other extreme, logit-CLP only conditions on the final linear layer (a.k.a. logit layer), *i.e.*, $\mathcal{S}_{\text{logit}} = \{\text{indices of last linear layer of LM}\}$. This instance is theoretically grounded [17] but we found it to have inferior steerability. Finally, a great middle ground for transformer LMs is to condition on the attention parameters and we call this attn-CLP, *i.e.*, $\mathcal{S}_{\text{attn}} = \{\text{indices of attention layers of LM}\}$.

attn-CLP is more parameter-efficient and nearly as steerable as full-CLP in our experiments. We highlight that CLP is agnostic to the LM architecture and $\mathcal{S}$ can be set appropriately for other models.

In our experiments, we observe that the more expressive parameterizations of $\mathcal{S}$ (*e.g.*, $\mathcal{S}_{\text{full}}$ and $\mathcal{S}_{\text{attn}}$) robustly lead to Pareto-dominating and highly steerable behaviors than existing baselines such as Rewarded Soups. We remark that expressivity is determined not just by the number of parameters in $\mathcal{S}$ but also where those parameters are in the LM (*e.g.* earlier vs. later layers).

Prompt-based conditioning: One can also augment the prompt with the reward weightings (Line 3 in Alg. 2); see also App. F.2 for more details on prompt-design. Prompting based MOFT has been explored in recent manuscripts [15, 11, 35] and has the advantage of not requiring additional parameters, but consumes part of the context and is sensitive to how these are encoded in the context. In App. A.2.2, we consider augmenting parameter-space conditioning with prompting.

4 Experiments

We consider the following questions:

- **Benchmarking:** How do different methods perform in terms of *performance* (ability to push out the Pareto Front) and *steerability* (ability to generate content that trades-off different objectives)?
- **Ablations:** How does the behavior of different approaches vary as a function of (a) number of finetuning steps, (b) model size? Furthermore, is parameter space conditioning composable with prompting based methods?
- **Automated Evaluations:** Going beyond Pareto fronts, we present automated evaluations that compare generations from CLP against baselines by having Gemini [33] rate the summaries in terms of quality and steerability.

Data and models. A majority of our experiments/ablations are performed using summarization with the widely-used XSum dataset [18]. We initialize the reference policy π_{ref} and reward models from the instruction finetuned (FLAN) checkpoints for T5 [4]. We use the large size (770M parameters) for reward models and we mostly use the base size (220M parameters) for policies, except in our model size ablation where we also use large size for policies. For policy optimization, we use REINFORCE with control variate, which is a lighter implementation than PPO [28] and has been successfully used for summarization [27].

Reward functions. We consider three reward functions: (1) ROUGE, a formulaic reward that measures similarity of generation to the ground truth summary [16]; (2) natural language inference (NLI), a learnt reward model for textual entailment and factuality [19, 27]; and (3) a reward model for summary quality learnt from the "too long; didn't read" (TLDR) dataset [32]. ROUGE and the quality model (referred to as TLDR) tend to favor verbose and descriptive summaries while NLI favors concise summaries; this gives our setup a distinct tension between various reward pairs.

Methods. We benchmark the three instances of CLP in Sec. 3: full-CLP, attn-CLP, logit-CLP. As a gentle reminder, full-CLP maintains one full LM per reward, akin to standard model soups; attn-CLP maintains replicas of all attention layers thus being more parameter efficient; and logit-CLP maintains replicas only the logit layer which is theoretically grounded but less expressive.

Baselines. We consider (a) Rewarded Soups (RS) [23] which independently trains one policy per reward and linearly interpolates the parameters with weightings at inference time – this is a 'zero-shot' version of full-CLP; (b) a prompting baseline [15, 11], which encodes the reward weightings into the prompt and is trained with our multi-task objective – for details on the prompt and how it was selected, see App. F.2. For the "single-reward, multi-KL" setting (App. A.1.1), instead of RS, we consider (c) the recent Decoding-time Realignment (DeRA) [17], which maintains two LMs (an LM optimized for $\alpha_{\min}$ and reference LM π_{ref}) and linearly interpolates their logits. All methods and baselines are run for *same number of training iterations*.

Note. Due to lack of space, we present experiment results in App. A, where we find that CLP consistently improves over the baselines in both steerability and Pareto-dominance. In addition, we also scale up the policy (to T5-xl size) and reward models (to T5-xxl size) and train the policy on the TLDR dataset in App. A.4 to check if our observations robustly translate to larger scales and other datasets, which we find that they do.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting reinforce style optimization for learning from human feedback in llms. *arXiv preprint arXiv:2402.14740*, 2024.

[3] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

[4] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024.

[5] Nikolaos Dimitriadis, Pascal Frossard, and François Fleuret. Pareto manifold learning: Tackling multiple tasks via ensembles of single-task models. In *International Conference on Machine Learning*, pages 8015–8052. PMLR, 2023.

[6] Alexey Dosovitskiy and Josip Djolonga. You only train once: Loss-conditional training of deep networks. In *International conference on learning representations*, 2020.

[7] Jacob Eisenstein, Chirag Nagpal, Alekh Agarwal, Ahmad Beirami, Alex D'Amour, DJ Dvijotham, Adam Fisch, Katherine Heller, Stephen Pfohl, Deepak Ramachandran, et al. Helping or herding? reward model ensembles mitigate but do not eliminate reward hacking. *arXiv preprint arXiv:2312.09244*, 2023.

[8] Jonathan Frankle, Gintare Karolina Dziugaite, Daniel Roy, and Michael Carbin. Linear mode connectivity and the lottery ticket hypothesis. In *International Conference on Machine Learning*, pages 3259–3269. PMLR, 2020.

[9] Jean-Baptiste Gaya, Laure Soulier, and Ludovic Denoyer. Learning a subspace of policies for online adaptation in reinforcement learning. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=4Muj-t_4o4`.

[10] Han Guo, Ramakanth Pasunuru, and Mohit Bansal. Autosem: Automatic task selection and mixing in multi-task learning. In *Proc. of NAACL*, 2019.

[11] Yiju Guo, Ganqu Cui, Lifan Yuan, Ning Ding, Jiexin Wang, Huimin Chen, Bowen Sun, Ruobing Xie, Jie Zhou, Yankai Lin, Zhiyuan Liu, and Maosong Sun. Controllable preference optimization: Toward controllable multi-objective alignment. *CoRR*, abs/2402.19085, 2024. doi: 10.48550/ARXIV.2402.19085. URL `https://doi.org/10.48550/arXiv.2402.19085`.

[12] 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. *arXiv preprint arXiv:2106.09685*, 2021.

[13] Costa Huang and Arash Ahmadian. Putting rl back in rlhf. `https://huggingface.co/blog/putting_rl_back_in_rlhf_with_rloo`, 2024. Accessed: 2024-06-14.

[14] Minyoung Hwang, Luca Weihs, Chanwoo Park, Kimin Lee, Aniruddha Kembhavi, and Kiana Ehsani. Promptable behaviors: Personalizing multi-objective rewards from human preferences. *CoRR*, abs/2312.09337, 2023. doi: 10.48550/ARXIV.2312.09337. URL `https://doi.org/10.48550/arXiv.2312.09337`.

[15] Joel Jang, Seungone Kim, Bill Yuchen Lin, Yizhong Wang, Jack Hessel, Luke Zettlemoyer, Hannaneh Hajishirzi, Yejin Choi, and Prithviraj Ammanabrolu. Personalized soups: Personalized large language model alignment via post-hoc parameter merging. *arXiv preprint arXiv:2310.11564*, 2023.

[16] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

[17] Tianlin Liu, Shangmin Guo, Leonardo Bianco, Daniele Calandriello, Quentin Berthet, Felipe Llinares, Jessica Hoffmann, Lucas Dixon, Michal Valko, and Mathieu Blondel. Decoding-time realignment of language models. *arXiv preprint arXiv:2402.02992*, 2024.

[18] Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1206. URL `https://aclanthology.org/D18-1206`.

[19] Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial NLI: A new benchmark for natural language understanding. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.441. URL `https://aclanthology.org/2020.acl-main.441`.

[20] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 2023.

[21] 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. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL `http://jmlr.org/papers/v21/20-074.html`.

[22] Alexandre Rame, Matthieu Kirchmeyer, Thibaud Rahier, Alain Rakotomamonjy, Patrick Gallinari, and Matthieu Cord. Diverse weight averaging for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 35:10821–10836, 2022.

[23] Alexandre Ramé, Guillaume Couairon, Corentin Dancette, Jean-Baptiste Gaya, Mustafa Shukor, Laure Soulier, and Matthieu Cord. Rewarded soups: towards pareto-optimal alignment by interpolating weights fine-tuned on diverse rewards. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL `http://papers.nips.cc/paper_files/paper/2023/hash/e12a3b98b67e8395f639fde4c2b03168-Abstract-Conference.html`.

[24] Adam Roberts, Hyung Won Chung, Gaurav Mishra, Anselm Levskaya, James Bradbury, Daniel Andor, Sharan Narang, Brian Lester, Colin Gaffney, Afroz Mohiuddin, et al. Scaling up models and data with t5x and seqio. *Journal of Machine Learning Research*, 24(377):1–8, 2023.

[25] Diederik M Roijers. Multi-objective decision-theoretic planning. *AI Matters*, 2(4):11–12, 2016.

[26] Diederik M Roijers, Peter Vamplew, Shimon Whiteson, and Richard Dazeley. A survey of multi-objective sequential decision-making. *Journal of Artificial Intelligence Research*, 48: 67–113, 2013.

[27] Paul Roit, Johan Ferret, Lior Shani, Roee Aharoni, Geoffrey Cideron, Robert Dadashi, Matthieu Geist, Sertan Girgin, Léonard Hussenot, Orgad Keller, et al. Factually consistent summarization via reinforcement learning with textual entailment feedback. *ACL*, 2023.

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

[29] Noam Shazeer and Mitchell Stern. Adafactor: Adaptive learning rates with sublinear memory cost. In *International Conference on Machine Learning*, pages 4596–4604. PMLR, 2018.

[30] Ruizhe Shi, Yifang Chen, Yushi Hu, ALisa Liu, Noah Smith, Hannaneh Hajishirzi, and Simon Du. Decoding-time language model alignment with multiple objectives. *arXiv preprint arXiv:2406.18853*, 2024.

[31] Taylor Sorensen, Jared Moore, Jillian Fisher, Mitchell L Gordon, Niloofar Mireshghallah, Christopher Michael Rytting, Andre Ye, Liwei Jiang, Ximing Lu, Nouha Dziri, Tim Althoff, and Yejin Choi. Position: A roadmap to pluralistic alignment. In *Forty-first International Conference on Machine Learning*, 2024. URL `https://openreview.net/forum?id=gQpBnRHwxM`.

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

[33] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[34] Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. Tl; dr: Mining reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, pages 59–63, 2017.

[35] Haoxiang Wang, Yong Lin, Wei Xiong, Rui Yang, Shizhe Diao, Shuang Qiu, Han Zhao, and Tong Zhang. Arithmetic control of llms for diverse user preferences: Directional preference alignment with multi-objective rewards. *CoRR*, abs/2402.18571, 2024. doi: 10.48550/ARXIV.2402.18571. URL `https://doi.org/10.48550/arXiv.2402.18571`.

[36] Kaiwen Wang, Nathan Kallus, and Wen Sun. The central role of the loss function in reinforcement learning. *arXiv preprint arXiv:2409.12799*, 2024.

[37] Kaiwen Wang, Dawen Liang, Nathan Kallus, and Wen Sun. Risk-sensitive rl with optimized certainty equivalents via reduction to standard rl. *arXiv preprint arXiv:2403.06323*, 2024.

[38] Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. Large language models are not fair evaluators. *arXiv preprint arXiv:2305.17926*, 2023.

[39] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256, 1992.

[40] Mitchell Wortsman, Maxwell C Horton, Carlos Guestrin, Ali Farhadi, and Mohammad Rastegari. Learning neural network subspaces. In *International Conference on Machine Learning*, pages 11217–11227. PMLR, 2021.

[41] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pages 23965–23998. PMLR, 2022.

[42] Rui Yang, Xiaoman Pan, Feng Luo, Shuang Qiu, Han Zhong, Dong Yu, and Jianshu Chen. Rewards-in-context: Multi-objective alignment of foundation models with dynamic preference adjustment. *CoRR*, abs/2402.10207, 2024. doi: 10.48550/ARXIV.2402.10207. URL `https://doi.org/10.48550/arXiv.2402.10207`.

[43] Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adaptive budget allocation for parameter-efficient fine-tuning. In *International Conference on Learning Representations*. Openreview, 2023.

[44] Yifan Zhong, Chengdong Ma, Xiaoyuan Zhang, Ziran Yang, Qingfu Zhang, Siyuan Qi, and Yaodong Yang. Panacea: Pareto alignment via preference adaptation for llms. *CoRR*, abs/2402.02030, 2024. doi: 10.48550/ARXIV.2402.02030. URL `https://doi.org/10.48550/arXiv.2402.02030`.

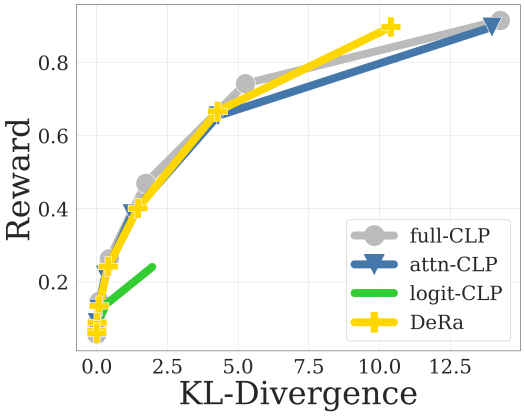
Appendices

A Experimental Results

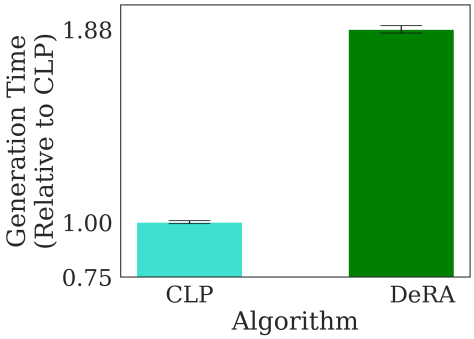
A.1 Core Benchmarking Results

A.1.1 Single Reward, Multi KL Regularizer

In the first setting, we fix a single reward function and vary the KL regularizer α to test the KL-reward trade-off. We use the reward $R = 0.9R_{\text{nli}} + 0.1R_{\text{rouge}}$, where R_{rouge} is mixed in to mitigate reward hacking the NLI model. In Fig. 2(a), we see that all methods except logit-CLP are able to evenly trade-off reward and KL, enabling a smooth transition from π_{ref} (when $\alpha = 1$) to a maximally finetuned model (when $\alpha = 0.01$). This suggests that parameter-mixing trained with the multi-task objective is competitive with the baseline DeRa, which is state-of-the-art for reward-KL trade-off. Moreover, Fig. 2(b) shows that CLP is also $\sim 2\times$ more computationally efficient than DeRa at inference-time, because DeRa performs two LM calls (both π_{ref} and $\pi_{\alpha_{\min}}$) per token. Hence, inference speed is a major benefit of parameter-mixing over logit-ensembling.



(a) reward-KL curve



(b) relative generation time for 1000 prompts.

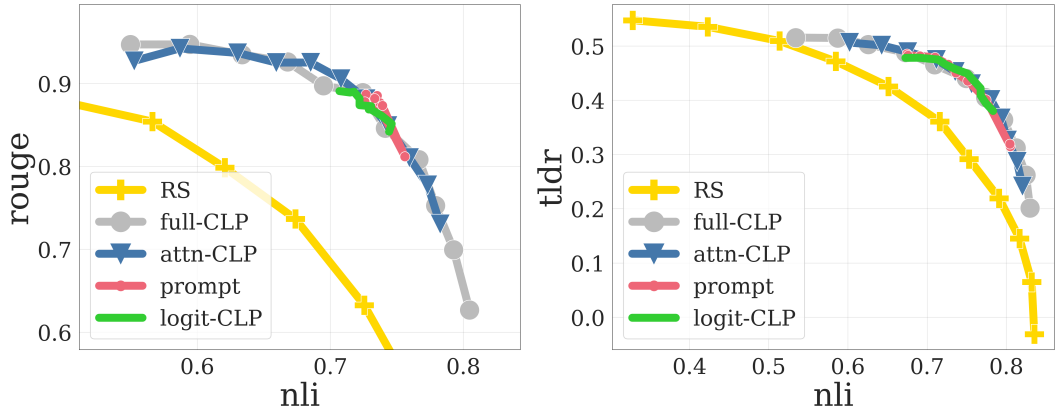
Figure 2: Pareto curves for single-reward, multi-α. Observe CLP variants (full-CLP and attn-CLP) are competitive with DeRA, a baseline that is nearly $2\times$ expensive to run at inference time.

A.1.2 Two Rewards, Fixed KL Regularizer

Here, we consider a pair of rewards with a fixed KL regularizer, to test the trade-off between both rewards. Fig. 3 shows the Pareto curves for (a) NLI v. Rouge and (b) NLI v. TLDR, where the x,y-axes are the KL-regularized rewards, *i.e.*, $x = V_{\alpha,R_1}(\pi(\cdot;\alpha,w)), y = V_{\alpha,R_2}(\pi(\cdot;\alpha,w))$, which recall is the true objective being maximized by finetuning (Eq. (1)). We see that CLP and prompting largely Pareto-dominate the baseline RS, which shows the benefits of multi-task training compared to RS's zero-shot approach. full-CLP and attn-CLP both exhibit Pareto-fronts that are more steerable (evenly spaced and spread out) than those of logit-CLP and prompting, which largely exhibit a mode-collapsed behavior. Importantly, while efficiently replicating only the attention weights, attn-CLP can Pareto-dominate the baseline RS while maintaining steerable Pareto-curves. Thus, attn-CLP offers the best trade-off between steerability and parameter count. See App. H.2 for more results.

A.1.3 Three Rewards, Fixed KL Regularizer

We now consider the three-reward setting to test the trade-off between all reward functions. Fig. 4 plots the KL-regularized value $V_{\alpha,w^\top \boldsymbol{R}}(\pi(\cdot;\alpha,w))$, normalized w.r.t. RS, for 13 different reward weightings w shown below the x-axis. At the extreme weights $w = e_i, i \in \{1,2,3\}$, RS is equivalent to single-objective finetuning and naturally outperforms CLP and prompting as expected. At the intermediate weights $w \neq e_i$, full-CLP consistently outperforms RS. We find that attn-CLP is competitive to full-CLP despite incurring only 20% of the memory overhead. Thus, akin to the



(a) NLI v. Rouge

(b) NLI v. TLDR

Figure 3: Pareto-curves for two-reward & $\alpha = 0.01$. Observe CLP variants (full-CLP and attn-CLP) offer improved spread (compared to prompting) while Pareto-dominating the Rewarded Soups (RS) baseline.

two-reward setting, attn-CLP achieves the best balance in terms of steerability and parameter count. For additional three reward results, see App. H.3.

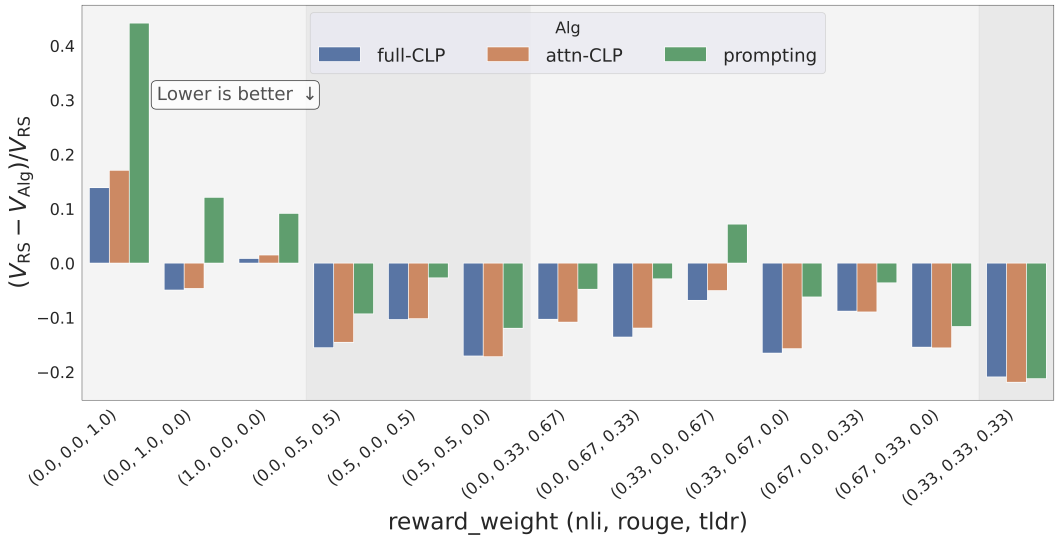


Figure 4: Barplot of $1.0 - V_{w^\top \boldsymbol{R}}(\pi_{\text{Alg}}(\cdot; w))/\widetilde{V}^{\text{RS}}_{w^\top \boldsymbol{R}}$ for three-reward experiments, where $\widetilde{V}^{\text{RS}}_{w^\top \boldsymbol{R}}$ is the KL-regularized reward of RS for weighting w. Lower is better and 0 is on-par with RS.

Summary of core benchmarking results. In terms of performance, we find that multi-task training enables CLP to improve over the zero-shot RS baseline. Importantly, we find that full-CLP and attn-CLP robustly maintain a steerable Pareto-front that is more spread out than logit-CLP and prompting baseline. In sum, attn-CLP presents a favorable trade-off in terms of Pareto-front and steerability, while using fewer parameters than existing baselines.

A.2 Ablation Studies

A.2.1 Effect of Training Iterations

Fig. 5 shows the progression of Pareto-curves over $90k$ training steps for CLP instances. full-CLP and attn-CLP are steerable after just $10k$ steps. In contrast, prompting becomes steerable at $60k$ steps; logit-CLP has a similar trend of being not as steerable at earlier training iterations. Furthermore, the

optimization dynamics of full-CLP and attn-CLP are much smoother than prompting and logit-CLP. Complete results are presented in App. H.4.

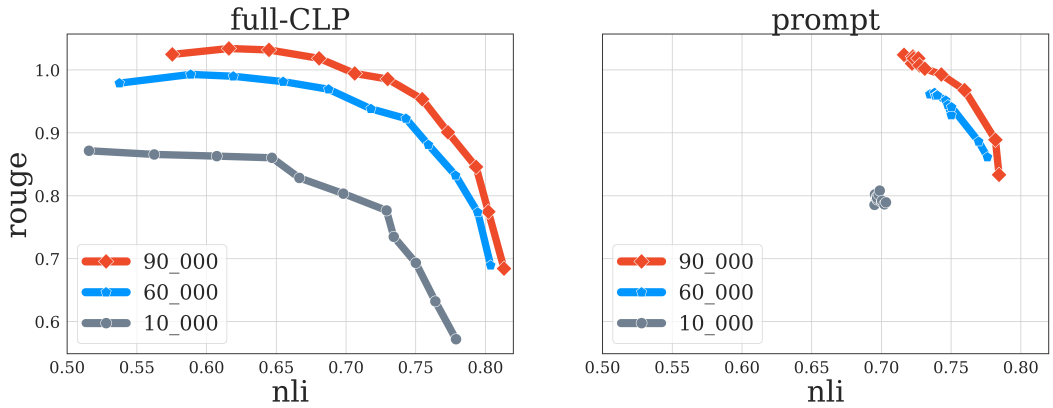
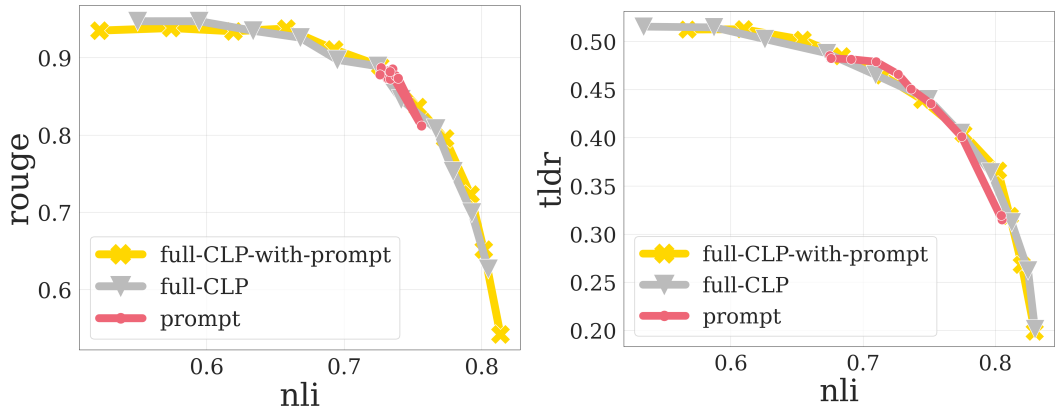


Figure 5: Pareto-curves at $10k, 60k, 90k$ training steps. Observe that prompting shows slightly improved steerability with a $3\times$ larger training budget but still isn't as steerable as full-CLP which exhibits a strong steerability even at $10k$ iterations.

A.2.2 CLP With Prompt Conditioning

Fig. 6 shows the Pareto-fronts of full-CLP, the prompting baseline, and full-CLP with prompting (CONDPROMPT = TRUE). For NLI v. Rouge, full-CLP with prompting showed a slight improvement in steerability, whereas there was little difference for NLI v. TLDR. Hence, prompt conditioning does not hurt performance but can be slightly beneficial. See App. H.5 for more results.



(a) NLI v. Rouge

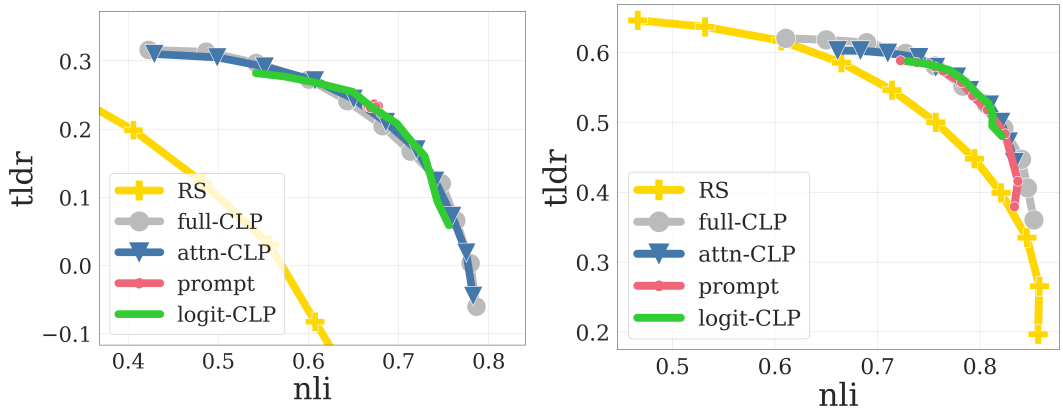
(b) NLI v. TLDR

Figure 6: Combining Prompt-based conditioning with Parameter-space condtioning for full-CLP. Observe that prompting slots in with full-CLP to produce steerable and Pareto dominating behaviors.

A.2.3 Model Size

We rerun our experiments with T5-small (60M) and T5-large (770M) to check how our findings change with different model sizes. Fig. 7 shows the NLI v. TLDR Pareto-fronts. We see that full-CLP and attn-CLP still robustly Pareto-dominate the RS baseline and maintain steerable fronts. Interestingly, prompting collapses to a point for T5-small but has much better spread for T5-large, suggesting that prompting is sensitive to model size and a larger model can improve the steerability of prompting. We provide results for other reward pairs in App. H.6.

Summary of ablations. While combining prompting with CLP did not significantly improve steerability, prompting may exhibit more steerability with larger models or training time. Across different



(a) Size: T5-small

(b) Size: T5-large

Figure 7: Ablation on model size for NLI v. TLDR. Observe that across model sizes, full-CLP and attn-CLP learn steerable (compared to prompting) and Pareto-dominating behaviors (compared to Rewarded Soups).

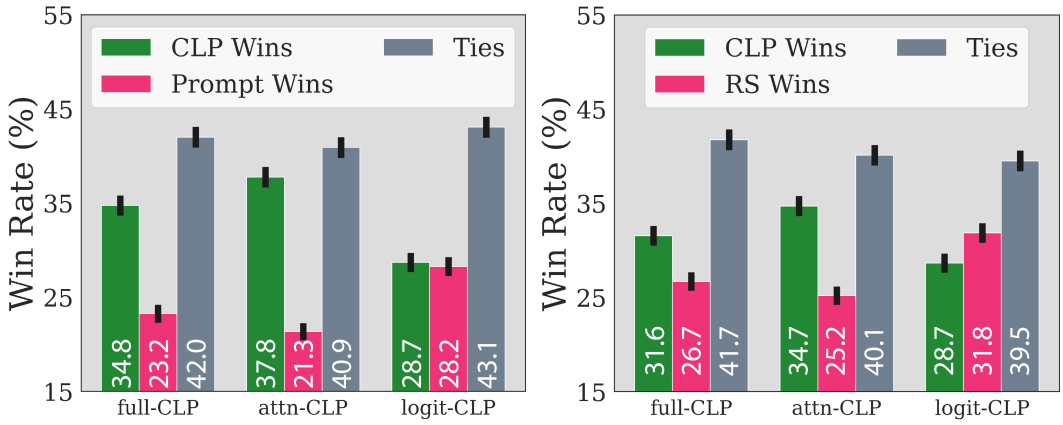
settings, full-CLP and attn-CLP consistently maintain their superior performance and steerability suggesting they are robust conditioning architectures.

A.3 Automated Evaluation

In order to understand if CLP's improved Pareto-fronts translate to improvements in generations compared to baselines (prompting and Rewarded Soups), we conduct an automated evaluation of generation quality and steerability. For this evaluation, we consider the NLI v. TLDR setup with T5-large models (from App. A.2.3) and use 2000 articles from the XSum validation set. We utilize Gemini 1.0 Ultra [33] as an automated evaluator to compare summaries from CLP instances to each baseline in terms of their conciseness and summary quality. Specifically, for each article, we sample conditioned summaries from both CLP and a baseline on weightings $w = (0.8, 0.2)$ for high NLI (resp. weightings $w = (0.2, 0.8)$ for high TLDR) and we ask the automated evaluator to compare which summary is more concise (resp. has higher quality). We permute the comparison order to account for position bias [38], marking a comparison as *consistent* if both permutations agree – please see App. I.1 for details. Then, for each article, we consider an algorithm to be the winner (*i.e.*, more steerable) if either the auto evaluator prefers its summary in both comparisons, or the auto evaluator prefers its summary in one case and was inconsistent in the other. If neither algorithm is the winner, we consider it a tie. With this setup in place, Fig. 8 shows the win-rate across different CLP variants against the two baselines. We observe that full-CLP (and attn-CLP) offers 11.6% (and 16.5%) improvement in raw win rates compared to the multi-task trained prompting baseline, and 4.9% (and 9.5%) improvements over RS. logit-CLP tends to fair comparably to prompting while being inferior to RS (dropping win rate by 3.1%). Notably, attn-CLP and full-CLP achieve the best win-rate relative to both baselines in this automatic evaluation, with attn-CLP having an additional desirable property of being more parameter-efficient. In sum, our automatic evaluation is consistent with prior Pareto-front results and validates that CLP produces higher quality multi-objective LMs with superior steerability both quantitatively and qualitatively.

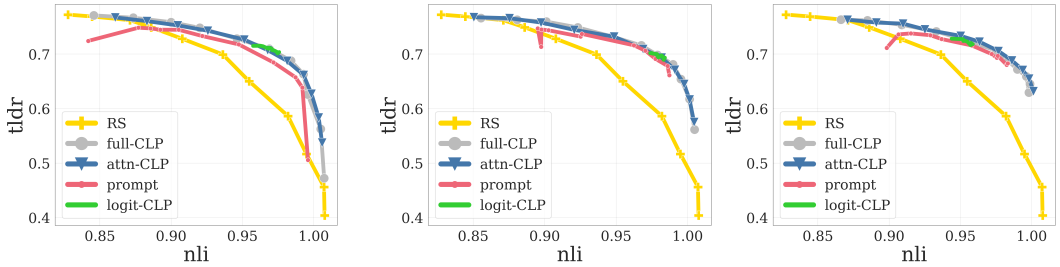
A.4 Scaling up CLP with larger policy and reward models

In the following experiment, we perform multi-reward finetuning on the TLDR dataset [34, 32] and increase the model size of both the policy and reward models. Namely, we train two reward models with T5-XXL (11B parameters), one for factuality (which we call NLI) and one for summary quality (which we call TLDR) where we adopt the same reward model training procedure as in [7]. For our policy model, we use T5-XL (3B parameters) initialized from the FLAN checkpoint [4] and all methods were trained for 6000 steps. We keep other configurations unchanged from the two-reward experiments on XSum (*e.g.*, App. A.1.2). We repeat this experiment for three sampling distributions and show these results in Fig. 9. We observe that attn-CLP and full-CLP both largely Pareto-dominate the baselines while maintaining smoothly steerable behaviors Pareto-curves. logit-CLP collapses to a



(a) CLP vs. prompting (b) CLP vs. Rewarded Soups

Figure 8: Automated Evaluation Win Rate comparison of CLP variants against prompting and Rewarded Soups baselines. CLP variants (full-CLP and attn-CLP) improve win rates by 5 to 10 (and 10 to 15) % compared to Rewarded-Soups (and prompting) baselines.



(a) $\mathcal{Q}_w \sim \text{Dir}(0.3)$ (b) $\mathcal{Q}_w \sim \text{Dir}(0.5)$ (c) $\mathcal{Q}_w \sim \text{Dir}(1.0)$

Figure 9: Results on the TLDR dataset with T5-XL policy and T5-XXL reward models. These experiments vary the sampling distribution $\mathcal{Q}_w$ of reward weightings (Dirichlet(0.3) is more narrow while Dirichlet(1.0) is uniform). We observe that the steerability of different methods improve as $\mathcal{Q}_w$ becomes narrower, though this may come at the cost of a slightly inferior Pareto front.

small region likely due to its representational bottleneck. We also observe that the steerability of the learned policy may improve as the weight sampling distribution becomes more narrow, though this may come at a cost of less Pareto-optimality. Finally, we have observed this trend repeatedly across different model sizes, datasets and sampling distributions, which gives credence to the robustness and reliability of CLP for multi-objective finetuning.

B Related Works

Approaches for multi-reward alignment (or MOFT) can be broadly classified into two categories: prompt-based and parameter-based conditioning. **Prompt-based conditioning** approaches include Personalized Soups [15] which use hand-crafted prompts for personalizing LMs based on binary weights on different rewards; CPO [11] which employs prompting within a DPO framework; RiC [42], DPA [35] use prompting within a supervised finetuning setup that differs from this paper which focuses on RL finetuning. On the **parameter-conditioning front**, Rewarded Soups (RS) [23] presents a zero-shot approach (i.e. without multi-task training) to multi-reward alignment by inference time averaging of parameters for LMs that are independently trained to optimize each of the rewards. A more recent manuscript [44] presents an approach where the reward weightings are embedded as singular values within the AdaLoRA framework [12, 43]; this can be framed as an instance of the proposed CLP framework.

With regards to KL realignment, decoding time realignment (DeRa) [17] linearly mixes logits between π_{ref} and another LM learned via SOFT with the minimum KL weight $\alpha_{\min}$. Shi et al. [30] showed

that this idea is also effective for trading off multiple rewards. Finally, model souping [40, 41], learning policy sub-spaces [9, 5], and objective weight conditioning [6] have been applied in domains beyond LMs. We leverage these advances along with multi-task training to develop steerable LMs at inference time.

C Theory for Logit Mixing and CLP

In this section, we perform a sensitivity analysis for logit mixing and derive regret bounds for its Pareto-front. While CLP uses parameter mixing instead of logit mixing, this analysis is still be instructive due to the similarity between parameter and ensemble mixing [41, 22].

C.1 Sensitivity Analysis for Logit Mixing

We focus on the "two-reward, fixed α" setting for simplicity and our analysis can be extended to the general case. For any $\lambda \in [0, 1]$, let ζ_λ be the logits of the optimal policy $\pi^\star_{\alpha,w} = \arg\max_\pi V_{\alpha,w^\top \boldsymbol{R}}(\pi)$ for weightings $w = [1-\lambda, \lambda]$. Via the analytical solution of KL-regularized reward maximization, Liu et al. [17] observed that the optimal logits at λ is expressible as mixture of the optimal logits for each individual reward, *i.e.*, $\zeta_\lambda = (1-\lambda)\zeta_0 + \lambda\zeta_1$. Thus, given *optimal* policies for R_1 & R_2, logit-mixing provides a zero-shot way to compute the optimal policy at any intermediate $\lambda \in [0, 1]$.

However, in practice, we of course are not given optimal policies and only have access to ε-approximations, so it is important to understand the sensitivity of logit-mixing. We now bound the sub-optimality of logit-mixing in terms of ε and a concentrability coefficient that measures policy coverage, defined as $C_{\pi_1,\pi_2} := \max_{x,y} \pi_2(y \mid x)/\pi_1(y \mid x)$, i.e. what is the least overlap in terms of ratio of probabilities of each policy (one per reward) over the actions (for e.g. tokens).

Theorem 1. *Suppose $\widehat{\pi}_1, \widehat{\pi}_2$ are ε-optimal policies for Eq.* (1) *with R_1, R_2, respectively. For any $\lambda \in [0, 1]$, let $\widehat{\pi}_\lambda$ be the logit mixture of $\widehat{\pi}_1$ and $\widehat{\pi}_2$. Then, the sub-optimality of $\widehat{\pi}_\lambda$ is bounded by:*

$$\mathcal{O}(((1-\lambda)C^{\lambda}_{\widehat{\pi}_2,\widehat{\pi}_1} + \lambda C^{1-\lambda}_{\widehat{\pi}_1,\widehat{\pi}_2} + p^{-2}_{\min}) \cdot \varepsilon),$$

where $p_{\min}$ is the minimum probability of $\widehat{\pi}_1, \widehat{\pi}_2$ over all input-outputs (x, y) that we care about.

The coverage terms however can be infinite if the policies $\widehat{\pi}_1, \widehat{\pi}_2$ don't cover each other. A zero-shot method will be robust to approximations when expert policies for each individual reward cover each other. But, as we show next, zero-shot approaches will fail when this coverage condition doesn't exist anymore. The full proof is in App. J.

C.2 Counterexample for zero-shot MOFT

Logit mixing cannot induce new behaviors since it can only mix behaviors from the two extremes, and so if an intermediate weighting requires a new behavior, logit mixing provably fails. Consider a toy problem with one context and three possible outputs y_1, y_2, y_3 with rewards $R_1(x, \cdot) = (1, 0, 0.75), R_2(x, \cdot) = (0, 1, 0.75)$. The optimal policies for R_1, R_2 (with $\alpha = 0$) are $\pi^\star_1(x) = (1, 0, 0)$ and $\pi^\star_2(x) = (0, 1, 0)$. However, $\pi^\star_{0.5} = (0, 0, 1)$, which is a *qualitatively new* behavior that cannot arise from zero-shot logit mixing thus being a failure case for zero-shot logit mixing. In appendix Fig. 21, we show that the RS baseline, which is zero-shot empirically fails to learn the Pareto-optimal policy in this example, while CLP which uses multi-task training succeeds.

D Additional Algorithmic Details

D.1 Illustration of CLP

The following supplemental figure illustrates one training round of CLP (Alg. 1).

D.2 Prompt-based Conditioning

We use prompt-conditioning only to trade-off between rewards (*i.e.*, not for trading off KL). Given an input prompt x and reward weightings w, we propose to concatenate w to the prompt by prefixing x

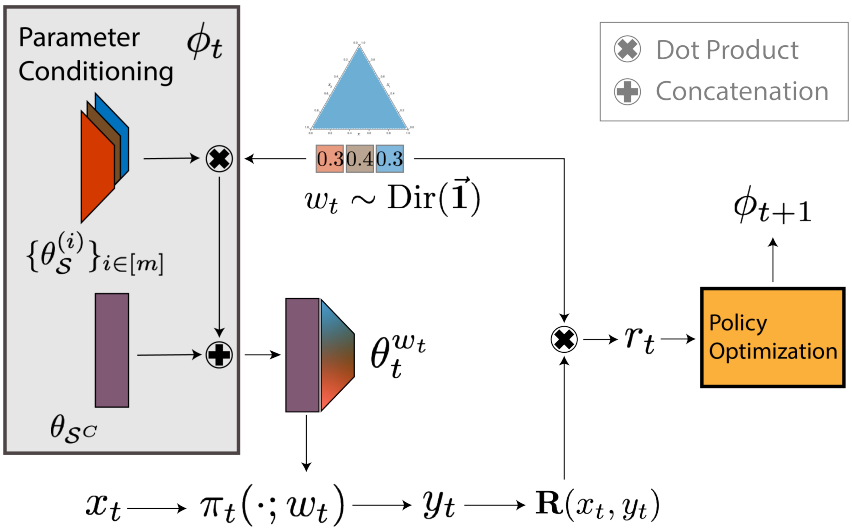


Figure 10: CLP samples diverse reward weightings w_t at training time to facilitate multi-task learning and uses parameter-space conditioning to compute a conditioned policy $\pi_t(\cdot; w_t)$.

with a formatted string. For example if R_1 is NLI (a measure of factuality) and R_2 is Rouge, and $w = [0.4, 0.6]$, we used the the prefix string

$p =$ ``factuality: 0.4 0.4 0.4 0.4 0.4; rouge: 0.6 0.6 0.6 0.6 0.6.''

We repeat the weights multiple times such that the LM can better pay attention to it and we found 5 repetitions to be reasonable. We found that repeating weights multiple times is helpful but the benefits are not monotonic with more repetitions.

D.3 Analysis of CLP Gradients

While computing CLP's gradient g_t in Line 7 is easy with autodiff libraries, it is insightful to analyze how gradient propagates through CLP's parameter conditioning step. Let $\gamma_t = r_t \nabla_{\theta_t^{\alpha_t, w_t}} \log \pi_t(y_t \mid x_t; \alpha_t, w_t)$ denote the gradient w.r.t. LM parameters evaluated at $\theta_t^{\alpha_t, w_t} = \theta_{t,\mathcal{S}^C} \oplus \theta_{t,\mathcal{S}}^{\alpha_t, w_t}$, where $\theta_{t,\mathcal{S}^C}$ is the unconditioned parameter and $\theta_{t,\mathcal{S}}^{\alpha_t, w_t}$ is the (α_t, w_t)-conditioned $\mathcal{S}$-parameter at round t. We decompose $g_t = (g_{t,\mathcal{S}^C}, \{g_{t,\mathcal{S}}^{(i)}\}_{i\in[m]})$ where $g_{t,\mathcal{S}^C} = \gamma_t[\mathcal{S}^C]$, and $g_{t,\mathcal{S}}^{(i)} = (1 - f_{\mathsf{mix}}(\alpha_t)) \cdot w_t[i] \cdot \gamma_t[\mathcal{S}]$ for all $i \in [m]$. Thus, the gradient for CLP's unconditioned parameters is exactly the LM gradient at $\mathcal{S}^C$, and the gradient for CLP's conditioned parameters point in the direction of the LM gradient at $\mathcal{S}$, and are scaled by the reward weights w_t and $1 - f_{\mathsf{mix}}(\alpha_t)$. *E.g.*, i-th conditioned parameter $\theta_{t,\mathcal{S}}^{(i)}$ is updated more if R_i has high weight (*i.e.*, $w_t[i]$ is large); conversely, it is not updated at all if R_i has zero weight. The scaling by $1 - f_{\mathsf{mix}}(\alpha_t)$ implies that the conditioning parameters are updated more when α_t is small and close to $\alpha_{\min}$, and updated less when α_t is close to 1.

D.4 Weightings Distribution and KL-Mixing

We suggest a default sampling distribution $\mathcal{Q}$ for the reward and KL weightings. We also suggest a compatible KL-mixing function f_{mix}, which transforms α before applying parameter-mixing in the conditioning mechanism.

First, a natural sampling distribution for reward weights $w \in \Delta_m$ is the Dirichlet distribution $\mathcal{Q}_w = \mathrm{Dir}(\boldsymbol{\beta})$ with parameters $\boldsymbol{\beta} \in \mathbb{R}_+^m$. Concentrated sampling (*e.g.*, large $\boldsymbol{\beta}[i] \to \infty$ values) can more easily cause unsteerable "mode-collapsed" behavior. From our experience, a good default is uniform Dirichlet with $\boldsymbol{\beta} = (1, 1, \ldots, 1)$ which consistently led to steerable behaviors in full-CLP and attn-CLP.

Next, we discuss how to set the KL-mixer map $f_{\mathsf{mix}}(\alpha)$. As motivation, recall the following theorem about KL-realignment from Liu et al. [17]. Let $\zeta_\alpha(x) \in \mathbb{R}^{|\mathcal{Y}|}$ be the logits of the optimal policy

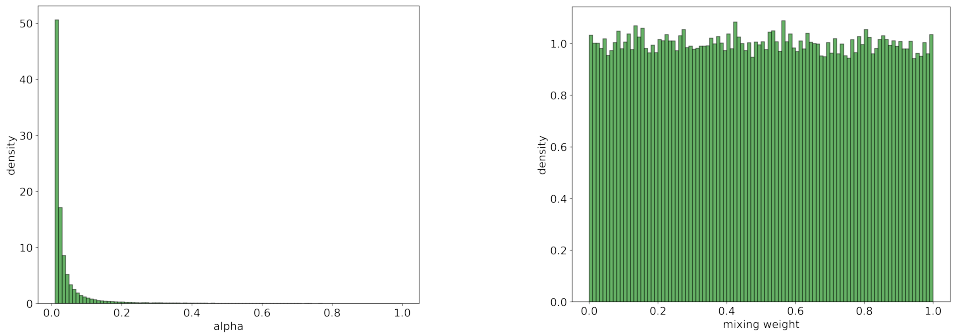
with KL-weight α. Then, $\zeta_\alpha(x) = \zeta_{\text{ref}}(x) + (1-\alpha)R(x,\cdot)/\alpha$, where ζ_{ref} are the logits of $\pi_{\text{ref}}(x)$. Rearranging terms, for all $\alpha \in [\alpha_{\min}, 1]$, we have $\zeta_\alpha = (1-\beta)\zeta_{\alpha_{\min}} + \beta \cdot \zeta_{\text{ref}}$ where $\beta = f_{\mathsf{mix}}(\alpha)$ and

$$f_{\mathsf{mix}}(\alpha) = \frac{\alpha - \alpha_{\min}}{\alpha(1-\alpha_{\min})}. \tag{4}$$

Note that f_{mix} satisfies $f_{\mathsf{mix}}(1) = 1$, so when $\alpha = 1$ we use purely $\theta_{\text{ref}}[\mathcal{S}]$ and do not mix in any learned $\theta_{\mathcal{S}}$; this makes intuitive as the minimizer of KL is indeed π_{ref}. Conversely, $f_{\mathsf{mix}}(\alpha_{\min}) = 0$, meaning that θ_{ref} is not mixed in when $\alpha = \alpha_{\min}$. Thus, since f_{mix} is the right mixing function for logits, we adopt this f_{mix} for general parameter mixing in CLP.

To sample KL weights $\alpha_t \sim \mathcal{Q}_\alpha$, we suggest to use the inverse CDF method with $f^{-1}_{\mathsf{mix}}(u) = \frac{\alpha_{\min}}{\alpha_{\min} \cdot u + (1-u)}$, *i.e.*, $\alpha_t = f^{-1}_{\mathsf{mix}}(U)$ with $U \sim \text{unif}[0,1]$. This ensures that the KL-mixing weights are uniformly distributed, *i.e.*, $f_{\mathsf{mix}}(\alpha_t) \sim \text{unif}[0,1]$. Since the median is $\frac{2\alpha_{\min}}{\alpha_{\min}+1}$, this distribution also places higher probabilities on smaller α values. This is desirable since small α corresponds to larger learning signals (large α just forces the model to be close to θ_{ref}). In App. F.5, we show samples from $\alpha_t \sim \mathcal{Q}_\alpha$ and $f_{\mathsf{mix}}(\alpha_t)$. Finally, we note that w or α can be fixed to a specific value if one cares about only varying the other.

D.5 KL weight distribution and mixing function



(a) Distribution of $\alpha_t \sim \mathcal{Q}_\alpha$.

(b) distribution of $f_{\mathsf{mix}}(\alpha_t) \sim \text{unif}[0,1]$.

Figure 11: Distributions of $\alpha_t \sim \mathcal{Q}_\alpha$ and $f_{\mathsf{mix}}(\alpha_t)$ with $\mathcal{Q}_\alpha, f_{\mathsf{mix}}$ defined in App. F.4.

E Experiment Details

Reward and normalization. For Rouge, we use the LSum variant [16] throughout the paper. Both the NLI and TLDR models were trained with T5-large [21]. We normalize rewards such that they all approximately lie in the range $[0,1]$ to ensure they are on a comparable scale. This is important as otherwise it is easy for a reward with large scale to dominate the objective. In particular, we linearly mapped the following ranges for each reward to $[0,1]$: $\{R_{\text{nli}} : (-7,0), R_{\text{rouge}} : (19,29), R_{\text{tldr}} : (-1.6,2)\}$.

KL regularizer. In the single-reward, multi-KL experiments, we use $\alpha_{\min} = 0.01$, and in the two-reward and three-reward experiments with fixed KL regularizer, we use $\alpha = 0.01$. This α value is chosen based on observing that, under our normalized reward setup, it is high enough to prevent reward hacking and low enough to exhibit interesting qualitative differences from the SFT π_{ref}.

Training. To train CLP and all our baselines, we use a batch size of 32. For CLP, we sample fresh reward or KL weightings per batch, instead of per example, as it is more efficient to only condition once per batch. We also do this for prompting for consistency. We use the T5X implementation [24] and the Adafactor optimizer [29]. In terms of training budget, we ran $10,000$ iterations for the single-reward, multi-KL experiments, and we ran $30,000$ iterations for the multi-reward, fixed-KL experiments. For the RS baseline, the training steps were divided evenly between the rewards; *e.g.*, in the two-reward setting, RS learns two LMs each trained for $15,000$ and initialized from π_{ref}. Thus CLP and RS are trained with the same number of iterations. For all results, we train on the XSum training set and report the reward functions and/or qualitative generations on the *validation set*.

Hyper-Parameter	Value
Model Family Details	
Policy/Value Model	T5-Base (220M) (ablations: T5-small (60M) and T5-large (770M))
Reward Models	Rouge LSum NLI/TLDR (learnt - T5-large)
Reward Normalization	$R_{\text{nli}} : (-7, 0) \to (0, 1)$ $R_{\text{tldr}} : (-1.6, 2) \to (0, 1)$ $R_{\text{rouge}} : (19, 29) \to (0, 1)$
Tokenizer	Sentence Piece Tokenizer (32k vocabulary size)
Code	T5X [24]
Computing Infra	TPU-v5e chips
Experiment Time	Using 16 TPU-v5e chips, our T5-base runs that includes $30k$ training steps and evaluations every $10k$ steps took 8 hours. Note that each evaluation over the validation set takes roughly 1 hour.
Policy Optimization Hyper-parameters	
Batch Size	32
Policy Optimizer	Adafactor [29] learning rate $3e-5$
Value Optimizer	Adafactor [29] learning rate $1e-4$
CLP/Multi-task training Hyper-parameters	
KL-Strength α	Multi-Reward Single KL: 0.01 Also tried 0.1 in ablations Single-Reward Multi KL: $\alpha_{\min} = 0.01$
Training Budget	2 Reward, Single KL: Rewarded Soups: $15,000$ ($\times 2$, one per reward) CLP variants: $30,000$. Single Reward, Multi KL:TODO:FILL 3 Reward, Single KL: Rewarded Soups: $10,000$ ($\times 3$, one per reward) CLP variants: $30,000$.
Sampling Distribution $\mathcal{D}$	Dirichlet(1.0) for main results Also Dirichlet(0.3) in ablations.

E.1 Policy Optimization

Recall that the policy optimization step in Line 7 can be implemented with any RL method (policy gradients, actor-critic etc.). This paper uses REINFORCE with a control variate (i.e., baseline) which uses fewer hyper-parameters and lighter implementation than PPO [28], and has been successfully used in prior works [27, 13]. To train the control variate, we maintain a value network of the same architecture as the policy network where the prediction is the logit value for some fixed token id. For example, in a problem with m rewards, we use the first m token ids where the i-th token's logit serves as the value function for the i-th reward, *i.e.*, $\widehat{V}_i(x) \approx \mathbb{E}_{w\sim\mathcal{Q}, y\sim\pi(\cdot;w)} R_i(x, y)$. This value network is trained by minimizing MSE, which is standard in RL. Then, these predictions are linearly combined with weightings to compute the value estimate for the weighted reward, *i.e.*, $\widehat{V}_t = \sum_i w_i \cdot \widehat{V}_i(x_t)$. The advantage is computed as usual $A_t = r_t - \widehat{V}_t$ and is batch-normalized before being multiplied with the $\nabla_{\phi_t} \log \pi_t$ term.

F Additional Algorithmic Details

F.1 Illustration of CLP

The following supplemental figure illustrates one training round of CLP (Alg. 1).

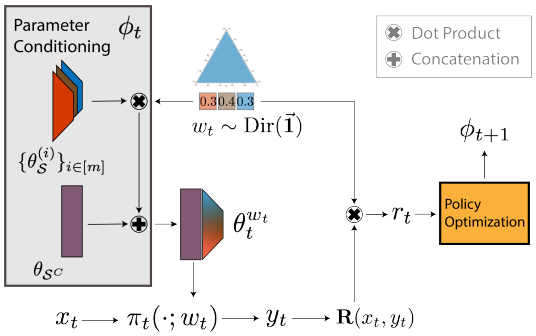


Figure 12: CLP samples diverse reward weightings w_t at training time to facilitate multi-task learning and uses parameter-space conditioning to compute a conditioned policy $\pi_t(\cdot; w_t)$.

F.2 Prompt-based Conditioning

We use prompt-conditioning only to trade-off between rewards (*i.e.*, not for trading off KL). Given an input prompt x and reward weightings w, we propose to concatenate w to the prompt by prefixing x with a formatted string. For example if R_1 is NLI (a measure of factuality) and R_2 is Rouge, and $w = [0.4, 0.6]$, we used the the prefix string

$p =$ ```factuality: 0.4 0.4 0.4 0.4 0.4; rouge: 0.6 0.6 0.6 0.6 0.6.''.

We repeat the weights multiple times such that the LM can better pay attention to it and we found 5 repetitions to be reasonable. We found that repeating weights multiple times is helpful but the benefits are not monotonic with more repetitions.

F.3 Analysis of CLP Gradients

While computing CLP's gradient g_t in Line 7 is easy with autodiff libraries, it is insightful to analyze how gradient propagates through CLP's parameter conditioning step. Let $\gamma_t = r_t \nabla_{\theta_t^{\alpha_t, w_t}} \log \pi_t(y_t \mid x_t; \alpha_t, w_t)$ denote the gradient w.r.t. LM parameters evaluated at $\theta_t^{\alpha_t, w_t} = \theta_{t,\mathcal{S}^C} \oplus \theta_{t,\mathcal{S}}^{\alpha_t, w_t}$, where $\theta_{t,\mathcal{S}^C}$ is the unconditioned parameter and $\theta_{t,\mathcal{S}}^{\alpha_t, w_t}$ is the (α_t, w_t)-conditioned $\mathcal{S}$-parameter at round t. We decompose $g_t = (g_{t,\mathcal{S}^C}, \{g_{t,\mathcal{S}}^{(i)}\}_{i\in[m]})$ where $g_{t,\mathcal{S}^C} = \gamma_t[\mathcal{S}^C]$, and $g_{t,\mathcal{S}}^{(i)} = (1 - f_{\mathsf{mix}}(\alpha_t)) \cdot w_t[i] \cdot \gamma_t[\mathcal{S}]$ for all $i \in [m]$. Thus, the gradient for CLP's unconditioned parameters is exactly the LM gradient at $\mathcal{S}^C$, and the gradient for CLP's conditioned parameters point in the direction of the LM gradient at $\mathcal{S}$, and are scaled by the reward weights w_t and $1 - f_{\mathsf{mix}}(\alpha_t)$. *E.g.*, i-th conditioned parameter $\theta_{t,\mathcal{S}}^{(i)}$ is updated more if R_i has high weight (*i.e.*, $w_t[i]$ is large); conversely, it is not updated at all if R_i has zero weight. The scaling by $1 - f_{\mathsf{mix}}(\alpha_t)$ implies that the conditioning parameters are updated more when α_t is small and close to $\alpha_{\min}$, and updated less when α_t is close to 1.

F.4 Weightings Distribution and KL-Mixing

We suggest a default sampling distribution $\mathcal{Q}$ for the reward and KL weightings. We also suggest a compatible KL-mixing function f_{mix}, which transforms α before applying parameter-mixing in the conditioning mechanism.

First, a natural sampling distribution for reward weights $w \in \Delta_m$ is the Dirichlet distribution $\mathcal{Q}_w = \mathrm{Dir}(\boldsymbol{\beta})$ with parameters $\boldsymbol{\beta} \in \mathbb{R}_+^m$. Concentrated sampling (*e.g.*, large $\boldsymbol{\beta}[i] \to \infty$ values) can more easily cause unsteerable "mode-collapsed" behavior. From our experience, a good default is uniform Dirichlet with $\boldsymbol{\beta} = (1, 1, \ldots, 1)$ which consistently led to steerable behaviors in full-CLP and attn-CLP.

Next, we discuss how to set the KL-mixer map $f_{\mathsf{mix}}(\alpha)$. As motivation, recall the following theorem about KL-realignment from Liu et al. [17]. Let $\zeta_\alpha(x) \in \mathbb{R}^{|\mathcal{Y}|}$ be the logits of the optimal policy with KL-weight α. Then, $\zeta_\alpha(x) = \zeta_{\text{ref}}(x) + (1 - \alpha)R(x, \cdot)/\alpha$, where ζ_{ref} are the logits of $\pi_{\text{ref}}(x)$. Rearranging terms, for all $\alpha \in [\alpha_{\min}, 1]$, we have $\zeta_\alpha = (1 - \beta)\zeta_{\alpha_{\min}} + \beta \cdot \zeta_{\text{ref}}$ where $\beta = f_{\mathsf{mix}}(\alpha)$

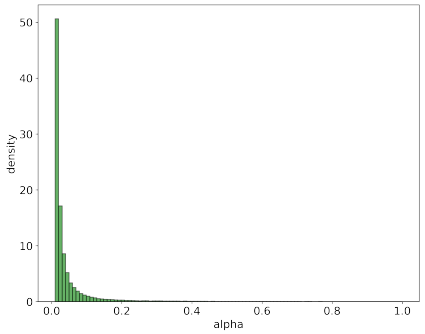
and

$$f_{\mathsf{mix}}(\alpha) = \frac{\alpha - \alpha_{\min}}{\alpha(1 - \alpha_{\min})}. \tag{5}$$

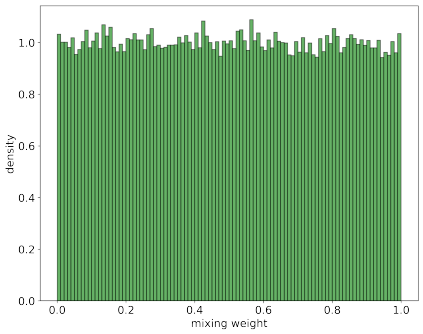
Note that f_{mix} satisfies $f_{\mathsf{mix}}(1) = 1$, so when $\alpha = 1$ we use purely $\theta_{\text{ref}}[\mathcal{S}]$ and do not mix in any learned $\theta_{\mathcal{S}}$; this makes intuitive as the minimizer of KL is indeed π_{ref}. Conversely, $f_{\mathsf{mix}}(\alpha_{\min}) = 0$, meaning that θ_{ref} is not mixed in when $\alpha = \alpha_{\min}$. Thus, since f_{mix} is the right mixing function for logits, we adopt this f_{mix} for general parameter mixing in CLP.

To sample KL weights $\alpha_t \sim \mathcal{Q}_\alpha$, we suggest to use the inverse CDF method with $f^{-1}_{\mathsf{mix}}(u) = \frac{\alpha_{\min}}{\alpha_{\min} \cdot u + (1-u)}$, *i.e.*, $\alpha_t = f^{-1}_{\mathsf{mix}}(U)$ with $U \sim \text{unif}[0, 1]$. This ensures that the KL-mixing weights are uniformly distributed, *i.e.*, $f_{\mathsf{mix}}(\alpha_t) \sim \text{unif}[0, 1]$. Since the median is $\frac{2\alpha_{\min}}{\alpha_{\min}+1}$, this distribution also places higher probabilities on smaller α values. This is desirable since small α corresponds to larger learning signals (large α just forces the model to be close to θ_{ref}). In App. F.5, we show samples from $\alpha_t \sim \mathcal{Q}_\alpha$ and $f_{\mathsf{mix}}(\alpha_t)$. Finally, we note that w or α can be fixed to a specific value if one cares about only varying the other.

F.5 KL weight distribution and mixing function



(a) Distribution of $\alpha_t \sim \mathcal{Q}_\alpha$.



(b) distribution of $f_{\mathsf{mix}}(\alpha_t) \sim \text{unif}[0, 1]$.

Figure 13: Distributions of $\alpha_t \sim \mathcal{Q}_\alpha$ and $f_{\mathsf{mix}}(\alpha_t)$ with $\mathcal{Q}_\alpha$, f_{mix} defined in App. F.4.

G Experiment Details

Reward and normalization. For Rouge, we use the LSum variant [16] throughout the paper. Both the NLI and TLDR models were trained with T5-large [21]. We normalize rewards such that they all approximately lie in the range $[0, 1]$ to ensure they are on a comparable scale. This is important as otherwise it is easy for a reward with large scale to dominate the objective. In particular, we linearly mapped the following ranges for each reward to $[0, 1]$: $\{R_{\text{nli}} : (-7, 0), R_{\text{rouge}} : (19, 29), R_{\text{tldr}} : (-1.6, 2)\}$.

KL regularizer. In the single-reward, multi-KL experiments, we use $\alpha_{\min} = 0.01$, and in the two-reward and three-reward experiments with fixed KL regularizer, we use $\alpha = 0.01$. This α value is chosen based on observing that, under our normalized reward setup, it is high enough to prevent reward hacking and low enough to exhibit interesting qualitative differences from the SFT π_{ref}.

Training. To train CLP and all our baselines, we use a batch size of 32. For CLP, we sample fresh reward or KL weightings per batch, instead of per example, as it is more efficient to only condition once per batch. We also do this for prompting for consistency. We use the T5X implementation [24] and the Adafactor optimizer [29]. In terms of training budget, we ran $10,000$ iterations for the single-reward, multi-KL experiments, and we ran $30,000$ iterations for the multi-reward, fixed-KL experiments. For the RS baseline, the training steps were divided evenly between the rewards; *e.g.*, in the two-reward setting, RS learns two LMs each trained for $15,000$ and initialized from π_{ref}. Thus CLP and RS are trained with the same number of iterations. For all results, we train on the XSum training set and report the reward functions and/or qualitative generations on the *validation set*.

G.1 Policy Optimization

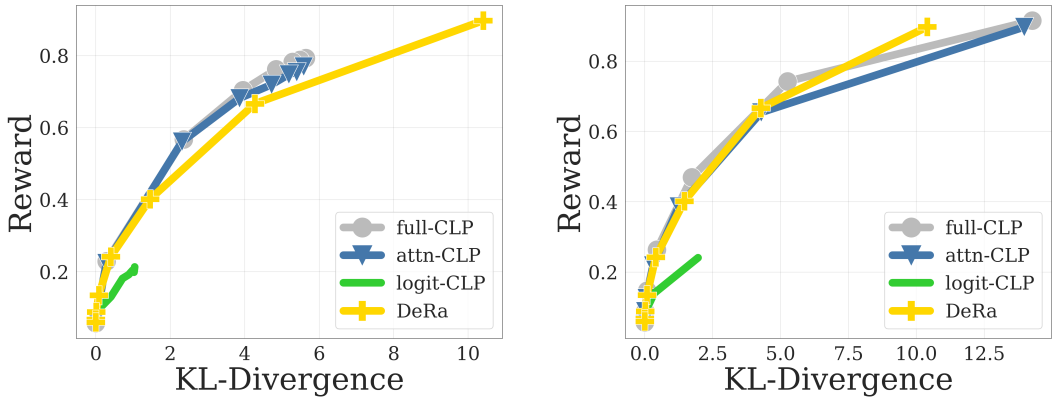
Recall that the policy optimization step in Line 7 can be implemented with any RL method (policy gradients, actor-critic etc.). This paper uses REINFORCE with a control variate (i.e., baseline) which uses fewer hyper-parameters and lighter implementation than PPO [28], and has been successfully used in prior works [27, 13]. To train the control variate, we maintain a value network of the same architecture as the policy network where the prediction is the logit value for some fixed token id. For example, in a problem with m rewards, we use the first m token ids where the i-th token's logit serves as the value function for the i-th reward, *i.e.*, $\widehat{V}_i(x) \approx \mathbb{E}_{w\sim\mathcal{Q}, y\sim\pi(\cdot;w)} R_i(x, y)$. This value network is trained by minimizing MSE, which is standard in RL. Then, these predictions are linearly combined with weightings to compute the value estimate for the weighted reward, *i.e.*, $\widehat{V}_t = \sum_i w_i \cdot \widehat{V}_i(x_t)$. The advantage is computed as usual $A_t = r_t - \widehat{V}_t$ and is batch-normalized before being multiplied with the $\nabla_{\phi_t} \log \pi_t$ term.

Hyper-Parameter	Value
Model Family Details	
Policy/Value Model	T5-Base (220M) (ablations: T5-small (60M) and T5-large (770M))
Reward Models	Rouge LSum NLI/TLDR (learnt - T5-large)
Reward Normalization	$R_{\text{nli}} : (-7, 0) \to (0, 1)$ $R_{\text{tldr}} : (-1.6, 2) \to (0, 1)$ $R_{\text{rouge}} : (19, 29) \to (0, 1)$
Tokenizer	Sentence Piece Tokenizer (32k vocabulary size)
Code	T5X [24]
Computing Infra	TPU-v5e chips
Experiment Time	Using 16 TPU-v5e chips, our T5-base runs that includes $30k$ training steps and evaluations every $10k$ steps took 8 hours. Note that each evaluation over the validation set takes roughly 1 hour.
Policy Optimization Hyper-parameters	
Batch Size	32
Policy Optimizer	Adafactor [29] learning rate $3e-5$
Value Optimizer	Adafactor [29] learning rate $1e-4$
CLP/Multi-task training Hyper-parameters	
KL-Strength α	Multi-Reward Single KL: 0.01 Also tried 0.1 in ablations Single-Reward Multi KL: $\alpha_{\min} = 0.01$
Training Budget	2 Reward, Single KL: Rewarded Soups: $15,000$ ($\times 2$, one per reward) CLP variants: $30,000$. Single Reward, Multi KL:TODO:FILL 3 Reward, Single KL: Rewarded Soups: $10,000$ ($\times 3$, one per reward) CLP variants: $30,000$.
Sampling Distribution $\mathcal{D}$	Dirichlet(1.0) for main results Also Dirichlet(0.3) in ablations.

H Additional Experimental Results

H.1 Additional Multi-KL Results

This section focuses on the single-reward, multi-KL setup as in App. A.1.1. We report CLP results with two choices of the KL-mixing function f_{mix}. First, linear mixing function is $f_{\text{mix}}(x) = x$. Second, softmdp mixing function is defined in Eq. (5) of App. F.4, which is more principled from the perspective of DeRa [17]. We see that softmdp mixing indeed yields better steerability for CLP.



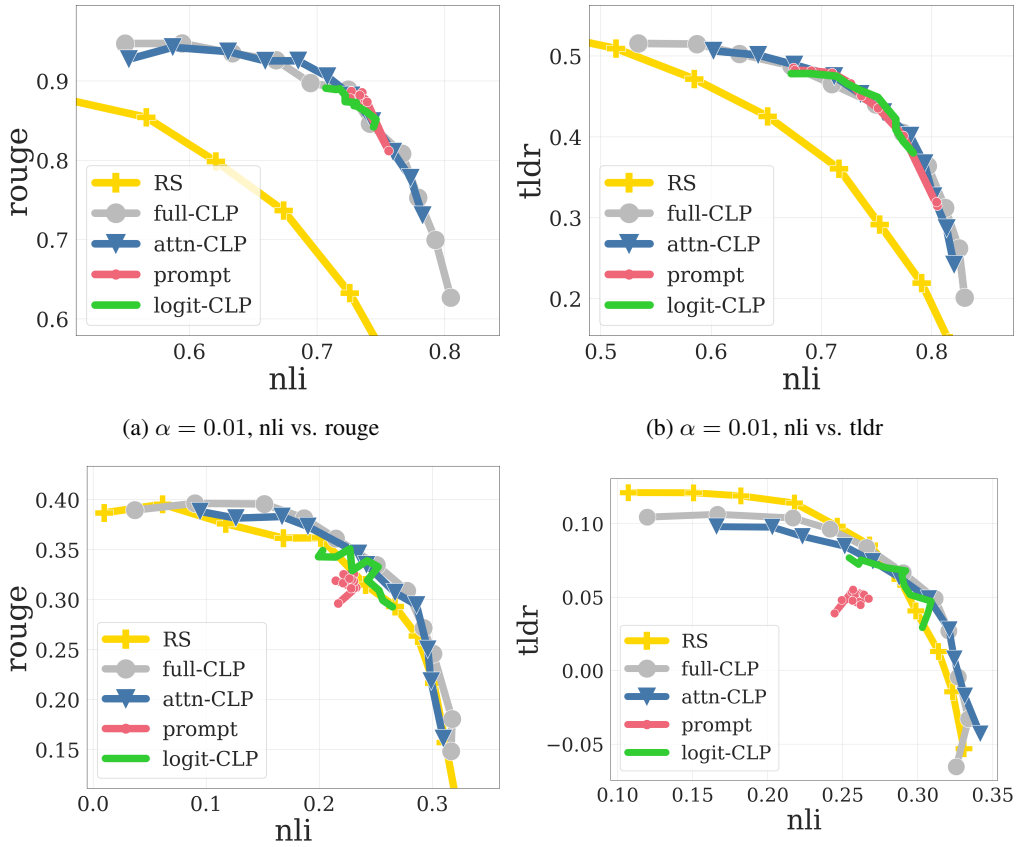
(a) linear mixing

(b) softmdp mixing

Figure 14: Ablation on the KL-weight mixing function f_{mix}. Linear mixing sets $f_{\text{mix}}(\alpha) = \alpha$. Softmdp mixing is the one described in App. F.4.

H.2 Additional Two-Reward Results

Results for (1) NLI v. Rouge, (2) NLI v. TLDR for fixed $\alpha \in \{0.01, 0.1\}$.



(a) $\alpha = 0.01$, nli vs. rouge

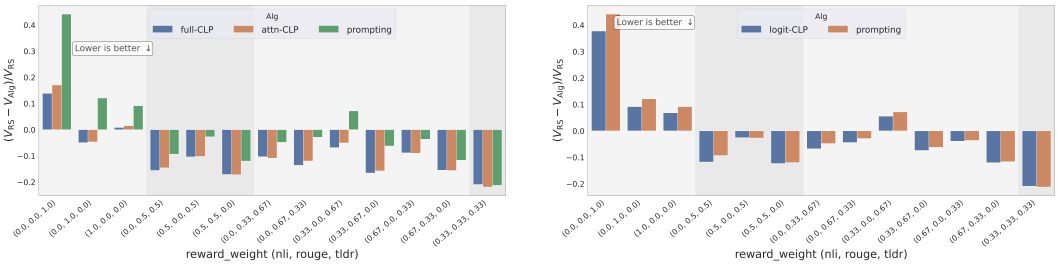
(b) $\alpha = 0.01$, nli vs. tldr

(c) $\alpha = 0.1$, nli vs. rouge

(d) $\alpha = 0.1$, nli vs. tldr

Figure 15: Plots comparing CLP instances (full-CLP, attn-CLP, logit-CLP) against Rewarded Soups (RS) [23] and prompting in the two-reward experiments.

H.3 Additional Three-Reward Results



(a) full-CLP, attn-CLP, prompting

(b) logit-CLP, prompting

Figure 16: Plots comparing CLP instances (full-CLP, attn-CLP, logit-CLP) against Rewarded Soups (RS) [23] and prompting in the three-reward experiments.

H.4 Effect of Training Iterations

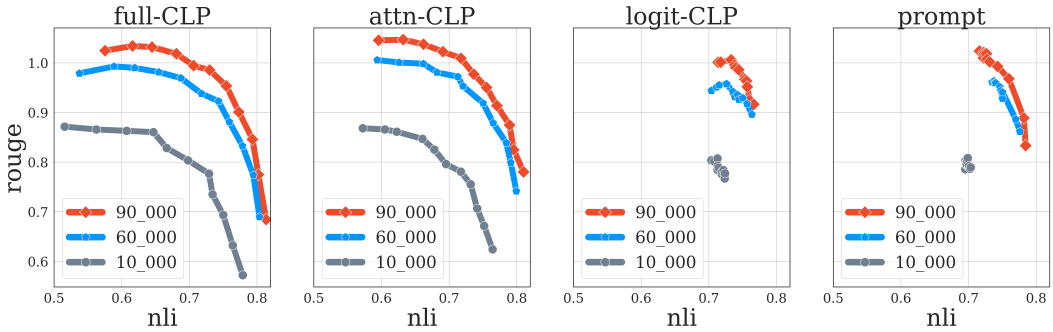
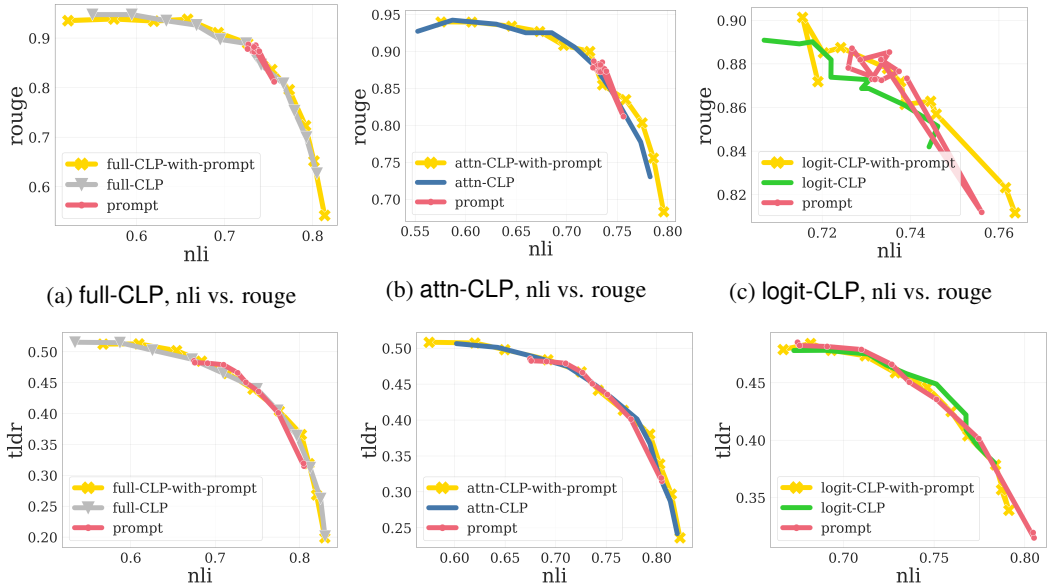


Figure 17: Effect of Training Iterations. Observe that CLP variants such as full-CLP and attn-CLP tend to present reasonably strong spread out behaviors even with smaller training budgets. prompting and logit-CLP tend to start spreading out just a little after running the expensive RLHF training procedure for $3\times$ – note that the spread of these methods is still far inferior to ones obtained by full-CLP and attn-CLP.

H.5 Combining Prompt and Parameter Conditioning



(a) full-CLP, nli vs. rouge (b) attn-CLP, nli vs. rouge (c) logit-CLP, nli vs. rouge

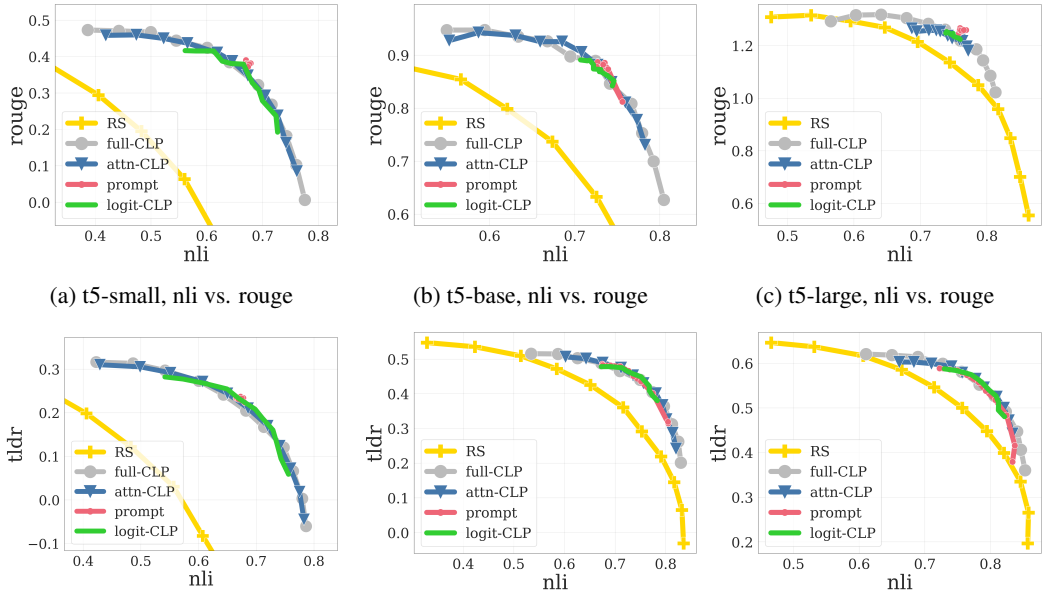
(d) full-CLP, nli vs. tldr (e) attn-CLP, nli vs. tldr (f) logit-CLP, nli vs. tldr

Figure 18: Ablation study that enables prompting for different CLP instances, across different experiments with a pair of reward functions.

In the above figures, we see that prompt-conditioning does not always contribute to more steerability in CLP, but it also does not hurt the steerability. We remark that the aberration in Fig. 18(c) is simply a zoomed in version of Fig. 18(b), showing that logit-CLP and prompting are not very steerable in this example.

H.6 Ablation on Model Size

We present results across two paired-reward setups (NLI vs. Rouge, NLI vs. TLDR) and three model sizes (T5-small, T5-base, T5-large). In T5-large which has 24 attention layers, we only mix the first 12 layers of the attention parameters for attn-CLP. We believe that mixing all attention layers can only further improve steerability. For the T5-large model, the first 12 attention layer parameters account for around 20% of the total parameter count.

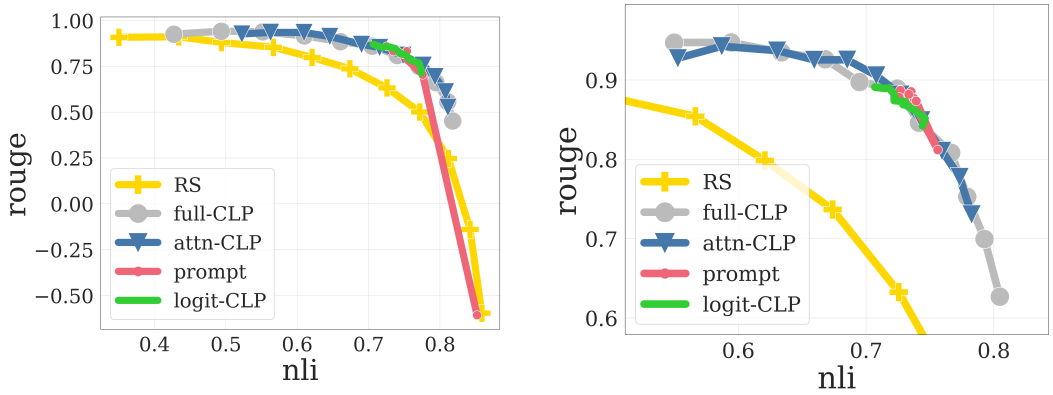


(a) t5-small, nli vs. rouge (b) t5-base, nli vs. rouge (c) t5-large, nli vs. rouge

(d) t5-small, nli vs. tldr (e) t5-base, nli vs. tldr (f) t5-large, nli vs. tldr

Figure 19: Ablation study involving different model sizes for parameterizing the steerable policy with the t5 family of models. Observe that CLP variants tend to perform in a predictable and robust manner across these model sizes and choices of rewards. Other distinct observations include (a) prompting behaves in rather unpredictable manners (collapsing and spreading out in rather non-transparent manners, see nli vs. rouge plots in 2nd row), (b) Rewarded Soups show predictable behavior that attempts to catch up with CLP variants but still retaining a healthy gap in terms of Pareto front even with t5-large model sizes.

H.7 Ablation on Sampling Strategy



(a) nli vs. rouge, sampling$\sim$ Dirichlet(0.3)

(b) nli vs. rouge, sampling$\sim$ Dirichlet(1.0)

Figure 20: Ablation on the weight sampling distribution. Setting is NLI v. Rouge with $\alpha = 0.01$. Observe that the behaviors of CLP variants tend to offer broadly consistent behaviors regardless of the choice of the reward weighting sampling distributions $\mathcal{D}$ while prompting tends to be sensitive to this choice – finding an reasonable sampling strategy for a new problem can be tricky to solve when employing prompting, while CLP variants remain robust to this choice.

I Additional Qualitative Results

I.1 Details of Automatic Evaluation

We evaluate generations from CLP variants (full-CLP, attn-CLP, logit-CLP) against both prompting and Rewarded Soups baselines. For this, we pick the NLI vs. TLDR setup, use the T5-large models trained in App. A.2.3) and evaluate on 2000 prompts from the XSum validation set.

We employ the following prompt to compare which one of a pair of summaries is more concise while being reasonable in quality in terms of capturing relevant information in the article.

> You are an expert summary rater. Given an ARTICLE and two summaries SUMMARY1 and SUMMARY2, compare SUMMARY1 and SUMMARY2 based on how **concise the summary is in capturing relevant information** in the ARTICLE.
>
> Output SUMMARY1 if it **more concisely captures relevant information** in the ARTICLE compared to SUMMARY2. Alternatively, output SUMMARY2 if it **more concisely captures relevant information** in the ARTICLE compared to SUMMARY1.
>
> ARTICLE: ¡Insert article to summarize here.¿
> SUMMARY1: ¡Insert summary from ALG1 here.¿
> SUMMARY2: ¡Insert summary from ALG2 here.¿
> Which summary **more concisely captures information** in the ARTICLE? You can answer one of SUMMARY1 or SUMMARY2. ANSWER:

Similarly, for capturing how comprehensive and high quality a summary is relative to the other, we employ the following prompt.

> You are an expert summary rater. Given an ARTICLE and two summaries SUMMARY1 and SUMMARY2, compare SUMMARY1 and SUMMARY2 based on **the quality and comprehensiveness of the summary** in the ARTICLE.
>
> Output SUMMARY1 if it is of **higher quality and more comprehensive** in summarizing the ARTICLE compared to SUMMARY2. Alternatively, output SUMMARY2 if it is of **higher quality and more comprehensive** in summarizing the ARTICLE compared to SUMMARY1.
>
> ARTICLE: ¡Insert article to summarize here.¿
> SUMMARY1: ¡Insert summary from ALG1 here.¿
> SUMMARY2: ¡Insert summary from ALG2 here.¿
> Which summary is of **higher quality and more comprehensive** in capturing information in the ARTICLE? You can answer one of SUMMARY1 or SUMMARY2. ANSWER:

For each instance in the validation set, we compare swapped pairs (SUMMARY1, SUMMARY2) and (SUMMARY2, SUMMARY1) with an automatic evaluator [33] for both the conciseness prompt and comprehensiveness prompt. We consider it to be a win for one of the algorithm if it wins with both of the swapped pairs on one of the prompts while at least obtaining a tied result on the other prompt.

I.2 Generations for NLI v. Rouge

In this section, we provide sample summaries from full-CLP, attn-CLP, logit-CLP and the prompting baseline on NLI v. Rouge where we evaluate three weightings $w_1 = (0.8, 0.2), w_2 = (0.5, 0.5), w_3 = (0.2, 0.8)$.

I.2.1 Document 9

Document #9. summarize: Cash, who still presents shows on BBC radio in the South, played his first record, Bill Haley and His Comets' Rock Around the Clock, in Canada. "This little job came up as holiday relief on 15 August 1964," he said. "It feels more like 50 minutes than 50 years. There were times when things weren't the best in the west, but 99.99% of the time they were great." Cash, aged 72, was born in London but went to Canada with the Merchant Navy. His first broadcast give him an audition tape to take to the pirate ships broadcasting offshore to Britain, and he joined Radio London. "I saw this guy huddling over and being very ill, because the ship was going up and down at a heck of a pace," he said. "He said, 'I'm Maurice Cole - they call me Kenny Everett'." When the pirates, which inspired the 2009 film The Boat That Rocked, were outlawed, Cash became one of the first DJs on Radio 1 in 1967 alongside the likes of Tony Blackburn, Pete Murray, Ed Stewart and John Peel. He also worked for Capital Radio for 20 years from 1973 before rejoining the BBC in 1999. "I love talking to the people and getting their reaction," he said. "And, of course, you get to play the best music on the planet, which can't be bad." Cash's shows are broadcast at weekends in Kent - where he lives - Surrey, Sussex, Berkshire, Dorset, Oxfordshire and Hampshire. There will be a special show on 25 August, but he said he was spending the 50th anniversary of his first show getting an MOT for his car. "Maybe I'll have a cup of coffee or something later on with the wife."
Ground Truth Summary: Former pirate radio and Radio One DJ Dave Cash is celebrating 50 years of broadcasting.

Algorithm	High nli, Low rouge	Mid nli, Mid rouge	Low nli, High rouge
full-CLP	A radio DJ has marked the 50th anniversary of his first broadcast.	A BBC DJ has marked the 50th anniversary of his first broadcast on the air.	DJ Kenny Cash has marked the 50th anniversary of his first radio show on Radio London 50 years ago.
attn-CLP	DJ Kenny Cash has marked the 50th anniversary of his first radio show.	DJ Kenny Cash has marked the 50th anniversary of his first radio show.	DJ Kenny Cash has marked the 50th anniversary of his first radio show 50 years ago.
logit-CLP	Kenny Cash has celebrated the 50th anniversary of his first radio show.	Kenny Cash has celebrated the 50th anniversary of his first radio show.	Kenny Cash has celebrated the 50th anniversary of his first radio show.
prompting	DJ Kenny Cash has marked the 50th anniversary of his first broadcast on the BBC.	DJ Kenny Cash has marked the 50th anniversary of his first radio broadcast.	DJ Kenny Cash has marked the 50th anniversary of his first broadcast on the BBC.

I.2.2 Document 11

Document #11. summarize: The game set a new record for the most-watched event in US television, beating last year's audience of 112.2 million. Viewing peaked at 120.8 million during the last quarter, which saw the New England Patriots beat the Seattle Seahawks 28-24. Some 118.5 million watched Katy Perry's half-time show - three million more than last year's show with Bruno Mars. Ratings compiler Nielsen said the 12-minute performance - which also featured Lenny Kravitz and Missy Elliot - attracted the biggest half-time audience since 1991. NBC, which broadcast the game, said the actual audience was likely to have been higher because official ratings did not count people watching in locations such as sports bars. An additional 600,000 viewers watched the game with Spanish-language announcers on NBC's Universo cable network. NBC Sports also offered a live stream to computers and tablets - attracting an average 800,000 people, according to Adobe Analytics. After the game, an estimated 26.5 million people kept their TVs on NBC to watch an episode of James Spader's crime drama The Blacklist, making it the most-watched scripted programme on the network in more than a decade. Next year's Super Bowl will be played in Santa Clara, California, and will be broadcast by CBS.
Ground Truth Summary: An estimated average of 114.4 million people watched Sunday's Super Bowl on NBC, according to initial figures.

Algorithm	High nli, Low rouge	Mid nli, Mid rouge	Low nli, High rouge
full-CLP	The Super Bowl has attracted a huge audience.	Some 120.8 million people watched the Super Bowl on TV on Sunday.	US viewers watched the Super Bowl in the United States with 120.8 million viewers.

attn-CLP	US viewers have watched the Super Bowl.	US viewers have watched the Super Bowl attracting more than 12 million people.	US viewers have watched the Super Bowl attracting more than 12 million people to watch the game.
logit-CLP	US viewers watched the Super Bowl with a huge audience.	US viewers watched the Super Bowl with a huge audience.	US viewers watched the Super Bowl with a huge audience.
prompting	Some 120.8 million viewers tuned in to the Super Bowl in the United States.	Some 120.8 million viewers tuned in to the Super Bowl in the United States.	Some 120.8 million viewers tuned in to the Super Bowl in the United States.

I.2.3 Document 16

Document #16. summarize: Colombian leader Juan Manuel Santos and the Farc rebel commander known as Timochenko signed the deal in an emotional ceremony on Monday evening. "I would like to ask for forgiveness for all the pain that we may have caused during this war," he said. The guests at the ceremony in Cartagena cheered when Timochenko apologised. Some shouted "Yes, we can!" while Farc members and heads of state from Latin America rose to their feet on the stage and applauded. The ceremony which marks the end of 52 years of armed conflict was broadcast live and shown on giant screens in the capital, Bogota, and other large cities. Farc rebels gathered in a number of camps also followed the broadcast. There was so much symbolism in this historic signing - a pen made from a bullet to sign the peace deal, the singing of Beethoven's Ode to Joy, everyone dressed in white. President Santos said this historic moment was a message from Colombia to the world: no more war. "No more war," the crowd chanted in return. This was the first time Timochenko addressed the nation live on TV. He promised the Farc would give up its guns, and more than that, he asked for forgiveness. It earned him a standing ovation. That would have been unthinkable not long ago. But after 50 years of war, many Colombians still aren't ready to forgive. As President Santos put it, the hard work of building peace now lies ahead. Under the deal, the Farc will now become a political party. The rebels will take part in legislative elections in 2018, and - as part of the peace deal - they will be guaranteed a minimum of 10 seats in Congress for their first two legislative periods. President Santos addressed the Farc rebels directly and praised their decision to lay down arms. "Swapping bullets for votes and weapons for ideas is the bravest and most intelligent decision that any rebel group could take," he said. "When you begin your return to society (...) as head of state of the homeland that we all love, I welcome you to democracy." Timochenko said that the rebels would not take up arms anymore. "War is over, we're starting to build peace," he said before his speech was interrupted by the sound of Colombian Air Force fighter jets flying over the venue. Looking up startled, Timochenko's worried expression then turned into a smile and he joked: "Well, this time they came to pay their respects to peace and not to drop bombs." A number of Farc leaders, such Raul Reyes and Mono Jojoy, were killed in bombing raids over the past decade. The peace deal was reached after four years of formal peace talks in the Cuban capital, Havana, which were preceded by two years of secret talks. But before it can come into force it will have to be approved by Colombians in a public vote scheduled for Sunday. While recent polls suggest that a majority of Colombians back the deal, there has been vocal opposition to the agreement. Those in the "no" camp, which is led by former President Alvaro Uribe, say that they do not trust the Farc to stick to their promises. They are also angry that Farc leaders will be entering Congress without serving time in jail for crimes committed during the conflict. President Santos has stressed that there is "no Plan B". He said that if the Colombian people were to reject the peace deal, the country would face more conflict. 1964: Set up as armed wing of Communist Party 2002: At its height, it had an army of 20,000 fighters controlling up to a third of the country. Senator Ingrid Betancourt kidnapped and held for six years along with 14 other hostages 2008: The Farc suffers a series of defeats in its worst year 2012: Start of peace talks in Havana 2016: Definitive ceasefire Full timeline of Farc conflict

Ground Truth Summary: The leader of the Farc rebel group has apologised to the victims of Colombia's armed conflict which ended with Monday's signing of a peace deal.

Algorithm	High nli, Low rouge	Mid nli, Mid rouge	Low nli, High rouge

full-CLP	A historic peace deal between Colombia's Farc rebels and the country's government has been signed.	Colombia has signed a peace deal with the Farc rebel group to end the 52-year conflict.	Colombia has signed a peace deal with the Farc rebel group to end the 50-year conflict.
attn-CLP	Colombia's Farc rebels have signed a peace deal.	Colombia's Farc rebel group have signed a peace deal to end the country's 52-year conflict.	Colombia's Farc rebel group have signed a peace deal to end the country's 52-year conflict.
logit-CLP	Colombia has signed a peace deal with the Farc rebels.	Colombia has signed a peace deal with the Farc rebels to end the 52-year war.	Colombia has signed a peace deal with the Farc rebels to end the 52-year war.
prompting	Colombia's Farc rebel group has signed a peace deal with the country.	Colombia's Farc rebel group has signed a peace deal with the country.	Colombia's Farc rebel group has signed a peace deal with the country.

I.3 Generations for NLI v. TLDR

In this section, we provide sample summaries from full-CLP, attn-CLP, logit-CLP and the prompting baseline on NLI v. TLDR where we evaluate three weightings $w_1 = (0.8, 0.2), w_2 = (0.5, 0.5), w_3 = (0.2, 0.8)$.

I.3.1 Document 2

Document #2. summarize: Two-time major winner McIlroy, 25, and Danish former world number one Wozniacki, 23, announced their engagement on New Year's Eve. Media playback is not supported on this device "The problem is mine," McIlroy said. "The wedding invitations issued at the weekend made me realise that I wasn't ready for all that marriage entails." The couple had been together for more than two years. McIlroy is currently at Wentworth for the BMW PGA Championship, the European Tour's flagship event which starts on Thursday. Wozniacki is set to compete in the French Open in Paris starting on 25 May. In a statement issued through his management company, McIlroy added: "There is no right way to end a relationship that has been so important to two people. "I wish Caroline all the happiness she deserves and thank her for the great times we have had." "The news has shocked the worlds of both golf and tennis, particularly with the timing of the split. McIlroy is due to start a crucial run of tournaments, while Wozniacki is playing the upcoming French Open." Speaking at Wentworth on Wednesday, McIlroy said the decision to end the engagement had been mutual and amicable. He said it was a difficult time, but that the move was best for both of them. McIlroy won the US Open in 2011 and US PGA Championship the following year. Wozniacki, who has reached the final of one grand slam tournament - the US Open in 2009 - is currently ranked 13th in the world. When McIlroy announced their engagement, he tweeted: "I have a feeling it is going to be a great year."
Ground Truth Summary: Northern Irish golf star Rory McIlroy has broken off his engagement with tennis player Caroline Wozniacki.

Algorithm	High nli, Low tldr	Mid nli, Mid tldr	Low nli, High tldr
full-CLP	Rory McIlroy and Caroline Wozniacki have announced they are splitting up.	Rory McIlroy has announced that he and Caroline Wozniacki have split up after two years of passionate commitment.	Rory McIlroy has announced he has ended his engagement with tennis player Caroline Wozniacki, stressing the timing of the split was "shocking" and that the move was best for both partners.
attn-CLP	Rory McIlroy has announced that he is ending his engagement with former tennis partner Caroline Wozniacki.	Rory McIlroy has announced he has ended his engagement with tennis player Caroline Wozniacki, saying the split was mutual and mutually amicable.	Rory McIlroy has announced he has ended his engagement with tennis player Caroline Wozniacki, saying the split was mutual and mutually amicable.
logit-CLP	Rory McIlroy has announced he has ended his engagement with Caroline Wozniacki.	Rory McIlroy has announced he has ended his engagement with Caroline Wozniacki after realising he was not ready for the commitment.	Rory McIlroy has announced he has ended his engagement with Caroline Wozniacki after realising he was not ready for the commitment.
prompting	Rory McIlroy has announced he has ended his engagement with Caroline Wozniacki.	Rory McIlroy has announced he has ended his engagement with tennis player Caroline Wozniacki, saying the split is "no right way to end a relationship that has been so important to two people".	Rory McIlroy has announced he has ended his engagement with tennis player Caroline Wozniacki, saying the split is "no right way to end a relationship that has been so important to two people".

I.3.2 Document 3

Document #3. summarize: The Battle of Britain Memorial Flight's Lancaster, known as Thumper, based at RAF Coningsby, took the skies for a test flight on Monday. The Lancaster, one of only two in the world able to fly, missed most of the 2015 display season. Squadron Leader Martin Morris said a schedule for subsequent flights will be announced over the next few weeks. Sqd Ldr Morris, who heads up the Battle of Britain Memorial Flight, said: "Spares and parts are difficult to find and some had to be manufactured. "The aluminium for the bulkhead had to be sourced from the same type of aluminium as the original aircraft - so it has not been without challenge - but our engineers have succeeded." Hundreds of people turned out to watch as Thumper took to the skies at about 14:00 BST.
Ground Truth Summary: The last airworthy Lancaster bomber in Britain has flown for the first time since being grounded by a fire in May.

Algorithm	High nli, Low tldr	Mid nli, Mid tldr	Low nli, High tldr
full-CLP	A WWII aircraft that missed most of the plane display season has returned to the skies.	A WWII airworthy Lancaster aircraft which missed much of the 2015 display season has taken to the skies.	A WWII vintage Lancaster aircraft has successfully tested flight following considerable engineering and spares challenges, with a schedule for subsequent flights to be announced in the coming weeks.
attn-CLP	A WWII aircraft that missed most of the plane display season has successfully flown.	A WWII Lancaster that missed most of the plane display season has successfully flown for an initial test flight.	A WWII Lancaster that missed most of the plane display season has successfully flown for an initial test flight after overcoming "not without challenges" in sourcing spare parts and aluminium.
logit-CLP	A WWII airworthy Lancaster has flown for the first time since last year.	A WWII airworthy Lancaster has flown for the first time since last year after an extensive overhaul which proved challenging, an RAF unit has said.	A WWII airworthy Lancaster has performed its first test fly since it missed much of 2015 display season, with organisers promising to announce future flights in coming weeks.
prompting	A historic aircraft that missed most of the plane display season has successfully flown.	A historic Lancaster bomber has successfully tested for the first time after some challenging parts had to be manufactured.	A replica Lancaster aircraft which missed most of the 2015 display season has flown for an initial test flight after overcoming "difficult" requirements, including the need to source the exact aluminium type as the original.

I.3.3 Document 6

Document #6. summarize: After the draft deal was published two weeks ago, an irate editorial in the French newspaper Le Monde fumed that the concessions made to Britain were ”exorbitant”. And while some in Mr Cameron’s own Conservative Party dismiss his renegotiation as a watery fudge, politicians in Paris have a very different view. Of particular concern here is the question of future financial regulation. UK ministers, too, think this is the most important part of the whole renegotiation. The issue is this: The majority of EU countries that have the euro are integrating further and building a banking union with common rules and regulations. That process will deepen in the coming years. Mr Cameron wants legal guarantees that the eurozone will not rig the rules in a way that could harm the interests of countries that do not have the euro. But France frets that the UK is seeking special treatment for the City of London, that could harm the single market in financial services. France is also concerned a new mechanism that would allow non-euro countries to escalate their concerns could clog up decision-making. The UK insists it is not asking for any veto or competitive advantage, but this is likely to be the focus of the hastily arranged talks here this evening. President Hollande also faces an election next year - and while he wants the UK to stay in the EU, he will not do a deal that gives his opponents a chance to claim he sold out to Britain. EU renegotiation: Did Cameron get what he wanted? Referendum timeline: What will happen when? The view from Europe: What’s in it for the others? More: BBC News EU referendum special
Ground Truth Summary: Prime Minister David Cameron is to meet French President Francois Hollande in Paris shortly - the latest leg in his drive to negotiate new EU membership terms for the UK.

Algorithm	High nli, Low tldr	Mid nli, Mid tldr	Low nli, High tldr
full-CLP	Talks to re-build the EU with David Cameron are being held in Paris this evening.	As Britain prepares to begin negotiations on a deal with the European Union, debate is likely to turn to the future of financial regulation, and the way this could harm non-euro countries in the single market.	Britain’s discussions with French President Hollande on a renegotiation of the EU’s terms for the UK’s bid to stay remain closely watched but France is concerned that the UK’s request for special treatment in financial services may seriously harm the single market in Europe.
attn-CLP	Britain’s discussions with French officials on a deal to resurrect Britain in the EU are likely to be marked by a number of concerns.	Britain’s discussions with French officials on a deal to resurrect Britain in the EU have drawn some ire, with French politicians concerned about future financial regulation issues.	Britain’s discussions with French officials on a deal to resurrect Britain in the EU this evening will likely be dominated by issues including future financial regulation and concerns that the UK is seeking special treatment for the City of London.
logit-CLP	Prime Minister David Cameron is negotiating with France on a renegotiated EU membership deal that will set out the UK’s demands in terms of banking and financial regulation.	Prime Minister David Cameron’s proposed EU renegotiation has faced heated criticism in France, with some worrying about the UK’s bid to extract special treatment for London.	Prime Minister David Cameron’s proposed EU renegotiation has faced heated criticism in France, with some worrying about the UK’s bid to extract special treatment for London in the single market.
prompting	There is concern that recent EU negotiations on Britain’s membership of the EU have left some countries concerned.	As Britain prepares to begin negotiations on a deal with the European Union, debate is about whether British ministers are getting what they want and whether France harbours fears that the UK is seeking special treatment for the City of London.	As Britain prepares to begin negotiations on a deal with the European Union, debate is about the crucial issue of future financial regulation - and with France worried over fears that the UK is seeking special treatment for City of London.

J Theory of Logit Mixing for Multi-Objective Finetuning

Recall that logit-CLP (described in **??**) amounts to linear combination of the logits. In this section, we show that mixing the logits of expert policies for each individual reward has provable guarantees under coverage conditions. We also provide a counterexample that rules out the optimality of zero-shot methods when this coverage condition is not satisfied. For simplicity, we focus on the two reward case as the general case follows naturally.

In this section, we will use s to denote the input prompt, a to denote the output generation and $|\mathcal{A}|$ to the cardinality of possible outputs. Recall that the optimal policy for a fixed (α, R) has the form: $\pi^\star_{\alpha,R}(a \mid s) \propto \pi_{\text{ref}}(a \mid s)\exp((1-\alpha)R(s,a)/\alpha)$ [20]. Taking the logarithm gives

$$\log \pi^\star_{\alpha,R}(a \mid s) = \log \pi_{\text{ref}}(a \mid s) + \frac{1-\alpha}{\alpha}R(s,a) - \log Z_{\alpha,R}(s),$$

where $Z_{\alpha,R}(s) = \sum_{a'} \pi_{\text{ref}}(a \mid s)\exp((1-\alpha)R(s,a)/\alpha)$ in the partition function. Fix any reward functions R_1, R_2 and weight $\lambda \in [0,1]$, and define $R_\lambda := (1-\lambda)R_1 + \lambda R_2$, we have

$$\begin{aligned}\log \pi^\star_{\alpha,R_\lambda}(a \mid s) &= (1-\lambda)\log \pi^\star_{\alpha,R_1}(a \mid s) + \lambda \log \pi^\star_{\alpha,R_2}(a \mid s) \\ &\quad + (1-\lambda)\log Z_{\alpha,R_1}(s) + \lambda \log Z_{\alpha,R_2}(s) - \log Z_{\alpha,R_\lambda}(s).\end{aligned}$$

Since the partition terms are independent of a, this implies that linearly interpolating the logits of $\pi^\star_{\alpha,R_1}$ and $\pi^\star_{\alpha,R_2}$ produces logits of the optimal policy for the combined reward. In other words, the optimal policy for the combined reward can be expressed as a multiplicative interpolation of the two expert policies. This was also alluded to in Liu et al. [17, Appendix B].

However, since we never know the optimal policy in practice, we can only assume access to ε-optimal policies for each individual reward. The following theorem quantities the sensitivity to ε for the logit mixing approach.

Theorem 2. *Fix any $\alpha, \lambda \in [0,1]$. Suppose $\widehat{\pi}_1$ is an ε-optimal policy for R_1, i.e., $V^\star_{\alpha,R_1} - V_{\alpha,R_1}(\widehat{\pi}_1) \le \varepsilon$. Similarly assume $\widehat{\pi}_2$ is ε-optimal for R_2. Let $\widehat{\pi}_\lambda$ be the logit mixing of $\widehat{\pi}_1, \widehat{\pi}_2$ as described above. Then,*

$$V^\star_{\alpha,R_\lambda} - V_{\alpha,R_\lambda}(\widehat{\pi}_\lambda) \le \varepsilon \cdot \Big(\exp(\eta^2/8)\Big((1-\lambda)C^{\lambda}_{\widehat{\pi}_2,\widehat{\pi}_1} + \lambda C^{1-\lambda}_{\widehat{\pi}_1,\widehat{\pi}_2}\Big) + 4|\mathcal{A}|p^{-1}_{\min}\Big),$$

where η is the maximum ℓ_∞ logit value of $\widehat{\pi}_1$ and $\widehat{\pi}_2$, and $p_{\min} = \min_{i\in\{1,2\}}\min_{x,y}\widehat{\pi}_i(y \mid x)$ is the minimum probability of an action.

Counterexample with bad coverage. In App. C.2, we describe a counterexample for which the zero-shot logit mixing approach provably fails due to lack of coverage. In the following experiment, we also find that zero-shot RS cannot learn the Pareto-optimal policy for this task, whereas CLP ultimately learns the Pareto optimal policy.

J.1 Proofs

Proof of Theorem 2. By Lemma 1, we can focus on bounding $\mathrm{KL}(\widehat{\pi}_\lambda(s) \parallel \pi^\star_\lambda(s))$, we can be simplified as follows:

$$\begin{aligned}&\mathrm{KL}(\widehat{\pi}_\lambda(s) \parallel \pi^\star_\lambda(s)) \\ &= \mathbb{E}_{a\sim\widehat{\pi}_\lambda}[\log \widehat{\pi}_\lambda(a \mid s) - \log \pi^\star_\lambda(a \mid s)] \\ &= \mathbb{E}_{a\sim\widehat{\pi}_\lambda}[(1-\lambda)(\log \widehat{\pi}_1(a \mid s) - \log \pi^\star_1(a \mid s)) + \lambda(\log \widehat{\pi}_2(a \mid s) - \log \pi^\star_2(a \mid s))] \\ &+ (1-\lambda)(\log \widehat{Z}_1(s) - \log Z^\star_1(s)) + \lambda(\log \widehat{Z}_2(s) - \log Z^\star_2(s)) - (\log \widehat{Z}_\lambda(s) - \log Z^\star_\lambda(s)).\end{aligned}$$

Bounding the log partition terms. First, we bound the difference of log partition functions. Let us parameterize the policies $\widehat{\pi}_1(\cdot \mid s) = \sigma(\widehat{\theta}_1(s))$ where $\widehat{\theta}_1(s) \in \mathbb{R}^A$ (where $A = |\mathcal{A}|$) and σ is the softmax operation, *i.e.*, $\sigma(x)[i] = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$. Without loss of generality, we fix $\widehat{\theta}_1(s)[A] = 0$ and thus can rewrite $\widehat{Z}_1(s) = 1 + \sum_{a=1}^{A-1}\exp(\widehat{\theta}(s)[a])$. In particular, we have $\log \widehat{\pi}_1(a \mid s) = \theta(s)[a] - \log \widehat{Z}_1(s)$.

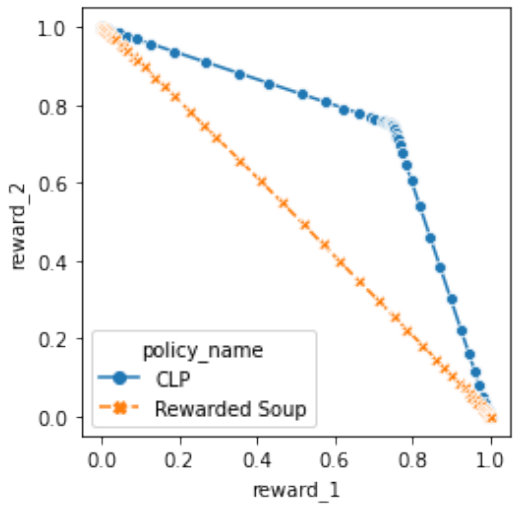


Figure 21: Pareto-fronts of RS and CLP in the counterexample of App. C.2.

We abuse notation and let $Z(\theta(s)) = \sum_a \exp(\theta(s)[a])$. Notice that $\mathrm{KL}(\widehat{\pi}(s) \parallel \pi^\star(s))$ is the Bregman divergence of $f(\theta(s)) := \log Z(\theta(s))$ because $\nabla_\theta f(\theta(s)) = \sigma(\theta(s)) = \pi_\theta(s)$, *i.e.*, we have,

$$\mathrm{KL}(\widehat{\pi}(s) \parallel \pi^\star(s)) = \mathbb{E}_{a\sim\widehat{\pi}(s)}[\widehat{\theta}(a \mid s) - \theta^\star(a \mid s)] + \log Z(\theta^\star(s)) - \log Z(\widehat{\theta}(s)) = \Delta_f(\theta^\star(s), \widehat{\theta}(s)).$$

By Lemma 4, we have that f is σ-strongly convex where $\sigma = \frac{p_{\min}}{A}$, and hence:

$$\|\theta_1^\star(s) - \widehat{\theta}_1(s)\|_2 \leq \frac{2}{\sigma}\Delta_f(\theta_1^\star(s), \widehat{\theta}_1(s)) = \frac{2}{\sigma}\,\mathrm{KL}(\widehat{\pi}_1(s) \parallel \pi_1^\star(s)) \leq 2\varepsilon\sigma^{-1},$$

where the last inequality is due to the premise.

Since $\nabla f(\theta) = \sigma(\theta)$, its ℓ_2-norm is bounded by 1 and thus f is 1-Lipschitz. Hence:

$$\left|f(\theta_1^\star(s)) - f(\widehat{\theta}_1(s))\right| \leq \|\theta_1^\star(s) - \widehat{\theta}_1(s)\|_2 \leq 2\varepsilon\sigma^{-1}.$$

Thus, the log partition terms are bounded as follows:

$$\begin{aligned}
&(1-\lambda)(\log \widehat{Z}_1(s) - \log Z_1^\star(s)) + \lambda(\log \widehat{Z}_2(s) - \log Z_2^\star(s)) - (\log \widehat{Z}_\lambda(s) - \log Z_\lambda^\star(s)) \\
&\leq (1-\lambda)2\varepsilon\sigma^{-1} + \lambda 2\varepsilon\sigma^{-1} + \|\theta_\lambda^\star(s) - \widehat{\theta}_\lambda(s)\|_2 \\
&\leq 2\varepsilon\sigma^{-1} + (1-\lambda)\|\theta_1^\star(s) - \widehat{\theta}_1(s)\|_2 + \lambda\|\theta_2^\star(s) - \widehat{\theta}_2(s)\|_2 \\
&\leq 4\varepsilon\sigma^{-1}.
\end{aligned}$$

Bounding the on-policy term. To perform a change of measure, we can compute the density of $\widehat{\pi}_\lambda$ with $\widehat{\pi}_1$ and $\widehat{\pi}_2$ as follows:

$$\begin{aligned}
\frac{\widehat{\pi}_\lambda(a|s)}{\widehat{\pi}_1(a|s)} &= \left(\frac{\widehat{\pi}_2(a|s)}{\widehat{\pi}_1(a|s)}\right)^\lambda \frac{\widehat{Z}_1(s)^{1-\lambda}\widehat{Z}_2(s)^\lambda}{\widehat{Z}_\lambda(s)}, \\
\frac{\widehat{\pi}_\lambda(a|s)}{\widehat{\pi}_2(a|s)} &= \left(\frac{\widehat{\pi}_1(a|s)}{\widehat{\pi}_2(a|s)}\right)^{1-\lambda} \frac{\widehat{Z}_1(s)^{1-\lambda}\widehat{Z}_2(s)^\lambda}{\widehat{Z}_\lambda(s)}.
\end{aligned}$$

Thus,

$$\begin{aligned}
&\mathbb{E}_{a\sim\widehat{\pi}_\lambda}[(1-\lambda)(\log \widehat{\pi}_1(a \mid s) - \log \pi_1^\star(a \mid s)) + \lambda(\log \widehat{\pi}_2(a \mid s) - \log \pi_2^\star(a \mid s))] \\
&= \frac{\widehat{Z}_1(s)^{1-\lambda}\widehat{Z}_2(s)^\lambda}{\widehat{Z}_\lambda(s)}\Bigg((1-\lambda)\mathbb{E}_{a\sim\widehat{\pi}_1(s)}\left(\frac{\widehat{\pi}_2(a|s)}{\widehat{\pi}_1(a|s)}\right)^\lambda(\log \widehat{\pi}_1(a \mid s) - \log \pi_1^\star(a \mid s)) \\
&\qquad + \lambda\mathbb{E}_{a\sim\widehat{\pi}_2(s)}\left(\frac{\widehat{\pi}_1(a|s)}{\widehat{\pi}_2(a|s)}\right)^{1-\lambda}(\log \widehat{\pi}_2(a \mid s) - \log \pi_2^\star(a \mid s))\Bigg) \\
&\leq \exp(\eta^2/8)\Big((1-\lambda)C^\lambda_{\widehat{\pi}_2,\widehat{\pi}_1} + \lambda C^{1-\lambda}_{\widehat{\pi}_1,\widehat{\pi}_2}\Big)\varepsilon,
\end{aligned}$$

where we used Lemma 3 to bound the factor in front, bounded the density ratios, and used the fact that $\mathrm{KL}(\widehat{\pi}_1(s) \parallel \pi_1^\star(s)) \leq \varepsilon$ and $\mathrm{KL}(\widehat{\pi}_2(s) \parallel \pi_2^\star(s)) \leq \varepsilon$ by premise. □

Lemma 1. *For any π:*

$$V_{\alpha,R}(\pi) = \alpha \mathbb{E}_{s\sim\mu}[\log Z_{\alpha,R}(s) - \mathrm{KL}(\pi(s) \parallel \pi^\star_{\alpha,R}(s))].$$

Therefore,

$$V^\star_{\alpha,R} - V_{\alpha,R}(\pi) = \alpha \mathbb{E}_{s\sim\mu}[\mathrm{KL}(\pi(s) \parallel \pi^\star_{\alpha,R}(s))].$$

Proof. Recall that $\pi^\star_{\alpha,R}(a \mid s) = \frac{\pi_{\text{ref}}(a|s)\exp((1-\alpha)R(s,a)/\alpha)}{Z_{\alpha,R}(s)}$, where $Z_{\alpha,R}(s) = \sum_{a'} \pi_{\text{ref}}(a' \mid s)\exp((1-\alpha)R(s,a')/\alpha)$ is the partition function. Then,

$$\begin{aligned}
V_{\alpha,R}(\pi) &= \mathbb{E}_{s\sim\mu,a\sim\pi(s)}[(1-\alpha)R(s,a) - \alpha(\log \pi(a \mid s) - \log \pi_{\text{ref}}(a \mid s))] \\
&= \alpha \mathbb{E}_{s\sim\mu}[\log \pi^\star_{\alpha,R}(a \mid s) - \log \pi(a \mid s) + \log Z_{\alpha,R}(s)] \\
&= \alpha \mathbb{E}_{s\sim\mu}[\log Z_{\alpha,R}(s) - \mathrm{KL}(\pi(s) \parallel \pi^\star_{\alpha,R}(s))].
\end{aligned}$$

□

Lemma 2 (Hoeffding's Lemma)**.** *Let X be a random variable such that $X \in [a,b]$ w.p. 1. Then for all $\lambda \in \mathbb{R}$,*

$$\log \mathbb{E}\exp(\lambda X) \leq \lambda \mathbb{E}X + \tfrac{\lambda^2(b-a)^2}{8}.$$

Lemma 3. *For any $x, y \in \mathbb{R}^n$ and $\lambda \in [0,1]$, we have*

$$\log\left(\frac{(\sum_i \exp(x_i))^{1-\lambda}(\sum_i \exp(y_i))^\lambda}{\sum_i \exp((1-\lambda)x_i + \lambda y_i)}\right) \leq B^2/8,$$

where $B = \|x\|_\infty \vee \|y\|_\infty$.

Proof.

$$\begin{aligned}
&(1-\lambda)\log \textstyle\sum_i \exp(x_i) + \lambda \log \sum_i \exp(y_i) \\
&= (1-\lambda)\log \mathbb{E}_{i\sim\mathcal{U}([n])}\exp(x_i) + \lambda \log \mathbb{E}_{i\sim\mathcal{U}([n])}\exp(y_i) + \log(n) \\
&\leq (1-\lambda)\mathbb{E}_{i\sim\mathcal{U}([n])}x_i + (1-\lambda)B^2/8 + \lambda \mathbb{E}_{i\sim\mathcal{U}([n])}y_i + \lambda B^2/8 + \log(n) && \text{(Lemma 2)} \\
&= \mathbb{E}_{i\sim\mathcal{U}([n])}[(1-\lambda)x_i + \lambda y_i] + \log(n) + B^2/8 \\
&\leq \log \mathbb{E}_{i\sim\mathcal{U}([n])}\exp((1-\lambda)x_i + \lambda y_i) + \log(n) + B^2/8 && \text{(Jensen's inequality)} \\
&= \log \textstyle\sum_i \exp((1-\lambda)x_i + \lambda y_i) + B^2/8.
\end{aligned}$$

□

Lemma 4. *Let $x \in \mathbb{R}^{n-1}$ and $f(x) = \log(1 + \sum_i \exp(x_i))$. Then f is σ-strongly convex for $\sigma \geq \frac{p_{\min}}{n}$ where $p_{\min} = \min(\frac{1}{1+\sum_i \exp(x_i)}, \frac{\exp(x_1)}{1+\sum_i \exp(x_i)}, \ldots, \frac{\exp(x_{n-1})}{1+\sum_i \exp(x_i)})$.*

Proof. The gradient and Hessian of f are:

$$\begin{aligned}
\nabla f(x) &= \sigma(x) \\
\nabla^2 f(x) &= \mathrm{diag}(\sigma(x)) - \sigma(x)\sigma(x)^\top.
\end{aligned}$$

Now, apply Lemma 5 to the Hessian, which completes the proof. □

Lemma 5. *Let $p \in \mathbb{R}_+^{n-1}$ such that $\sum_i p_i < 1$. Then $H = \mathrm{diag}(p) - pp^\top$ satisfies $\lambda_{\min}(H) \geq \frac{p_{\min}}{n}$ where $p_{\min} = \min(p_1, p_2, \ldots, p_{n-1}, 1 - \sum_i p_i)$.*

Proof. To lower bound the minimum eigenvalue, we'll lower bound the quadratic form for any $v \in \mathbb{R}^{n-1}$ with $\|v\|_2 = 1$. Let $\mu = \sum_i p_i v_i$ and note that

$$\textstyle\sum_i p_i (v_i - \mu)^2 = \sum_i p_i v_i^2 - \mu^2 + (\sum_i p_i - 1)\mu^2.$$

Hence,

$$\begin{aligned} v^\top H v &= \textstyle\sum_i p_i v_i^2 - \mu^2 \\ &= \textstyle\sum_i p_i (v_i - \mu)^2 + (1 - \sum_i p_i)\mu^2 \\ &\geq p_{\min}(\textstyle\sum_i (v_i - \mu)^2 + \mu^2). \end{aligned}$$

It remains to show that $f(\mu) := \sum_i (v_i - \mu)^2 + \mu^2 \geq \frac{1}{n}$. Note that $f'(\mu) = 2n\mu - 2\sum_i v_i$ and $f''(\mu) = 2n$. Thus, f is convex and its minimizer is $\widetilde{\mu} := \frac{1}{n}\sum_i v_i$. Finally,

$$\begin{aligned} f(\widetilde{\mu}) &= n\widetilde{\mu}^2 + \textstyle\sum_i v_i^2 - 2\sum_i v_i \widetilde{\mu} \\ &= 1 + \widetilde{\mu}(n\widetilde{\mu} - 2\textstyle\sum_i v_i) && (\|v\|_2 = 1) \\ &= 1 - \tfrac{1}{n}(\textstyle\sum_i v_i)^2 \\ &\geq 1 - \tfrac{n-1}{n} = \tfrac{1}{n}. && (\,(\textstyle\sum_i v_i)^2 \leq (n-1)\sum_i v_i^2 = n-1) \end{aligned}$$

Thus, we've shown that $v^\top H v \geq \frac{p_{\min}}{n}$ for all $v \in \mathbb{R}^{n-1}$ with $\|v\|_2 = 1$, which implies the claim. □

K Conclusion

We introduced CLP, a flexible framework for MOFT that leverages techniques from multi-task training and parameter efficient finetuning to develop steerable LMs that adapt their generations to produce near Pareto optimal behavior across different weightings of individual rewards. We provide extensive benchmarking and ablations to better understand factors that enable the development of steerable LMs within the CLP framework. We supplement this with theoretical results that present conditions under which zero shot approaches work and when multi-task training is *necessary* to obtain near-optimal behavior. Promising future directions include (a) understanding other conditioning mechanisms such as soft tokens [14], (b) automated tuning of the weight sampling distributions [10], (c) non-linear reward scalarizations [26] and risk-sensitive RL alignment [37], and (d) designing adaptive algorithms with instance-dependent guarantees [36].