

---

# Imitating Language via Scalable Inverse Reinforcement Learning

---

Markus Wulfmeier   Michael Bloesch   Nino Vieillard   Arun Ahuja   Jörg Bornschein  
Sandy Huang   Artem Sokolov   Matt Barnes   Guillaume Desjardins   Alex Bewley  
Sarah Maria Elisabeth Bechtle   Jost Tobias Springenberg   Nikola Momchev  
Olivier Bachem   Matthieu Geist \*   Martin Riedmiller  
Google DeepMind

## Abstract

The majority of language model training builds on imitation learning. It covers pretraining, supervised fine-tuning, and affects the starting conditions for reinforcement learning from human feedback (RLHF). The simplicity and scalability of maximum likelihood estimation (MLE) for next token prediction led to its role as predominant paradigm. However, the broader field of imitation learning can more effectively utilize the sequential structure underlying autoregressive generation. We focus on investigating the inverse reinforcement learning (IRL) perspective to imitation, extracting rewards and directly optimizing sequences instead of individual token likelihoods and evaluate its benefits for fine-tuning large language models. We provide a new angle, reformulating inverse soft-Q-learning as a temporal difference regularized extension of MLE. This creates a principled connection between MLE and IRL and allows trading off added complexity with increased performance and diversity of generations in the supervised fine-tuning (SFT) setting. We find clear advantages for IRL-based imitation, in particular for retaining diversity while maximizing task performance, rendering IRL a strong alternative on fixed SFT datasets even without online data generation. Our analysis of IRL-extracted reward functions further indicates benefits for more robust reward functions via tighter integration of supervised and preference-based LLM post-training.

## 1 Introduction

In recent years, the imitation of existing human knowledge via large datasets has become a key mechanism underlying increasingly capable and general artificial intelligence systems [17, 45, 9]. Pretraining and supervised fine-tuning phases for large language models (LLMs) predominantly rely on imitation learning, in particular next token prediction via maximum likelihood estimation (MLE). In addition, preference-based fine-tuning is affected by imitation via initial online data generation and optimization objectives such as regularization towards the previously fine-tuned LLM [46, 12].

The field of imitation learning for sequential decision making has a long-standing history for applications such as robotic control [4, 29]. Recently, perspectives to language modeling have shifted towards explicit treatment as a sequential decision making problem – in particular for later stages of model adaptation via reinforcement learning from human feedback (RLHF) [46, 13, 17, 66]. This vantage point opens up new opportunities for the effective use of different data sources and obtaining aligned models that better represent human intent. It includes a broader scope for which data contains information about rewards and preferences as well as the dynamics-aware optimization of each action based on its future impact – all while taking into account computational scalability.

---

\*Now at Cohere.

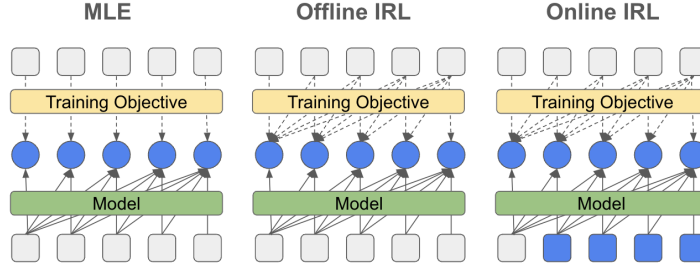


Figure 1: Data usage and optimization flow in MLE, offline and online IRL. Independent of the method, current models use the history of past tokens to predict the next. However, MLE purely optimizes the current output for exact matching the corresponding datapoint while IRL-based methods take into account the impact on future tokens. Online optimization additionally conditions on past model generations rather than the original dataset. Grey and blue objects respectively represent training data and model generations. The impact of future datapoints is often indirect and mediated via learned functions (e.g. the discriminator in GAIL [25] and the Q-function in IQLearn [20]).

Due to the importance of imitation for language modelling, we believe detailed analysis of the underlying problem for sequential decision making is warranted. Despite the widespread perspective of RL for aligning and fine-tuning language models (via RLHF), supervised learning via maximum likelihood estimation for next token prediction remains the dominant component of our imitation learning pipelines due to its simplicity and scalability. However, pure MLE for next token prediction (including “teacher forcing” [64]) can create challenges in autoregressive models, many of which being related to classic challenges with behavior cloning [8], its equivalent in the context of sequential decision making. Compounding errors can occur due to iterative model application creating data sequences which further shift from the training distribution [56, 30], growing increasingly likely for longer sequences [16]. In particular, a model’s own samples can cause such distribution shifts and exposure bias [52, 6, 5]. Taking the RL perspective to imitation aims to mitigate these issues via dynamics-aware optimization, where each action is optimized for the impact on the whole future trajectory. It further enables a shift from passive to active learning, where the system actively generates data. Furthermore, best performance of the fine-tuned model is only one metric of importance. Indeed, continued alignment of language models to human preferences requires sampling for a given prompt, collecting human preferences over these completions, and finally aligning the model via preference fine-tuning. Improving the diversity of sampled completions via temperature sampling [7], or using mixtures of past models [61] has been linked to performance improvements. Studying inverse RL, potential divergences, and regularizations provides another angle to increasing diversity [23].

In this paper, we investigate RL-based optimization, in particular the distribution matching perspective to inverse reinforcement learning (IRL), for fine-tuning language models; which can be contrasted with standard MLE as depicted in Figure 1. Our goal is improved understanding of when, and how, IRL can be used as an effective alternative for supervised MLE in the fine-tuning pipeline. The evaluation covers both adversarial and non-adversarial, offline and online methods. We further extend inverse soft Q-learning [20] to explicitly provide a principled connection to classical behavior cloning or MLE. Our experiments range from 250M to 3B parameter models for the encoder-decoder T5 [51] and decoder-only PaLM2 [3] models. Throughout evaluation, we investigate task performance and diversity of model generations illustrating clear benefits of inverse RL over behavior cloning for imitation. A further, principal benefit of RL-centric imitation – close to RLAIIF, RL from Artificial Intelligence Feedback – is the natural connection to later RLHF stages via rewards obtained from demonstration data and we take first steps to analyse the value of IRL-obtained reward functions.

Our key contributions are:

- We investigate the RL-centric perspective to imitation for LLMs, extracting rewards and directly optimizing actions for sequence generation instead of individual token likelihood.
- We reformulate inverse soft Q-learning as temporal difference regularized extension of MLE. This explicitly bridges between MLE and algorithms exploiting the sequential nature underlying language generation and enables computationally cheap offline training.
- We compare MLE and IRL formulations including adversarial and non-adversarial, offline and online methods to improve our understanding of imitation in LLMs. Our main results show better or on par task performance, with increased diversity of model generations, in particular demonstrating that key improvements can be obtained via (better scalable) offline IRL.

- Finally, we analyze the extracted reward functions indicating the potential usefulness of IRL to obtain discriminative in addition to generative improvements from demonstrations.

## 2 Methods

Language generation can be modeled as a sequential decision making problem. On a token level, it is the problem of generating the next token  $x_i$  given the already generated sequence of tokens  $(x_0, \dots, x_{i-1})$ . We thus seek a distribution  $\pi(x_i|x_0, \dots, x_{i-1})$ , which we will also refer to as a policy. For a given policy the sequence likelihood  $\mathbf{x} = (x_0, \dots, x_N)$  can be computed autoregressively:

$$p(\mathbf{x}) = \prod_{i=0}^N \pi(x_i|x_0, \dots, x_{i-1}). \quad (1)$$

The classic maximum likelihood estimation based approach leverages this factorization in order to efficiently train the policy by maximizing the log-likelihood of the training sequences,  $\mathcal{D} = \{\mathbf{x}^0, \dots, \mathbf{x}^M\}$ :

$$\arg \max_{\pi} \sum_{\mathbf{x} \in \mathcal{D}} \log p(\mathbf{x}) = \arg \max_{\pi} \sum_{\mathbf{x} \in \mathcal{D}} \sum_{i=0}^N \log \pi(x_i|x_0, \dots, x_{i-1}). \quad (2)$$

For fine-tuning problems, where a pre-trained model is fine-tuned to a particular set of tasks, the problem can be formulated analogously but may have additional conditioning variables which we will leave out for the sake of clarity.

**Distribution matching.** State-action distribution matching algorithms [25, 20], which are well-established in the field of imitation learning – and can be seen as solving an IRL problem see e.g. [25] – approach the problem in a different manner: They seek to minimize the divergence between the  $\gamma$ -discounted state-action distribution of the policy  $\pi(a|s)$  and the discounted state-action distribution of the expert policy  $\pi_E(a|s)$ . We define the discounted state distribution for a Markov Decision Process (MDP) as  $\rho(s) = (1 - \gamma) \sum_{i=0}^{\infty} \gamma^i P(s_i = s|\pi)$  and the discounted state-action distribution is accordingly defined as  $\mu_{\pi}(s, a) = \pi(a|s)\rho(s)$ ; where  $P(s_i = s|\pi)$  is the probability of seeing state  $s$  when acting according to the policy  $\pi$ . For our autoregressive generation MDP, the state corresponds to the concatenation of already generated tokens (commonly including the prefix or prompt),  $s_i = (x_0, \dots, x_{i-1})$  and the action is the next token,  $a_i = x_i$  thus yielding a problem with deterministic dynamics. We omit state and action arguments in the following discussion whenever it is clear from the context.

In order to enable more straightforward algebraic manipulation, the divergence is often combined with a weighted causal entropy term  $H(\pi) = -E_{\mu_{\pi}} \log(\pi(a|s))$ . The goal is then to find a policy that minimizes the objective  $\mathcal{J}(\pi)$ :

$$\mathcal{J}(\pi) := D_f(\mu_{\pi}||\mu_E) - \lambda H(\pi), \quad (3)$$

where  $D_f = E_{\mu_E} [f(\frac{\mu_{\pi}(a,s)}{\mu_E(a,s)})]$  is an  $f$ -divergence. Different  $f$ -divergences have been used in the literature [25, 23]. When taking the example of the reverse KL divergence, with  $f(t) = f_{\text{RKL}}(t) = -\log(t)$ , we can decompose the objective into a state distribution and an MLE term:

$$\min_{\pi} D_{f_{\text{RKL}}}(\mu_{\pi}||\mu_E) = \text{KL}(\rho_E||\rho) + E_{\rho_E} [\text{KL}(\pi_E||\pi)]; \quad (4)$$

where KL denotes the KL-divergence which corresponds to maximum likelihood (on the discounted state distribution) for the second part after dropping terms independent of  $\pi$ . In comparison to MLE, IRL algorithms thus also try to match expert actions but additionally attempt to match the state visitations of the expert. The use of different  $f$ -divergences can influence mode seeking and mode covering properties of the optimal policy [23].

**Adversarial imitation.** The state-action divergence can be hard to evaluate for two reasons: first it requires many samples from the current policy and second it requires access to the unknown expert densities. In a first step, a variational representation of the  $f$ -divergence can be leveraged to avoid expert densities:

$$\min_{\pi} \mathcal{J}(\pi) = \min_{\pi} \max_{g: \mathcal{S} \times \mathcal{A} \rightarrow \text{dom}_{f^*}} -E_{\mu_E} [f^*(g)] + E_{\mu_{\pi}} [g] + \lambda E_{\mu_{\pi}} [\log \pi], \quad (5)$$

where  $f^*$  is the convex conjugate of  $f$  and  $\text{dom}_{f^*}$  is its domain. Notably, GAIL [25] can be retrieved by using the Jensen-Shannon divergence with its convex conjugate  $f^*(g) = -\log(1 - \exp(g(s, a)))$  and defining the discriminator  $D(s, a) := \exp(g(s, a))$ :

$$\min_{\pi} \max_{D: \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}} E_{\mu_E}[\log(1 - D)] + E_{\mu_{\pi}}[\log(D) + \lambda \log \pi]. \quad (6)$$

**Non-adversarial imitation.** From here, we can re-derive IQLearn [20], but instead consider state value rather than state-action value functions. This representation further allows us to establish a clearer relationship to MLE. Using a change of variable  $r : \mathcal{S} \times \mathcal{A} \rightarrow -\text{dom}_{f^*}$  with  $r(s, a) = -g(s, a)$  and re-arranging the terms, we start by rendering the internal RL-problem more explicit:

$$\min_{\pi} \mathcal{J}(\pi) = - \max_{\pi} \min_r E_{\mu_E}[f^*(-r)] + E_{\mu_{\pi}}[r - \lambda \log \pi], \quad (7)$$

where  $r(s, a)$  can be interpreted as an ‘implicit’ reward function and we omit its arguments in the following for brevity. Due to the convexity of  $f^*$  and the concavity of the causal entropy (other terms are linear) this is a saddle point problem [20] and the max-min can be swapped:

$$\min_{\pi} \mathcal{J}(\pi) = - \min_r E_{\mu_E}[f^*(-r)] + \max_{\pi} E_{\mu_{\pi}}[r - \lambda \log \pi], \quad (8)$$

where the second term now corresponds to the discounted cumulative reward of a soft, or entropy regularized, RL problem [74]. The soft-RL problem is well understood [22] and we can use analytical identities regarding its solution for further simplification. In particular, we have that [42]

$$r(s, a) - \lambda \log \pi^r(a|s) = v^r(s) - E_{s' \sim p(\cdot|a, s)}[\lambda v^r(s')] \quad \forall s \in \mathcal{S}, a \in \mathcal{A}, \quad (9)$$

where  $\pi^r$  is the optimal policy for the reward  $r$  and  $v^r$  and  $\mu^r$  are respectively the corresponding state value function and discounted state-action distribution<sup>2</sup>. This is applied to the right hand term of Eq. (8) after maximization to obtain (after dropping the arguments for conciseness).

$$\min_{\pi} \mathcal{J}(\pi) = - \min_r E_{\mu_E}[f^*(-r)] + E_{\mu^r}[v^r - E_{s'} \gamma v^{r'}]. \quad (10)$$

Using a telescoping argument [20, 57], we can relate the value of the initial state distribution of a policy to the difference in values on the state-action distribution induced by arbitrary other policies, i.e.  $E_{\rho_0}[v^r] = E_{\mu_{\pi}}[v^r - \lambda v^{r'}]$ . In particular, here we can change the second expectation’s state-action distribution to one induced by the expert policy yielding:

$$\min_{\pi} \mathcal{J}(\pi) = - \min_r E_{\mu_E}[f^*(-r) + v^r - E_{s'} \gamma v^{r'}] \quad (11)$$

$$= - \min_r E_{\mu_E}[f^*(-r) + r - \lambda \log \pi^r], \quad (12)$$

where we use Equation (9) again in the second line. Here we obtain an explicit MLE term (the last term in the above equation), albeit via the intermediate of the optimal policy of the reward. We currently also have a minimization problem in the reward  $r$  and the optimal policy being a function thereof. This can be changed by leveraging the bijective relationship between the reward and the optimal Q-value,  $r(q^*) = q^* - E_{s'}[\lambda v^{r'}]$  [20, 42], or analogously between the reward and a state value and a policy,  $r(v^r, \pi^r) = v^r + \lambda \log \pi^r - E_{s'}[\lambda v^{r'}]$  (using  $q^r = v^r + \lambda \log \pi^r$ ). We can thus reparameterize the problem and optimize the *optimal* state value and policy instead of the reward.

$$\min_{\pi} \mathcal{J}(\pi) = - \min_{v^r, \pi^r} E_{\mu_E}[f^*(-r(v^r, \pi^r)) + r(v^r, \pi^r) - \lambda \log(\pi^r)]. \quad (13)$$

Choosing the  $\chi^2$ -divergence with convex conjugate  $f^*(t) = -\frac{t^2}{4} + t$  is particularly convenient at this point since it can be combined with rescaling the value  $v_{\lambda}^r = v^r/\lambda$  to obtain our reformulated IQLearn objective to be minimized.

$$\boxed{\mathcal{J}_{\text{IQLearn}}(v_{\lambda}^r, \pi^r) = E_{\mu_E}[\underbrace{\lambda (v_{\lambda}^r + \log \pi^r - E_{s'}[\gamma v_{\lambda}^{r'}])^2}_{\text{regularization}} - \underbrace{\log(\pi^r)}_{\text{MLE}}]}. \quad (14)$$

With this derivation we show the clear relation between MLE and the inverse RL perspectives (expanding on [48]): we can see distribution matching (and thus inverse RL) as performing maximum

<sup>2</sup>Note that in practice, we parameterize  $v_{\lambda}^r$  based on the logits of  $\pi$  using the identities  $\pi^r(a|s) \propto \exp q^r(s, a)$  and  $v^r(s) = \log \sum_a \exp q^r(s, a)$  as explained in the appendix.

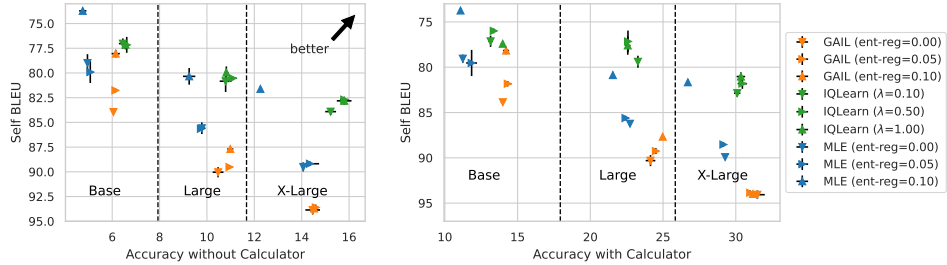


Figure 2: GSM8k results for fine-tuning with MLE, IQLearn, and GAIL across different regularization strengths. In particular MLE shows strong performance reduction with higher entropy cost. Larger models demonstrate higher performance but also stronger self similarity across generations, rendering effective trading of between task performance and diversity highly relevant. Error bars indicate the standard error of the mean after repeating the experiment with 3 different seeds.

likelihood with a dynamics dependent temporal difference regularization term. In contrast to the adversarial setting, this objective does not require samples from the current policy but uses expert samples only<sup>3</sup>. The regularization term couples the learned policy to a value function and favors policies where the log probability of actions matches the difference in state values. An advantage of the above formulation is that it allows annealing of the regularization term where setting  $\lambda = 0$  retrieves standard MLE and where by adjusting  $\lambda$  the regularization strength can be flexibly increased.

### 3 Experiments

In this section, we evaluate the benefits of different inverse RL based methods in comparison to MLE for training large language models. We assess their impact on task performance, diversity of model generations, and computational requirements. Concretely, we compare MLE-based next token prediction and different IRL methods for fine-tuning LLMs on common benchmarks. In addition, we perform ablations on online<sup>3</sup> vs offline versions of IQLearn, showing results across dataset and model sizes. We finally add analysis of the implicitly learned rewards extracted from SFT data via IRL methods (which bring the potential downstream use to aid RLHF/RLAIF [46, 37] training stages).

These experiments mainly aim to answer the following questions:

- Do IRL methods provide a scalable, effective alternative to MLE for fine-tuning?
- How are different algorithms placed on the Pareto front of task performance and diversity?
- For which task and dataset properties is IRL particularly relevant?
- What is the impact of online data for IRL training?
- How informative are rewards extracted from SFT data?

#### 3.1 Algorithms, baselines, and datasets

In addition to naive maximum likelihood estimation for next token prediction, we evaluate the following IRL methods. Generative adversarial imitation learning (GAIL) [25] represents a common adversarial framework training a discriminator to distinguish transitions from agent and dataset and a separate policy. We heuristically adapt the algorithm to mitigate instability for adversarial training [69, 67] including the start for MLE-trained checkpoints and additional loss terms (including MLE) with further details in Appendix A.1.3. IQLearn [20] departs from adversarial learning and our reformulation from Eq. 14 enables us principled control of the temporal difference regularization component to retain stable training. We will use the reformulated offline variant of the algorithm in all experiments and further add an ablation to its online version in Section 3.3.1. Since we compare all methods with respect to task performance and diversity of generations, we additionally evaluate further entropy regularization terms for the MLE baseline; clearly denoted with ‘ent-reg’ in all plots, corresponding to the respective regularization parameter  $\lambda$  in GAIL and MLE (see Appendix A.2 for a more detailed description). In line with previous work on inverse RL for language modelling,

<sup>3</sup>An online version of IQLearn is derived in Appendix A.1.2.

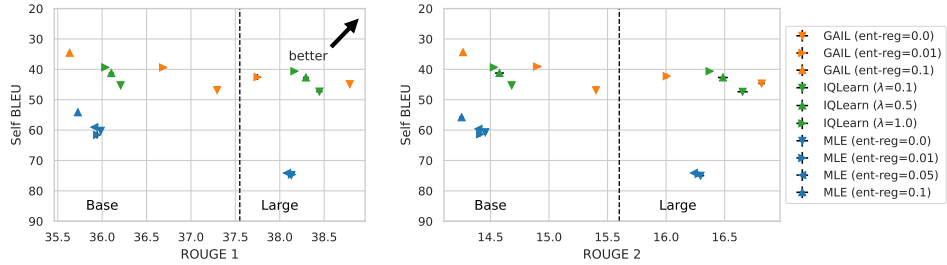


Figure 3: XSUM results for models trained with MLE, IQLearn, and GAIL across different regularization strengths. ROUGE 1 and ROUGE 2 are used as performance metrics on the x-axes with Self-BLEU as diversity measure on the y-axis. Entropy regularizing large MLE and GAIL trained models with 0.1 leads to catastrophic results outside the limits of the plot. Figure 9 in the appendix shows the corresponding plots for ROUGE-LSUM.

we apply a short warm-up phase with pure MLE [15, 65]. Rather than separate experiments or heuristic combinations, the explicit MLE term emerging out of our IRL objective in Section 2 enables principled integration of this mechanism.

We use the following datasets and subsets for ablation in the following sections: XSUM [43], GSM8k [14], TLDR [58], and WMT22 [34]. Unlike parameter-efficient fine-tuning via adapters [27] as used in prior work [15], we focus on the full fine-tuning setting to decouple our analysis from the specifics of adapter-based optimization dynamics [10].

### 3.2 Quality-diversity evaluations

We evaluate both encoder-decoder and decoder-only model classes, respectively using the T5 [51] and PALM2 [3] models. Our main visualizations focus on task performance and diversity of model generations. For task performance, we use the standard metrics for the respective benchmarks (e.g. ROUGE-1 and accuracy percentage). To measure diversity of model generations we calculate self-similarity of generated examples as measured by Self-BLEU [73]. A high score denotes low diversity and vice versa. Using these different axes of evaluation allows us to visualize Pareto fronts between performance and diversity which we will use to assess algorithmic differences. We further add the evaluation via per-token-entropy in Appendix A.3.1.

#### 3.2.1 T5 models

We perform experiments with the base, large, and xl T5 models [51] on the XSUM [43] and GSM8k [14] tasks. These models further serve as foundation for our later ablations. We are able to obtain small but notable gains in performance across all tasks, as shown in Figures 2 and 3, in particular math and reasoning tasks show clear improvements in accuracy when fine-tuning with IRL compared to MLE. We hypothesize that specific and shared structure of responses is better exploited via IRL methods. There is a more emphasized boost in the diversity of model generations for IQLearn over MLE. In comparison to prior work [65], we have been able to stabilize GAIL training, with minor changes described in Appendix A.1.3, but are required to start from a checkpoint previously trained via MLE to ensure strong similarity between dataset and model generations starting GAIL training. This only applies to the T5 model class and for PaLM2 models GAIL is highly

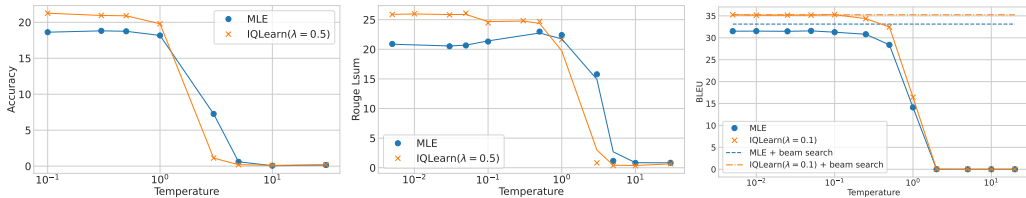


Figure 4: PaLM2 results for various sampling temperatures with MLE and IQLearn. Left: GSM8k, Mid: TLDR, Right: WMT22, including beam search. By propagating sequence information during training, IQLearn reduces inference time dependency on beam search for improving performance.

challenging to stabilize with details in Appendix A.1.3. While entropy terms can further be added to MLE optimization, we hypothesize that better trade-offs between diversity and performance can be obtained via methods able to aggregate information across trajectories to optimize entropy over a different space, complete trajectories rather than just per step policy entropy.

### 3.2.2 PaLM2 models

We also perform experiments with PALM2 models [3], specifically fine-tuning from a pre-trained PALM2 ‘Gecko’ model. We evaluate offline IQLearn on the summarization task TLDR [58], the mathematical reasoning task GSM8k [14], and the large (285M examples) English-to-German translation dataset WMT22 [34]. We limit these experiments to a single choice of the regularization parameter  $\lambda$  per task to save computational costs and instead include an analysis of the effect of the sampling temperature parameter during sampling (noting that we similarly observed IQLearn outperforming MLE at varying temperatures for the T5 models). We selected the checkpoints with early stopping for WMT22 as BC was overfitting on the task, unlike TLDR and GSM8K for which we evaluated their latest checkpoints.

Similar to the previous section, we perceive improvements over MLE on all three benchmarks, though for lower accuracy values MLE covers a part of the front. Figure 4 summarizes these improvements, showing the performance of the trained models depending on the temperature used during sampling responses for the test set prompts. These results show a similar behavior between all three tasks, where IQLearn achieves higher performance in a low temperature regime.

### 3.3 Analysis and ablations

We perform additional experiments and post-training analyses to better understand the impact of dataset size, initial checkpoints and computational demands of offline and online algorithms.

#### 3.3.1 Computational efficiency and accuracy for online & offline inverse RL

One of the key benefits of (offline) MLE compared with (online) IRL-based methods is its lower computational costs. These are principally related to online rollouts – slow, autoregressive sampling from the model. Our re-formulation of IQLearn results in an algorithm that can be applied offline on a fixed dataset, which underlies all IQLearn results presented, mitigating this limitation. In this section, we additionally present update and sampling times across algorithms<sup>4</sup> and the comparison with the online application of IQLearn (see Appendix A.1.2). Figure 5 demonstrates minimal task performance gains (for the T5 base mode on GSM8k), though considerably improved diversity for model generations. At the same time, Table 1 visualizes the relative cost of sampling in comparison to different algorithm updates, excluding sampling. Note that MLE batch sizes are smaller than IRL ones as we add additional online samples to the batch. Experiment times for online IRL can be reduced, but at the cost of additional hardware, via distributed training with separate sampling infrastructure. The application choice of online or offline IRL finally lies with the practitioner trading of additional computational cost with diversity benefits.

<sup>4</sup>Computational resources and time are in practice highly dependent on hardware and implementation but always include crucial, additional sampling costs.

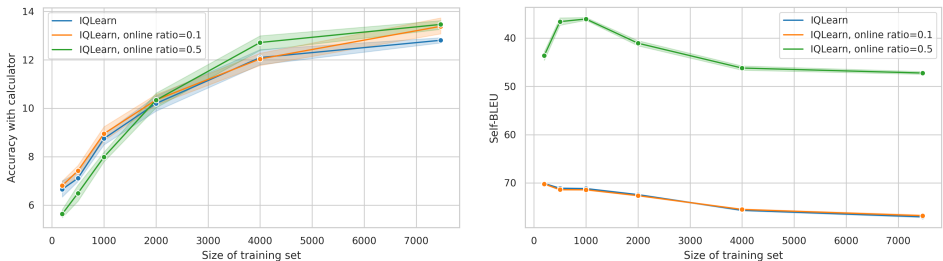


Figure 5: Left: performance of offline and online inverse RL performance with online ratio describing the ratio of offline data used. Right: diversity of model generations. While only showing limited gains in performance, diversity clearly improves.

We provide intuition for low differences between offline and online Inverse RL via toy experiments in Appendix A.3.3. At its core, the dynamics of autoregressive language generation, in particular the concatenation underlying single-turn settings, prevents exact recovery after mistakes, which could be otherwise learned via online IRL. Therefore, extensions of the LLM action space to enable recovery is a fruitful direction for increased benefits from online data with RL or IRL methods [15].

Table 1: Algorithm profiling with computation in milliseconds. ‘Sampling’ refers to generating a number of sequences equivalent to batch size and often uses equal or more time than updates. These times depend on hardware, implementation and code optimization.

	T5-base	T5-large	T5-XL
MLE Update	189 ± 11	422 ± 17	1031 ± 22
GAIL Update	451 ± 13	1064 ± 25	1410 ± 17
IQLearn Update	196 ± 13	606 ± 13	1355 ± 46
+ Sampling (for online IRL)	443 ± 45	1345 ± 211	1823 ± 327

### 3.3.2 Dataset size ablations

Evaluating training on smaller subsets of GSM8k and XSUM with T5 base is respectively pictured in Figures 6 and 7. Performance improvements are consistent across dataset sizes. The analysis of subsets of XSUM further demonstrates increased robustness against overfitting with IQLearn not showing any of the performance loss over time that plagues MLE in particular with the smallest subsets which cannot be overcome with simple entropy regularization. In line with arguments around the compounding of errors in imitation learning [56], we find that both datasets with longer targets and smaller datasets show stronger task performance gains for IRL.

### 3.4 Reward analysis

In comparison to related work on applying IRL to classical control domains, there is no access to ground truth reward functions underlying the process of data generation. Instead, we measure the correlation between IRL extracted rewards and other task-specific performance metrics. High correlation here tells us how informative a reward function is w.r.t. task performance.

In particular, IQLearn represents learned rewards implicitly via the Q-function as  $r_t = Q(s_t, a_t) - \gamma V(s_{t+1})$ , and its online version (i.e. with a non-zero mix-in ratio  $\alpha$  of on-policy examples, see Appendix A.1.2) additionally exposes the algorithm to (initially, low reward) policy rollouts to help discriminate between them and (high reward) SFT data. In Table 2, we report the Spearman’s rank correlation coefficient between accumulated rewards (over complete sampled trajectories for the full validation sets) for online IQLearn ( $\alpha = 0.1$ ) and task-specific metrics. Compared to MLE rewards, which, as expected, are not strongly correlated with any metric, we see a clear increase in correlation pointing to IQLearn incorporating the task-relevant quality into the extracted rewards. We find that using online data is important for consistent correlations across all tasks, in particular as we evaluate over agent generated rollouts. We hypothesize that the comparably lower correlations for GSM8k are likely to be explained by the task’s idiosyncratic metric: only the correctness of the final generated numerical expression affects the accuracy calculation, effectively ignoring most of the trajectory (and

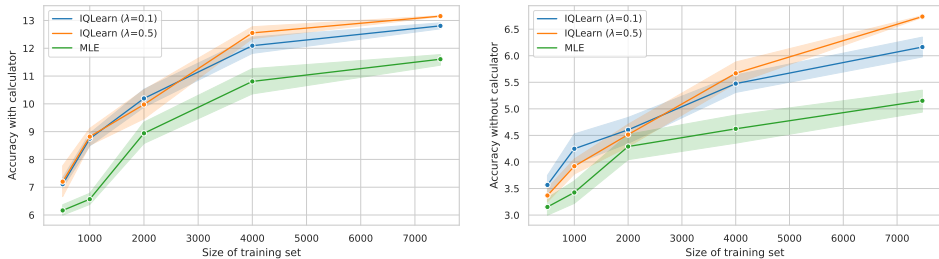


Figure 6: Different subsets of GSM8k. Performance gains persist across dataset scales with larger datasets demonstrating minimal preference for larger regularization coefficients. Each experiment was run with 3 random seeds to obtain uncertainty estimates.



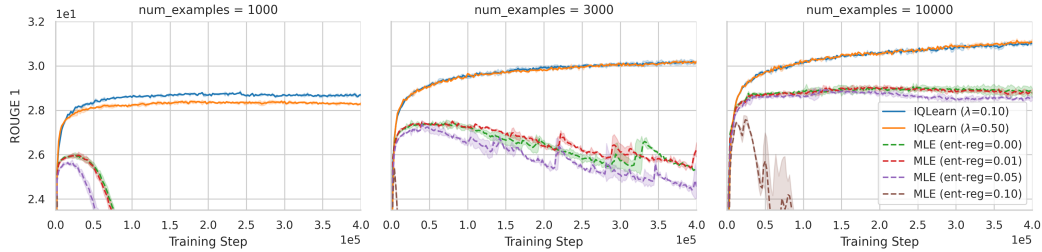


Figure 7: Learning curves for subsets of the XSUM training data. The smallest subsets demonstrate strong overfitting for pure MLE which the TD regularization in IQLearn mitigates. Pure entropy regularization is unable to obtain similar robustness and directly conflicts with task performance.

so its transition’s rewards) up to a specific answer segment. This highly targeted reward function becomes harder to learn. Finally, Figure 8 displays how the reward alignment with the task metrics increases with larger TD regularization  $\lambda$ .

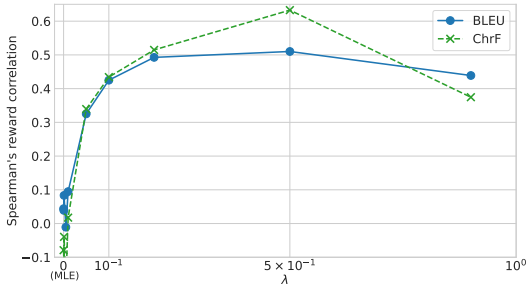


Figure 8: Reward correlation on WMT22 as a function of  $\lambda$  for a fixed mix-in  $\alpha = 0.1$  for online data.

Task	Metric	IQLearn	MLE
TLDR	ROUGE-1	0.64	-0.05
	ROUGE-2	0.41	-0.04
	ROUGE-Lsum	0.65	-0.05
WMT22	BLEU	0.43	-0.05
	ChrF	0.43	-0.01
GSM8k	Acc. w/ calculator	0.17	-0.02
	Acc. w/o calculator	0.17	0.04

Table 2: The Spearman’s rank correlation for online IQLearn ( $\alpha = 0.1$ ) with  $\lambda = 0.1$  (for GSM8k, WMT22) and  $\lambda = 0.5$  (for TLDR), compared to MLE (i.e.  $\lambda = 0.0$ ).

## 4 Related Work

**General imitation learning.** Imitation learning assumes a dataset of expert demonstrations, and the aim is to train a policy that matches the expert. There are two broad categories of imitation learning approaches: behavioral cloning (BC) and inverse reinforcement learning (IRL). In BC, a policy is trained using regression to directly mimic the expert demonstrations [49]. This is analogous to supervised fine-tuning of language models via MLE. BC requires sufficient data coverage to perform well, and suffers from compounding errors at evaluation time, as the policy deviates from the state distribution covered by the demonstrations. This can be alleviated by additionally querying the expert for the correct action in the states visited by the agent [56].

**Inverse reinforcement learning.** In contrast, IRL jointly infers the policy and reward function, such that the provided expert demonstrations are optimal under the reward function, and the learned policy maximizes this reward function [44]. By using additional environment interactions beyond the demonstrations, IRL can in theory overcome the compounding errors observed with BC [70]. Note that IRL is related to RL from human feedback (RLHF) but differs in key aspects: IRL also learns both a reward and policy, but extracts information from demonstration and agent data rather than paired preference data [31]. The game-theoretic approach to IRL treats the optimization problem as a zero-sum two-player game [60]. A subset of recent IRL methods can be seen as combining game-theoretic IRL with entropy regularization of the policy, where the doubly-nested optimization is implemented with a classifier (GAIL [25], DAC [35], AIRL [19]), implicit reward functions (ValueDICE [36], IQLearn [20]), or a Lagrangian dual objective (PPIL [62]). The stable, large-scale application of inverse RL methods has been a persistent goal throughout these developments. The classical requirement for complete RL optimization before updating the reward function has presented a limitation [75] which can be overcome via abstracted [68, 9] or linearized models [18], iterative adversarial training [25, 19] or lastly saddlepoint-based value function based formulations [20]. We expand on the insights of the latter to evaluate the competitive performance of computationally cheap offline IRL and emphasise the connection between MLE and IRL [48].

**Imitation learning for language modeling.** Understanding language modeling as an imitation problem has been previously explored. Indeed, the link can already be made from MLE, commonly referred to as Behavioral Cloning (BC) [8] from an imitation perspective. Although the link to imitation has been made explicit recently, either theoretically [59], or in the case of distillation and divergence minimisation [1, 28, 40, 26, 39], some works had already tackled the issues of MLE with imitation-like techniques. For example, adversarial training of text generation an alternative to MLE was first proposed in SeqGAN [71], and followed-up by a series of work using GANs for text generation [32]. These methods have been shown to work only in the temperature 1 regime [11], a possible shortcoming that we address in Section 3. Then, leveraging the literature of imitation learning, GAIL was successfully adapted to language [65], showing an improvement over MLE. Closer to our contributions, IQLearn is utilized for language in SequenceMatch [15]. Key differences to our work include the reformulation as temporal difference regularized MLE, large-scale analysis including other inverse RL methods and focus on computational costs via the application of offline IQLearn. Indeed, SequenceMatch requires the use of online data, via the introduction of the "backward" token, that allows the model to change a previously chosen token during sampling.

## 5 Discussions

Our investigation focuses on diversity measures such as Self-BLEU or model entropy which are easily calculable but limited with respect to their ability to describe the impact on later training stages. Future evaluation and practical application will demonstrate if the increased diversity is relevant to RLHF such as for human raters in preference data evaluation or improved exploration during subsequent RL optimization [53].

The field of imitation learning has led to a gamut of algorithms, many of which are intuitively simple to implement with existing RL or RLHF infrastructure (e.g. [54, 63, 24, 50]). Ease of adaptation and hyperparameter tuning have principal impact on our practical algorithm choices and the methods and extensions discussed in this work enabled quick first results and iteration. Looking forward, the evaluation and adaptation of further imitation learning methods to LLMs is likely to lead to fruitful results in the coming years. While our analysis focuses on specific algorithms, our key arguments should be seen in the light of the benefits of underlying mechanisms rather than the specific methods.

The sampling-free application of RL mechanism can eventually extend to even larger datasets such as pretraining data, domains with high requirements for computational efficiency. While these datasets are, in a certain sense, less optimal and representative of preferred model behavior, different aspects of our analysis have the potential to generalize, such as increased efficiency for modelling sequential decision making or increased diversity of model generations. Pretraining and other sub-optimal data could further be used as additional data source for IQLearn and related algorithms [41, 33, 57] instead of autoregressive sampling data.

Finally, RLHF’s key role lies in the alignment of models with respect to user preferences. By integrating SFT data into RL-based optimization, we hope to expand data used for preference description from paired comparisons to individual demonstrations. Integrating generative and discriminative information from different data sources into a unified RL-based framework for unified LLM post-training has further practical potential as previously demonstrated for the generative side [46] with first steps towards reward transfer [38]. Our reward analysis in Section 3 provides an initial signal about IRL-extracted reward functions including crucial information about LLM task performance.

## 6 Conclusions

This paper presents a detailed investigation of the potential of IRL algorithms for imitation in language model tuning focusing on performance, diversity, and computational requirements. We introduce a reformulation of IQLearn which enables principled interpolation between robust, standard supervised fine-tuning and more effective IRL algorithms. Our experiments demonstrate particularly strong gains for IRL on the Pareto front of task performance and diversity of model generations. While prior work primarily focused on online IRL, we demonstrate that computationally cheaper offline IRL, without the requirement of online sampling, already obtains crucial performance gains over MLE-based optimization. Additional correlation analysis between IRL-extracted rewards and performance metrics further emphasises the potential to obtain more accurate and robust reward function for language modelling. We hope this work will help to pave the way for better compromises between data and compute efficiency via RL-based algorithms across the complete LLM training pipeline.

## Acknowledgments and Disclosure of Funding

We extend our gratitude to Daniele Calandriello, Doina Precup, and our anonymous reviewers for their invaluable feedback on the final manuscript drafts. We also would like to acknowledge the contributions of our internal engineering teams, whose robust and flexible RLHF infrastructure significantly accelerated algorithmic development and evaluations. Special thanks are due to Piotr Stanczyk and Leonard Hussenot for their swift and effective solutions to numerous infrastructure challenges. Finally, we deeply appreciate the strategic feedback and guidance provided by Satinder Singh, Dale Schuurmans, and David Silver.

## References

- [1] Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, 2024. 10
- [2] Firas Al-Hafez, Davide Tateo, Oleg Arenz, Guoping Zhao, and Jan Peters. Ls-iq: Implicit reward regularization for inverse reinforcement learning. *arXiv preprint arXiv:2303.00599*, 2023. 16
- [3] Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyeontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussalem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta, Rajkumar Samuel, Renee Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniel Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Kelvin Xu, Yunhan Xu, Linting Xue, Pengcheng Yin, Jiahui Yu, Qiao Zhang, Steven Zheng, Ce Zheng, Weikang Zhou, Denny Zhou, Slav Petrov, and Yonghui Wu. PaLM 2 technical report, 2023. 2, 6, 7
- [4] Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and autonomous systems*, 57(5):469–483, 2009. 1
- [5] Kushal Arora, Layla El Asri, Hareesh Bahuleyan, and Jackie Chi Kit Cheung. Why exposure bias matters: An imitation learning perspective of error accumulation in language generation. In *Findings*, 2022. 2
- [6] Gregor Bachmann and Vaishnavh Nagarajan. The pitfalls of next-token prediction. *ArXiv*, abs/2403.06963, 2024. 2
- [7] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022. 2
- [8] Michael Bain and Claude Sammut. A framework for behavioural cloning. In *Machine Intelligence 15*, pages 103–129, 1995. 2, 10
- [9] Matt Barnes, Matthew Abueg, Oliver F. Lange, Matt Deeds, Jason Trader, Denali Molitor, Markus Wulfmeier, and Shawn O’Banion. Massively scalable inverse reinforcement learning in google maps. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 9

- [10] Dan Biderman, Jose Gonzalez Ortiz, Jacob Portes, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, Cody Blakeney, and John P. Cunningham. Lora learns less and forgets less, 2024. 6
- [11] Massimo Caccia, Lucas Caccia, William Fedus, Hugo Larochelle, Joelle Pineau, and Laurent Charlin. Language gans falling short. *arXiv preprint arXiv:1811.02549*, 2018. 10
- [12] Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, J’er’emy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Wang, Samuel Marks, Charbel-Raphaël Ségerie, Micah Carroll, Andi Peng, Phillip J. K. Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, Anand Siththaranjan, Max Nadeau, Eric J. Michaud, Jacob Pfau, Dmitrii Krasheninnikov, Xin Chen, Lauro Langosco di Langosco, Peter Hase, Erdem Biyik, Anca D. Dragan, David Krueger, Dorsa Sadigh, and Dylan Hadfield-Menell. Open problems and fundamental limitations of reinforcement learning from human feedback. *ArXiv*, abs/2307.15217, 2023. 1
- [13] Paul Christiano, Jan Leike, Tom B Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *arXiv preprint arXiv:1706.03741*, 2017. 1
- [14] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. 6, 7
- [15] Chris Cundy and Stefano Ermon. Sequencematch: Imitation learning for autoregressive sequence modelling with backtracking, 2024. 6, 8, 10, 16
- [16] Anna Dawid and Yann LeCun. Introduction to latent variable energy-based models: A path towards autonomous machine intelligence. *ArXiv*, abs/2306.02572, 2023. 2
- [17] Gemini Team et al. Gemini: A family of highly capable multimodal models, 2024. 1
- [18] Chelsea Finn, Sergey Levine, and P. Abbeel. Guided cost learning: Deep inverse optimal control via policy optimization. *ArXiv*, abs/1603.00448, 2016. 9
- [19] Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adversarial inverse reinforcement learning. *arXiv preprint arXiv:1710.11248*, 2017. 9
- [20] Divyansh Garg, Shuvam Chakraborty, Chris Cundy, Jiaming Song, Matthieu Geist, and Stefano Ermon. Iq-learn: Inverse soft-q learning for imitation, 2022. 2, 3, 4, 5, 9, 16
- [21] Shivam Garg, Dimitris Tsipras, Percy S Liang, and Gregory Valiant. What can transformers learn in-context? a case study of simple function classes. *Advances in Neural Information Processing Systems*, 35:30583–30598, 2022. 16
- [22] Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized markov decision processes. *CoRR*, abs/1901.11275, 2019. 4
- [23] Seyed Kamyar Seyed Ghasemipour, Richard S. Zemel, and Shixiang Gu. A divergence minimization perspective on imitation learning methods. *CoRR*, abs/1911.02256, 2019. 2, 3
- [24] Joey Hejna and Dorsa Sadigh. Inverse preference learning: Preference-based rl without a reward function. *ArXiv*, abs/2305.15363, 2023. 10
- [25] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. *Advances in neural information processing systems*, 29:4565–4573, 2016. 2, 3, 4, 5, 9
- [26] Luca Hormann and Artem Sokolov. Fixing exposure bias with imitation learning needs powerful oracles. *arXiv preprint arXiv:2109.04114*, 2021. 10
- [27] 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, 2021. 6
- [28] Rebekka Hubert, Artem Sokolov, and Stefan Riezler. Improving end-to-end speech translation by imitation-based knowledge distillation with synthetic transcripts. In *Int. Workshop on Spoken Language Translation*, 2023. 10

- [29] Ahmed Hussein, Mohamed Medhat Gaber, Eyad Elyan, and Chrisina Jayne. Imitation learning: A survey of learning methods. *ACM Comput. Surv.*, 50(2), apr 2017. 1
- [30] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12), mar 2023. 2
- [31] Timo Kaufmann, Paul Weng, Viktor Bengs, and Eyke Hüllermeier. A Survey of Reinforcement Learning from Human Feedback, 2023. 9
- [32] Pei Ke, Fei Huang, Minlie Huang, and Xiaoyan Zhu. Araml: A stable adversarial training framework for text generation. *arXiv preprint arXiv:1908.07195*, 2019. 10
- [33] Geon-Hyeong Kim, Seokin Seo, Jongmin Lee, Wonseok Jeon, HyeongJoo Hwang, Hongseok Yang, and Kee-Eung Kim. Demodice: Offline imitation learning with supplementary imperfect demonstrations. In *International Conference on Learning Representations*, 2022. 10, 16
- [34] Tom Kocmi, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Thamme Gowda, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Rebecca Knowles, Philipp Koehn, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Michal Novák, Martin Popel, and Maja Popović. Findings of the 2022 Conference on Machine Translation (WMT22). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, 2022. 6, 7
- [35] Ilya Kostrikov, Kumar Krishna Agrawal, Debidatta Dwibedi, Sergey Levine, and Jonathan Tompson. Discriminator-actor-critic: Addressing sample inefficiency and reward bias in adversarial imitation learning. In *International Conference on Learning Representations*, 2019. 9
- [36] Ilya Kostrikov, Ofir Nachum, and Jonathan Tompson. Imitation learning via off-policy distribution matching. *arXiv preprint arXiv:1912.05032*, 2019. 9
- [37] Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. *arXiv*, abs/2309.00267, 2023. 5
- [38] Jiayang Li, Siliang Zeng, Hoi-To Wai, Chenliang Li, Alfredo García, and Mingyi Hong. Getting more juice out of the sft data: Reward learning from human demonstration improves sft for llm alignment. *ArXiv*, abs/2405.17888, 2024. 10
- [39] Ziniu Li, Congliang Chen, Tian Xu, Zeyu Qin, Jiancong Xiao, Ruoyu Sun, and Zhimin Luo. Entropic distribution matching in supervised fine-tuning of llms: Less overfitting and better diversity, 2024. 10
- [40] Alexander Lin, Jeremy Wohlwend, Howard Chen, and Tao Lei. Autoregressive knowledge distillation through imitation learning. *arXiv preprint arXiv:2009.07253*, 2020. 10
- [41] Yecheng Ma, Andrew Shen, Dinesh Jayaraman, and Osbert Bastani. Versatile offline imitation from observations and examples via regularized state-occupancy matching. In *International Conference on Machine Learning*. PMLR, 2022. 10, 16
- [42] Ofir Nachum, Mohammad Norouzi, Kelvin Xu, and Dale Schuurmans. Bridging the gap between value and policy based reinforcement learning. In *Advances in Neural Information Processing Systems*, 2017. 4
- [43] 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. *ArXiv*, abs/1808.08745, 2018. 6
- [44] Andrew Y. Ng and Stuart J. Russell. Algorithms for inverse reinforcement learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML ’00, page 663–670, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc. 9
- [45] OpenAI. Gpt-4 technical report, 2023. 1

- [46] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022. 1, 5, 10
- [47] Krishna Pillutla, Lang Liu, John Thickstun, Sean Welleck, Swabha Swayamdipta, Rowan Zellers, Sewoong Oh, Yejin Choi, and Zaïd Harchaoui. Mauve scores for generative models: Theory and practice. *ArXiv*, abs/2212.14578, 2022. 17
- [48] Bilal Piot, Matthieu Geist, and Olivier Pietquin. Boosted and reward-regularized classification for apprenticeship learning. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-Agent Systems, AAMAS '14*, page 1249–1256, Richland, SC, 2014. International Foundation for Autonomous Agents and Multiagent Systems. 4, 9
- [49] Dean A Pomerleau. Alvin: An autonomous land vehicle in a neural network. *Advances in neural information processing systems*, 1, 1988. 9
- [50] Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From r to q\*: Your language model is secretly a q-function, 2024. 10
- [51] 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, 2023. 2, 6
- [52] Marc’Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks, 2016. 2
- [53] Noam Razin, Hattie Zhou, Omid Saremi, Vimal Thilak, Arwen Bradley, Preetum Nakkiran, Joshua Susskind, and Etai Littwin. Vanishing gradients in reinforcement finetuning of language models. *arXiv preprint arXiv:2310.20703*, 2023. 10
- [54] Siddharth Reddy, Anca D. Dragan, and Sergey Levine. Sqil: Imitation learning via reinforcement learning with sparse rewards, 2019. 10
- [55] Juntao Ren, Gokul Swamy, Zhiwei Steven Wu, J. Andrew Bagnell, and Sanjiban Choudhury. Hybrid inverse reinforcement learning. *ArXiv*, abs/2402.08848, 2024. 19
- [56] Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011. 2, 8, 9
- [57] Harshit Sikchi, Qinqing Zheng, Amy Zhang, and Scott Niekum. Dual rl: Unification and new methods for reinforcement and imitation learning. In *International Conference on Learning Representations*, 2023. 4, 10, 16
- [58] 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*, 2020. 6, 7
- [59] Hao Sun. Supervised fine-tuning as inverse reinforcement learning. *arXiv preprint arXiv:2403.12017*, 2024. 10
- [60] Umar Syed and Robert E. Schapire. A game-theoretic approach to apprenticeship learning. In *Advances in Neural Information Processing Systems*, 2007. 9
- [61] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shrutu Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 2
- [62] Luca Viano, Angeliki Kamoutsi, Gergely Neu, Igor Krawczuk, and Volkan Cevher. Proximal point imitation learning. In *Advances in Neural Information Processing Systems*, 2022. 9
- [63] Joe Watson, Sandy H. Huang, and Nicolas Heess. Coherent soft imitation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 10

- [64] Ronald J. Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1:270–280, 1989. 2
- [65] Qingyang Wu, Lei Li, and Zhou Yu. Textgail: Generative adversarial imitation learning for text generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 2021. 6, 10, 19
- [66] Markus Wulfmeier, Arunkumar Byravan, Sarah Bechtle, Karol Hausman, and Nicolas Manfred Otto Heess. Foundations for transfer in reinforcement learning: A taxonomy of knowledge modalities. *ArXiv*, abs/2312.01939, 2023. 1
- [67] Markus Wulfmeier, Ingmar Posner, and P. Abbeel. Mutual alignment transfer learning. *ArXiv*, abs/1707.07907, 2017. 5
- [68] Markus Wulfmeier, Dominic Zeng Wang, and Ingmar Posner. Watch this: Scalable cost-function learning for path planning in urban environments. *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2089–2095, 2016. 9
- [69] Yue Xing, Qifan Song, and Guang Cheng. On the algorithmic stability of adversarial training. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 26523–26535. Curran Associates, Inc., 2021. 5
- [70] Tian Xu, Ziniu Li, and Yang Yu. Error bounds of imitating policies and environments. *Advances in Neural Information Processing Systems*, 2020. 9
- [71] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017. 10
- [72] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *ArXiv*, abs/1904.09675, 2019. 17
- [73] Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. Taxygen: A benchmarking platform for text generation models. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 1097–1100, 2018. 6
- [74] Brian D Ziebart. *Modeling Purposeful Adaptive Behavior with the Principle of Maximum Causal Entropy*. PhD thesis, Carnegie Mellon University, 2010. 4
- [75] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, Anind K Dey, et al. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008. 9

## A Appendix

### A.1 Implementation Details

#### A.1.1 IQLearn

For IQLearn, fine-tuning starts with a policy which is subsequently finetuned as a Q-function. In order to do so, we take the logits underlying the LLM softmax layer and continue training as Q-values. In entropy regularized RL, we can build on the equality between optimal policy  $\pi^*$  and optimal Q-function  $Q^*$  via  $\pi^*(a|s) = 1/Z_s \exp Q^*(s, a)$  with normalization factor  $Z_s = \sum_{a' \in A} \exp Q^*(s, a')$ . We further obtain  $V$  via  $V(s) = \log \sum_{a \in A} \exp Q(s, a)$ .

Due to the translation invariance of the softmax function, there can be a considerable state-based offset between initial logits and final values. In other words, with  $Q(s, a) = V(s) + A(s, a)$  only the advantage function  $A$  has to be accurately reflected by the initial policy logits, while the state-based value  $V$  can be arbitrarily inaccurate. In practice, we can add further KL regularization, or separate the representation of state and advantage function, to stabilize training during the additional identification of correct offsets but found it to be unnecessary for our experiments in comparison to recent work [15].

We handle terminal states by setting their values to zero in line with the original IQLearn paper [20]. Improvements such as learned values for the terminal states [2, 15] did not contribute to improved performance in this setting.

Table 3: IQLearn and MLE hyperparameters

Hyperparameter	Value
Learning rate T5	1e-4
Learning rate PaLM2	1e-4
Warmup steps	2000
Batch size T5 (base/large/xl)	32/32/16
Batch size PaLM2	16
Random seeds / experiment	3

#### A.1.2 Online IQLearn

We also implemented an online version of IQLearn, which makes use of additional (non-expert) samples. In contrast to IQLearn [21] which makes use of these examples to estimate the initial value,  $E_{\rho_0} v(s)$  (which is equivalent between online and offline data when the prompt is the same as in our case), we integrate the additional samples to relax the distribution matching loss:

$$\min_{\pi} D_f((1 - \alpha)\mu + \alpha\mu_B || (1 - \alpha)\mu_E + \alpha\mu_B) - \lambda H(\pi), \tag{15}$$

where  $\mu_B$  is the discounted state-action distribution of the additional samples and where  $\alpha \in [0, 1]$  is the strength of the mix-in. The problem is thus relaxed and allows the policy to match the mix-in distribution  $\mu_B$  in case the expert is too difficult to match.

From the adapted distribution matching loss the same steps can then be taken as in Section 2. This results in the generalised loss:

$$- \min_{v^r, \pi^r} E_{(1-\alpha)\mu_E + \alpha\mu_B} [f^*(-r(v^r, \pi^r)) + r(v^r, \pi^r)] - E_{\mu_E} \lambda \log(\pi^r). \tag{16}$$

with

$$r(v^r, \pi^r) = v^r + \frac{\lambda}{1 - \alpha} \log \pi^r - E_{s'} v'^r \tag{17}$$

This formulation strongly relates to the offline version of the algorithm but additionally applies the temporal difference based regularization term on the additional non-expert transitions. A similar algorithm is derived in Appendix D.1 [57] with strong results on classical, continuous control domains of its state-action value function based version. It further connects to prior methods enabling the integration of sub-optimal data into imitation learning [41, 33] with a useful overview on the broader field of related methods in [57].



### A.1.3 GAIL

For GAIL, both the policy and discriminator are represented via separate networks initialized from the initial, pre-trained, LLM. Policy optimization is performed similar to PPO with an A2C update, with re-scaled advantage and KL constraint to the initial policy. The KL constraint has a weight hyperparameter that is annealed over 10,000 steps to a final cost weight displayed below. The value network is also initialized from the initial LLM. The discriminator is trained with a cross-entropy objective. The reward is re-shaped from the the discriminator output to be a positive reward:  $r_t = \log(1 + \exp(D(s_t)))$ .

We explore the heuristic combination of GAIL with standard MLE training by using a weighted combination of GAIL and MLE losses for the policy. While this is not required for GSM8k, it leads to considerable improvements for XSUM.

Policy, value function and discriminator are all updated with the Adam optimizer, with a constant learning rate and linear warm-up. The discriminator is updated after every step of policy optimization.

Table 4: GAIL hyperparameters

Method	T5-base	T5-large	T5-xl
Batch size	32	32	16
Learning rate	1e-4	1e-4	1e-4
Warmup steps	2000	2000	2000
KL strength	1e-3	1e-3	1e-3
Random seeds / experiment	3	3	3

## A.2 MLE & Entropy Regularization

As mentioned in the experiments section, we compare our methods to an entropy regularized version of MLE, to disentangle between the imitation contribution and the regularization. This algorithm simply follows the objective

$$\min_{\pi} E_{\mu_E}[-\log(\pi) - \lambda \mathcal{H}(\pi)], \tag{18}$$

where we compute the entropy at each token of the sequences from  $\mu_E$ .

### A.2.1 Computational Requirements

Our experiments with T5 models use TPU v3 infrastructure and are running between approximately 3 days and 2 weeks. Our experiments with PaLM2 models use TPU v4 infrastructure and are running under 1 week.

## A.3 Additional Experiments

We include a set of further experiments to complement the results in the main paper.

### A.3.1 Quality-Diversity Evaluation

In addition to the Self-BLEU metric, we further add plots for model entropy and add the ROUGE-LSUM results in Figure 9. Further performance metrics like MAUVE [47] and BertScore [72] can be of use in the future to further represent human judgement and preferences.

### A.3.2 PALM2 Additional Results

We complement results on the temperature sweep over PALM2 models in Figure 10, with additional metrics for GSM8k and TLDR (accuracy with calculator and rouge scores). This confirms the results of the main paper experiment, showing that IQLearn can consistently outperform MLE in accuracy, even when sampling with a lower temperature than the training one.

### A.3.3 Toy Experiments for Offline and Online IRL with Autoregressive Generation

The toy scenario displayed by the MDP in Figure 11 is used to represent a key difference between many classical control settings and autoregressive language generation. The agent starts in the left

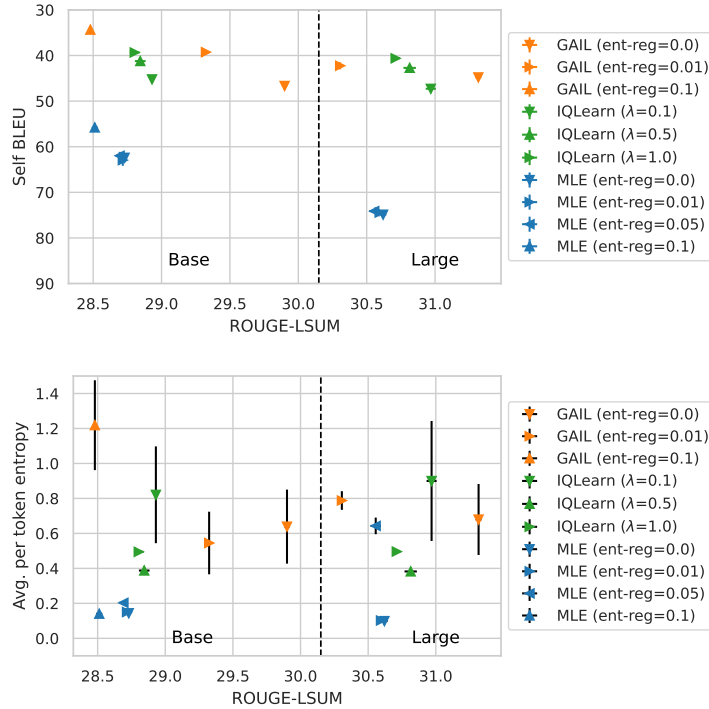


Figure 9: XSUM results for models trained with MLE, IQLearn, and GAIL across different regularization strengths. Top: we show ROUGE-LSUM performance metric on the x-axis and Self-BLEU diversity measure on the y-axis. Bottom: with the per-token entropy as diversity metric. Error bars indicate the standard error of the mean after repeating the experiment with 3 different seeds. Compare to Figure 3 for more results.

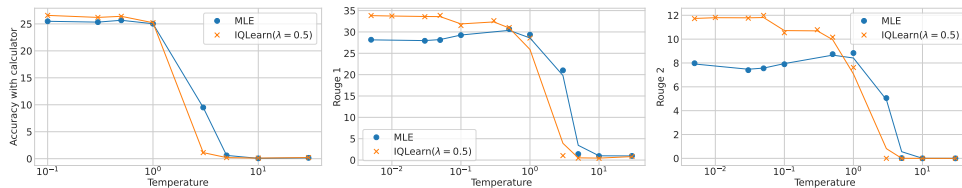


Figure 10: Additional scores against temperature for PALM2 models. From left to right: GSM8k, TDLR (ROUGE-1) and TLDR (ROUGE-2).

black state and has to reach the green on the right. In each of the bottom states the agent has 2 actions, move left or move up. The transition dynamics are noisy so that the agent executes the unwanted action 10% of the time. There are two variants of the MDP, one where the agent can return from the top states by learning to execute the right action and one where it cannot. The latter represents one aspect of concatenation dynamics, the agent cannot return to the exact same state after sampling the wrong action. In a way, it cannot correct its behavior exactly.

When training offline and online variants IQLearn in this setting with demonstrations without mistakes (and their correction), we clearly see that the online version of IQLearn outperforms offline learning by over 11% in success rates, while in the setting without recovery, the difference is considerably smaller and results are within each others confidence bounds with more variance for the offline agent.

### A.3.4 Analyzing GAIL Stabilization

We empirically find GAIL overall more complex to tune and stabilize and aim to provide further insights here. Intuitively, control for language modelling differs from many classical applications via large discrete action space and terminations not being state but action conditioned. In other words, the agent can directly chose to terminate. Therefore the agent can learn to directly terminate if

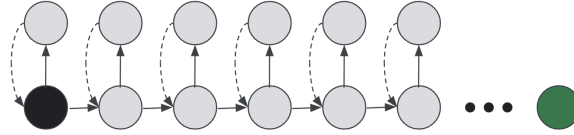


Figure 11: Simplified MDP to represent characteristics of concatenation-based autoregressive generation in comparison to many classical control domains. Dashed lines visualize the potential to return or correct mistakes, missing from autoregressive generation. The agent starts in the black state and has to complete the sequence to reach green to receive rewards.

the discriminator provides negative rewards. If we provide positive rewards, tuning those rewards becomes a complex task as seen in Figure 12, where without additional MLE objective the GAIL agent often uses the maximum sample length with correlated loss of performance (visualized by the drop on ROUGE-1 values). Additional MLE training can help to both shape policy behavior but further also provide more relevant data for the discriminator [55]. Especially in high dimensional action spaces it otherwise becomes challenging to obtain a useful reward signal from the discriminator as seen in prior work [65].

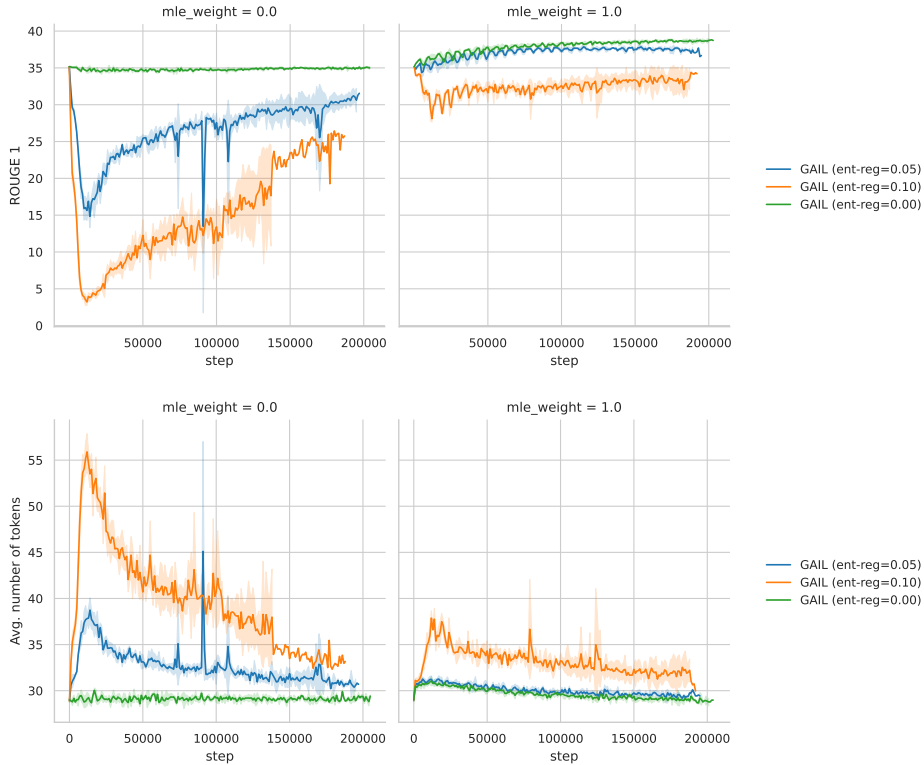


Figure 12: Effect of adding a standard MLE loss (`mle_weight=1`) on the training data combined with GAIL on XSUM. We show the ROUGE 1 metric and the average length of the generated summaries when training a T5-Large model with GAIL.

#### A.4 Expanded Experiments on WMT22

For WMT22, we additionally evaluate IQLearn performance for a wide range of  $\lambda$  values. Here, IQLearn consistently gains (up to +2.16 BLEU) over BC until very high values of  $\lambda$  (Table 5). Note, that unlike the sampling performance curves in Figure 4, here beam search decoding (size 4) is used with a length penalty 0.6.

Table 5: WMT22 results for offline IQLearn initialised with a PaLM2 checkpoint. *Italic* – best dev BLEU in group (i.e. same mix value), **bold** – best overall.

$\lambda$	<b>mixin</b>	<b>dev-BLEU</b>	<b>test-BLEU</b>
0.0	0.0	26.92	32.34
0.05	0.0	29.14	34.84
0.1	0.0	28.93	34.51
0.3	0.0	29.07	34.79
0.5	0.0	29.20	34.63
0.7	0.0	28.89	35.68
0.9	0.0	28.89	34.35
1.0	0.0	28.92	33.69
0.0	0.1	27.43	32.77
0.05	0.1	29.43	34.50
0.1	0.1	29.23	34.56
0.3	0.1	28.89	34.84
0.5	0.1	28.95	34.39
0.7	0.1	28.78	34.17
0.9	0.1	28.83	34.41
1.0	0.1	28.73	34.29
0.0	0.2	27.10	32.31
0.05	0.2	29.08	34.40
0.1	0.2	<b>29.46</b>	34.54
0.9	0.2	28.85	33.99

## NeurIPS Paper Checklist

### 1. **Claims**

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: The abstract provides key information about methods and results.

### 2. **Limitations**

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: Parts of the discussion section explicitly discuss limitations of the performed work.

### 3. **Theory Assumptions and Proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: All methods are properly derived.

### 4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Experimental recreation of the results might be limited by required computational resources. Other than that, algorithms are accurately described including hyperparameters and the T5 model classes should enable the recreation of results.

### 5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: We have been unable to provide public access to our code.

### 6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: All experimental detailed information is either given in the main paper or more likely in the appendix.

### 7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Where computationally possible, the experiments include error bars and related information.

### 8. **Experiments Compute Resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Approximate details are given in the appendix

**9. Code Of Ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The NeurIPS Code of Ethics has been read by the authors and all applicable areas have been verified.

**10. Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: This paper includes fundamental research on imitation learning with respect to the application to language modelling and does not itself introduce new risks and concerns in addition to those related to existing optimisation methods for language models.

**11. Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: No data or model are publicly released.

**12. Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: References are included for all datasets and models.

**13. New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: We release no new assets.

**14. Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing or research with human subjects.

**15. Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not include research with human subjects.