Learning to Teach for Distributional Robustness: Reinforcement Learning for Universal Curriculum Design Across Tasks and Datasets

Anonymous Authors

Abstract

In practice, models intended for a small, high-stakes domain are trained alongside much larger auxiliary datasets. This "many-to-one" regime accelerates learning but can mask failure on the target domain: for example, a mixture in which roughly 85–90% of training examples come from a dominant source and the remaining 10–15% are spread across specialized datasets can yield strong average accuracy while performing poorly on the minority slices that matter. We revisit *learning to teach* and design a reinforcement-learning teacher that explicitly guards against this imbalance by adaptively controlling three levers of the curriculum: the per-dataset mixture, the learning rate, and the lesson size (budget usage). Our formulation makes the teacher–student interaction *Markov* by grounding transitions in SGD dynamics and exposing a compact, task/architecture-agnostic observation with three groups: (A) **dataset-level DeepSets embeddings** computed from permutation-invariant probe statistics (loss/uncertainty summaries, gradient-norm moments, and diversity sketches), (B) a **model-complexity** vector, and (C) a **training-progress** vector. A set-based encoder with mean pooling and shared per-dataset policy heads ensures *number-of-datasets invariance*, producing valid mixtures for any N and any permutation, while a constrained-MDP view manages total budget. We validate *approximate Markovity* via one-step prediction R^2 with only small gains from short histories, demonstrate invariance under varying N and permutations, and show cross-task/architecture portability (e.g., classification, detection, sequence labeling, QA), including leave-one-family-out and zero/few-shot transfer from toy episodes to large models. Despite its compactness, the teacher achieves strong sample efficiency and improves balanced metrics (macro-accuracy and worst-dataset performance) across heterogeneous mixtures.

1 Introduction

In production pipelines, a model intended for a narrow, high-stakes domain is often trained with help from many auxiliary datasets. This "many-to-one" regime is pragmatic—large sources accelerate learning—but brittle: when evaluation or deployment hinges on a small curated dataset, optimizing an aggregate objective can obscure failures on that minority slice. In extremis, if a single dataset contributes 99% of all training examples, a seemingly strong mean accuracy can coexist with poor performance on the 1% that matters. This phenomenon mirrors the *repeated loss minimization* and *group shift* observations in the fairness and robustness literature: naively minimizing average risk can systematically underserve minority groups [Hashimoto et al., 2018, Sagawa et al., 2020].

We study *learning to teach*: a reinforcement-learning (RL) *teacher* that observes a student's competence and adaptively decides *which datasets* to emphasize, *how aggressively* to spend budget, and *what learning rate* to use. The goal is to maximize balanced end-of-training metrics—e.g., *macro-accuracy* across N datasets or worst-dataset performance—so that small, specialized datasets are not eclipsed by large, easy ones.

A central challenge is engineering the teacher's interface so that it is (1) *Markov*, enabling stable RL control; (2) *invariant in the number of datasets*, so one policy naturally scales to any N and any permutation; and (3) *portable* across task families and architectures, so a single teacher trained on small, inexpensive episodes can be deployed in varied settings. Existing learning-to-teach approaches demonstrate the promise of dynamic data selection and learned losses [Fan et al., 2018, Wu et al., 2018], yet typically do not combine a formal Markov interface with a *set-based* design for many datasets and a compact, task-agnostic observation that can transfer broadly. Meanwhile, results in group-robust optimization argue for objectives that explicitly guard minority performance [Hashimoto et al., 2018, Sagawa et al., 2020]; we bring that spirit to curriculum design by reallocating training mixture and budget over time.

Our approach. We formalize the hidden state of training under SGD and construct a grouped, task/architecture/optimizer-agnostic observation with three components: *(A) dataset-level embeddings* built via DeepSets from small, fixed probes (loss/uncertainty histograms, gradient-norm moments, diversity sketches, light meta-statistics), *(B) model-complexity* summaries, and *(C) training-progress* dynamics. A set-based encoder with mean pooling ensures permutation and cardinality invariance; shared per-dataset policy heads emit mixture logits w_t for any N. Additional heads control learning rate and usage, allowing the teacher to modulate *both* exposure and step size. We adopt a CMDP view over budget.

Contributions. (1) A Markov teacher–student formulation with a *grouped, universal state* (dataset embeddings + model complexity + training progress) that is compact yet empirically *approximately* Markov. (2) A *set-invariant* policy/value parameterization whose per-dataset heads and mean-pooled encodings produce valid mixtures for any N and any permutation. (3) A meta-training/evaluation protocol demonstrating *portability* across task families and architectures, including leave-one-family-out and zero-/few-shot transfer. (4) A CMDP treatment that jointly controls mixture, learning rate, and usage, improving balanced outcomes (macro and worst-dataset metrics) over uniform sampling, myopic/bandit schedulers, and heuristic curricula. Conceptually, our teacher operationalizes the group-robust principle—do not ignore small groups—through an adaptive curriculum rather than a static objective.

Why this matters. Practitioners increasingly face mixtures where large auxiliary datasets threaten to dominate optimization while deployment risk concentrates on a small, specialized dataset. A teacher that is *Markov*, *set-invariant*, and *task/architecture-invariant* can be trained once and reused, reallocating budget toward underperforming datasets as training unfolds. Our results suggest that a compact, grouped observation suffices to steer learning dynamics toward balanced performance, complementing distributionally robust objectives with a procedural lever: the curriculum itself.

2 Related Work

Learning to teach and curriculum RL. Classic curriculum and self-paced learning emphasize the sequencing of examples [Bengio et al., 2009, Kumar et al., 2010]. Fan et al. [2018] frame *Learning to Teach* as RL, selecting instances to present and reporting cross-architecture generalization. Wu et al. [2018] learn dynamic loss functions to guide optimization. These lines establish that teachers can shape learning trajectories, but typically focus on per-example selection or loss design *without* a formal Markov interface, an explicitly *set-invariant* treatment of many datasets, or a compact, task-agnostic observation sufficient for transfer across modalities.

Fairness and group-robust objectives. Optimizing average risk can degrade minority-group performance; Hashimoto et al. [2018] formalize this phenomenon in repeated loss minimization and propose a robust objective, and Sagawa et al. [2020] analyze distributionally robust neural networks under group shifts, highlighting regularization for worst-group performance. We share the goal of protecting small groups, but tackle it *procedurally*: by adaptively reweighting exposure (mixture), learning rate, and budget over time, rather than by altering the static loss. Our teacher can be layered atop standard objectives, widening applicability.

Multi-task scheduling and loss weighting. Methods that weight losses or balance gradients [Kendall et al., 2018, Chen et al., 2018] influence multi-task outcomes, yet do not explicitly reason about Markov state, budgeted control, or variable-cardinality sets. Our approach is complementary: we control *which* data the student sees and *how much* to train at each step, while such methods can still govern per-task losses.

Meta-learning and RL controllers. Meta-RL [Duan et al., 2016, Finn et al., 2017] and hyperparameter controllers (PBT/BOHB) [Jaderberg et al., 2017, Falkner et al., 2018, Snoek et al., 2012] optimize learning procedures or schedules. We differ in providing (i) a compact, task-agnostic, *group-sensitive* observation; (ii) *set-invariant* state/action encodings for any N; and (iii) an explicit CMDP view of budget, enabling balanced control of exposure and learning rate.

Permutation-invariant encoders for sets. DeepSets [Zaheer et al., 2017] and Set Transformer [Lee et al., 2019] provide universal forms for permutation-invariant/equivariant processing. We adopt these to encode dataset-level statistics and to parameterize per-dataset action heads, ensuring that the teacher's decisions are valid and stable for *any* number and ordering of datasets.

Position relative to prior work. Compared to Fan et al. [2018], Wu et al. [2018], we (i) formalize a Markov teacher–student MDP and empirically diagnose *approximate Markovity* of a compact *grouped* observation; (ii) enforce *number-of-datasets invariance* with set encoders and shared per-dataset heads; and (iii) target *task/architecture invariance* via task-agnostic statistics (probabilities, losses, gradient magnitudes, diversity sketches) plus model/progress blocks. Relative to fairness and GroupDRO [Hashimoto et al., 2018, Sagawa et al., 2020], we provide a complementary mechanism—*adaptive curricula*—that can protect minority datasets without modifying the base loss.

3 Problem Formulation

We consider N datasets $\{\mathcal{D}_i\}_{i=1}^{N}$ drawn from an episode-specific task $\tau \sim \mathcal{T}$. A student model f_θ is trained under a teacher policy π_ϕ over horizon H with initial budget B_0. Our objective emphasizes *balanced* end performance (macro- or worst-dataset metrics), reflecting the practical risk that small, high-stakes datasets be overshadowed by large auxiliaries.

3.1 Markov Teacher–Student MDP

Definition (Markov property). An RL environment with observable state S_t and action a_t is Markov if

$$\mathbb{P}(S_{t+1}, r_t \mid S_t, a_t) \;=\; \mathbb{P}(S_{t+1}, r_t \mid S_t, a_t, S_{t-1}, a_{t-1}, \dots). \tag{1}$$

Definition 1 (Approximate Markovity). *Let $\mathcal{F}$ be a rich predictor class. Define $\mathcal{E}_0 \triangleq \inf_{F\in\mathcal{F}} \mathbb{E}\|S_{t+1} - F(S_t, a_t)\|^2$ and $\mathcal{E}_1 \triangleq \inf_{F\in\mathcal{F}} \mathbb{E}\|S_{t+1} - F(S_t, a_t, S_{t-1}, a_{t-1})\|^2$. We say g is (ε, δ)-Markov if $\mathcal{E}_0 \le \varepsilon$ and $\mathcal{E}_0 - \mathcal{E}_1 \le \delta$ with $\delta \ll \varepsilon$. Empirically, we report ΔR^2 from adding (S_{t-1}, a_{t-1}).*

Hidden state and SGD dynamics. Let the *hidden* state be $x_t \triangleq (\theta_t,\, B_t,\, \tau)$, where θ_t are student parameters, B_t the remaining budget, and τ task parameters (e.g., margins, noise, imbalances). Given an action $a_t = (w_t, \ell_t, u_t)$—mixture over datasets, learning rate, and usage fraction—the student samples with *replacement* $m_t = \lfloor u_t B_t \rfloor$ examples from the mixture $\sum_i w_{t,i}\mathcal{D}_i^\tau$ and performs $K_t = \lceil m_t/b \rceil$ SGD steps (batch size b) with step size ℓ_t:

$$\theta_{t+1} \;=\; U(\theta_t; a_t, \tau, \xi_t), \qquad B_{t+1} = B_t - m_t, \tag{2}$$

where ξ_t denotes minibatch randomness. Sampling at t thus depends only on (x_t, a_t). *Remark:* If drawing *without* replacement from finite pools, augment x_t with per-dataset remaining counts so the process remains Markov; we otherwise assume with-replacement sampling in experiments.

Assumption 1 (With-replacement sampling). *Unless stated otherwise, mini-batches at time t are sampled* with replacement *from the mixture $\sum_i w_{t,i}\mathcal{D}_i^\tau$. This makes the data-generation mechanism conditionally independent of history given (x_t, a_t).*

Observable state (grouped, task/model/optimizer-agnostic). We decompose the observable state into three blocks and concatenate them for control:

$$S_t \;=\; \Big(\underbrace{\{z_{t,i}\}_{i=1}^{N}}_{\text{dataset-level set}} \;,\; \underbrace{g_t^{\text{model}}}_{\text{model complexity}} \;,\; \underbrace{g_t^{\text{progress}}}_{\text{training progress}} \Big).$$

(A) Dataset-level DeepSets embeddings. For each dataset i, compute a permutation-invariant embedding

$$z_{t,i} \;=\; \text{DeepSets}\Big(\{\text{probe stats from } \mathcal{D}_i\}\Big) \in \mathbb{R}^{d_z},$$

where the input set contains only task-agnostic statistics computed on a tiny, fixed probe slice each step:

- *Predictive performance & uncertainty:* macro-Acc/F1 (when labels available), mean NLL (length-normalized for sequences), confidence percentiles (p_{10}, p_{50}, p_{90}), ECE-5.
- *Loss/difficulty:* loss histogram or quantiles (8 bins).
- *Gradient-based difficulty:* mean/std of per-example gradient norms w.r.t. logits (log-scale).
- *Diversity sketch:* embedding variance or SimHash/Sketch counts from a small reservoir.
- *Light meta:* log dataset size and label entropy.

All quantities are standardized using robust EMAs (median/IQR) and are architecture- and label-permutation agnostic, following the spirit of Fan et al. [2018] (student feedback on samples) while remaining model-agnostic.

Definition 2 (Grouped state: minimal-viable instantiation)**.** *At each step t, the observable grouped state is $S_t = \big(\{z_{t,i}\}_{i=1}^{N},\, g_t^{\text{model}},\, g_t^{\text{progress}}\big)$, where, for each dataset i:*

$$z_{t,i} := \text{DeepSets}\Big(\{\textit{loss histogram (8 bins)},\ \textit{confidence percentiles } (p_{10}, p_{50}, p_{90}),$$
$$\textit{ECE-5},\ \textit{mean NLL (length-normalized)},\ \textit{grad-norm mean/std (log)},\ \textit{diversity sketch},\ \log|\mathcal{D}_i|,\ \textit{label}$$
$$g_t^{\text{model}} := \big[\log|\theta|,\ \log \textit{FLOPs/forward},\ \textit{overfit gap} = \textit{loss}_{\textit{train}} - \textit{loss}_{\textit{val}},\ \textit{LR},\ \textit{momentum or } (\beta_1, \beta_2),\ \textit{weight deca}$$
$$g_t^{\text{progress}} := \big[t/H,\ B_t/B_0,\ \text{EMA}(\textit{train loss}),\ \text{EMA}(\textit{val loss}),\ \Delta\textit{loss/step},\ \textit{gen gap},\ \textit{grad-norm mean/std},$$
$$\cos\angle(\nabla_t, \nabla_{t-1}),\ \textit{LR}/\textit{LR}_0\big].$$

All scalars are robustly normalized (median/IQR clipping). When labels on the probe are unavailable, macro-Acc/F1 terms are omitted and proper scores (NLL, calibration) are used.

(B) Model-complexity block. A small vector

$$g_t^{\text{model}} = \big[\log|\theta|,\ \log \text{FLOPs/forward},\ \text{overfit gap} = \text{loss}_{\text{train}} - \text{loss}_{\text{val}},\ \text{LR},\ \text{momentum or } (\beta_1, \beta_2),\ \text{weight decay}\big].$$

This captures capacity and optimizer settings without accessing architecture internals.

(C) Training-progress block. A small vector

$$g_t^{\text{progress}} = \big[t/H,\ B_t/B_0,\ \text{EMA(train loss)},\ \text{EMA(val loss)},\ \Delta\text{loss/step},\ \text{gen gap},\ \text{grad-norm mean/std},\ \cos\angle(\nabla_t,$$

These dynamics features approximate the transition-relevant information needed for Markov control.

Confidence calibration (ECE-5). We compute a 5-bin expected calibration error on the maximum predicted class probability $p_{\max}$. Let $\mathcal{B}_b$ be instances with $p_{\max} \in ((b-1)/5, b/5]$. Then $\text{ECE}_5 = \sum_{b=1}^{5} \frac{|\mathcal{B}_b|}{n} \big|\text{acc}(\mathcal{B}_b) - \text{conf}(\mathcal{B}_b)\big|$, where $\text{conf}(\mathcal{B}_b)$ is the average $p_{\max}$ and $\text{acc}(\mathcal{B}_b)$ the empirical accuracy. Here n is the number of evaluated instances (tokens for sequences; matched boxes for detection). For regression-style probes, ECE-5 is replaced by a proper calibration score (e.g., empirical coverage error under a Gaussian head).

Proposition 1 (Hidden-state Markovity)**.** *Under SGD with batches drawn according to w_t* with replacement *and fixed update rule U, the process on hidden states satisfies* $\mathbb{P}(x_{t+1}, r_t \mid x_t, a_t) = \mathbb{P}(x_{t+1}, r_t \mid x_t, a_t, x_{t-1}, a_{t-1}, \dots)$.

Proof sketch. Sampling at time t depends only on (x_t, a_t); the SGD update is a measurable function of $(\theta_t, a_t, \tau, \xi_t)$. Therefore (x_{t+1}, r_t) depends on history only through (x_t, a_t). If sampling without replacement, include remaining counts in x_t. □

Proposition 2 (Strict Markovity under information-preserving observation)**.** *If g is injective (i.e., $S_t = g(x_t)$ is an information-preserving encoding), then the observable process is Markov:* $\mathbb{P}(S_{t+1}, r_t \mid S_t, a_t) = \mathbb{P}(S_{t+1}, r_t \mid S_t, a_t, S_{t-1}, a_{t-1}, \dots)$.

Reward (macro-accuracy across datasets) and episode termination. Let $\text{Acc}_{i,t}$ be validation accuracy on dataset i; macro-accuracy $\text{MacroAcc}_t = \frac{1}{N}\sum_{i=1}^{N} \text{Acc}_{i,t}$. We use either terminal $r_t = \mathbf{1}\{t = H\}\text{MacroAcc}_H$ or incremental $r_t = \text{MacroAcc}_t - \text{MacroAcc}_{t-1}$. To avoid reward shaping issues from stochastic early termination, we either forbid early termination (fixed H) or use terminal-only rewards and report MacroAcc_H.

Budget as a CMDP. Budget consumption $c_t = m_t$ induces a constrained MDP $\max_\pi \mathbb{E}[\sum_{t=1}^H r_t]$ s.t. $\mathbb{E}[\sum_{t=1}^H c_t] \le B_0$. We optimize the Lagrangian $\mathcal{L} = \mathbb{E}\Big[\sum_{t=1}^H (r_t - \lambda\, c_t)\Big]$, with a per-episode dual update $\lambda \leftarrow \big[\lambda + \eta_\lambda\big(\sum_{t=1}^H c_t - B_0\big)\big]_+$.

3.2 Number-of-Datasets Invariance

State as a set. Let the per-dataset embeddings at time t be $\{z_{t,i}\}_{i=1}^N$. We require the encoder to be *permutation-invariant* and to admit variable cardinality N without architectural changes or scale drift. By the DeepSets theorem [Zaheer et al., 2017], any continuous permutation-invariant function f on sets can be written as

$$f(\{z_{t,i}\}) = \rho\left(\frac{1}{N}\sum_{i=1}^{N}\phi(z_{t,i})\right), \tag{3}$$

for suitable ϕ and ρ (universal approximators in practice). We implement the policy/value encoders as shared per-dataset blocks ϕ followed by *mean* pooling and a readout ρ, then concatenate g_t^{model} and g_t^{progress}, making them invariant to dataset order and stable in N.

Action as per-dataset logits. For mixture control, the policy emits a scalar logit $\alpha_{t,i}$ for each dataset via a *shared* head applied to every $z_{t,i}$: $\alpha_{t,i} = h_{\text{mix}}(\phi(z_{t,i}))$, $\quad w_{t,i} = \frac{\exp(\alpha_{t,i})\,\mathbf{1}\{n_i>0\}}{\sum_{j=1}^N \exp(\alpha_{t,j})\,\mathbf{1}\{n_j>0\}}$, where n_i is the available size for dataset i if sampling without replacement. Under with-replacement sampling, we drop the mask and set $w_t = \text{softmax}(\{\alpha_{t,i}\})$. Shared heads and normalization over observed elements yield permutation equivariance and well-defined actions for any N.

Implication. The same learned policy applies to unseen numbers of datasets or new orderings without retraining; only the set S_t changes in size or order, which the DeepSets encoder and per-element policy heads support natively.

4 Method

4.1 Policy and Training

We use an invariant encoder f_ψ with a shared per-dataset block ϕ_ψ and *mean*-pooled DeepSets over $\{z_{t,i}\}$ to produce a representation h_t, then concatenate the grouped globals g_t^{model} and g_t^{progress}. Mean pooling (vs. sum) avoids scale drift as N varies. Policy heads output (i) per-dataset mixture logits $\{\alpha_{t,i}\}$, normalized to w_t, (ii) learning rate ℓ_t (bounded via squashing to $[\ell_{\min}, \ell_{\max}]$), and (iii) usage $u_t \in (0,1)$ (sigmoid). A value head $V_\psi(S_t)$ shares the invariant encoder and globals.

Action regularizers and barriers. To prevent collapse and boundary hugging, we add entropy and barrier regularizers to the PPO loss: $\mathcal{L}_{\text{ent}} = \beta_{\text{mix}}\, H(w_t) + \beta_u\, H(u_t)$, $\quad \mathcal{L}_{\text{bar}} = \kappa\big[-\log(\ell_t - \ell_{\min}) - \log(\ell_{\max} - \ell_t)\big] + \kappa'\big[-\log u_t - \log(1-u_t)\big]$, with standard clipping of pre-squash outputs for numerical stability.

Student update semantics (removing confounds). We fix batch size b. Usage chooses a sample count $m_t = \lfloor u_t B_t \rfloor$, inducing $K_t = \lceil m_t/b \rceil$ optimizer steps with step size ℓ_t. Mini-batches are drawn *with replacement* from the mixture with proportions w_t. This separates the effects of step size and data volume; any residual coupling is tempered by $\mathcal{L}_{\text{bar}}$ and the CMDP penalty.

Algorithm 1 PPO teacher on a Markov, set-valued curriculum MDP (updated)

Initialize policy/value params ϕ, ψ
for iteration $= 1, 2, \ldots$ **do**
  for each parallel episode **do**
    Sample task $\tau \sim \mathcal{T}$; reset student; $B_1 \leftarrow B_0$
    for $t = 1$ to H **do**
      ▷ build grouped state on fixed probes
      For each dataset i, compute probe statistics and $z_{t,i}$; set $h_t = \rho\big(\frac{1}{N}\sum_i \phi(z_{t,i})\big) \,\|\, g_t^{\text{model}} \,\|\, g_t^{\text{progress}}$
      Emit $\{\alpha_{t,i}\}$, ℓ_t, u_t; set $w_t = \text{softmax}(\{\alpha_{t,i}\})$
      Apply K_t SGD steps using batches drawn *with replacement* from mixture w_t with step size ℓ_t
      $m_t = \lfloor u_t B_t \rfloor$; $B_{t+1} = B_t - m_t$
      (Optional, finite pools) re-mask any dataset with $n_i = 0$
      Observe (S_{t+1}, r_t)
    end for
  end for
  Compute advantages; update ϕ, ψ with PPO and dual variable λ (per-episode update)
end for

5 Experiments

5.1 Cross-Task Meta-Training and Evaluation

Task families. We instantiate three families for meta-training: (i) classification (synthetic mixtures + small real subsets), (ii) sequence labeling (synthetic span tasks + small NER subsets), and (iii) detection (synthetic shapes with boxes + small real subsets). We construct episodes that randomize dataset difficulty, size, and noise. The teacher never observes task IDs; it only receives the grouped state (dataset embeddings + model/training blocks).

Leave-one-family-out (LOFO). We train the teacher on two families and evaluate zero-shot on the held-out family (e.g., train on classification+NER, test on detection), reporting macro metrics appropriate to each family (accuracy/F1, F1 for NER, mAP/AR for detection), along with *worst-dataset* performance and sample-efficiency AUC.

Large-model transfer. We stress-test transfer by deploying the teacher zero-shot to a large ViT-based detector trained over 10 detection datasets. The teacher controls dataset mixture, learning rate, and usage; the state is the same grouped observation computed on a fixed probe slice. We report mAP@[.50:.95], $\text{AP}_{\text{S/M/L}}$, and worst-dataset AP.

Diagnostics. We measure (i) *approximate Markovity*: one-step R^2 of S_{t+1} from (S_t, a_t) and the ΔR^2 from adding history; (ii) *N-invariance*: training at $N \in \{3, 5\}$ and evaluating at $N \in \{2, 4, 7, 10\}$ under random permutations; and (iii) entropy/barrier usage to verify stable control.

Baselines. Static Uniform, Easy→Hard, Myopic Greedy (one-step gain surrogate), bandits (LinUCB/Thompson over datasets), and schedule controllers (PBT/BOHB).

5.2 Validating Markovity

We fit a one-step predictor $\hat{F}_\eta$ trained to minimize $\|S_{t+1} - \hat{F}_\eta(S_t, a_t)\|^2$ on logged rollouts; high R^2 supports approximate Markovity of the grouped state (Def. 1). We split train/test *by episode* to avoid leakage and report R^2 per feature, alongside a trivial "no-change" baseline. We also test whether augmenting inputs with history (S_{t-1}, a_{t-1}) improves R^2; a small ΔR^2 indicates limited additional information in history.

Table 1: One-step prediction on the grouped observation (placeholders). Higher is better.

Env	$R^2(S_t, a_t \to S_{t+1})$	ΔR^2 w/ history	GRU policy gain
Gaussian	0.00	+0.00	+0.00
Linear	0.00	+0.00	+0.00
Shapes	0.00	+0.00	+0.00

5.3 Testing Number-of-Datasets Invariance

We train teachers with $N \in \{3, 5\}$ and evaluate zero-shot on $N \in \{2, 4, 7, 10\}$, as well as under random dataset permutations at each episode start. We report MacroAcc and permutation robustness σ_π(MacroAcc), the standard deviation across permutations of the same episode.

Table 2: N-scaling and permutation robustness across tasks (placeholders).

Train N	Test N	MacroAcc ↑	Worst Acc ↑	Permutation σ
3	2	00.0	00.0	0.00
3	7	00.0	00.0	0.00
5	10	00.0	00.0	0.00

5.4 Main Results (Placeholders)

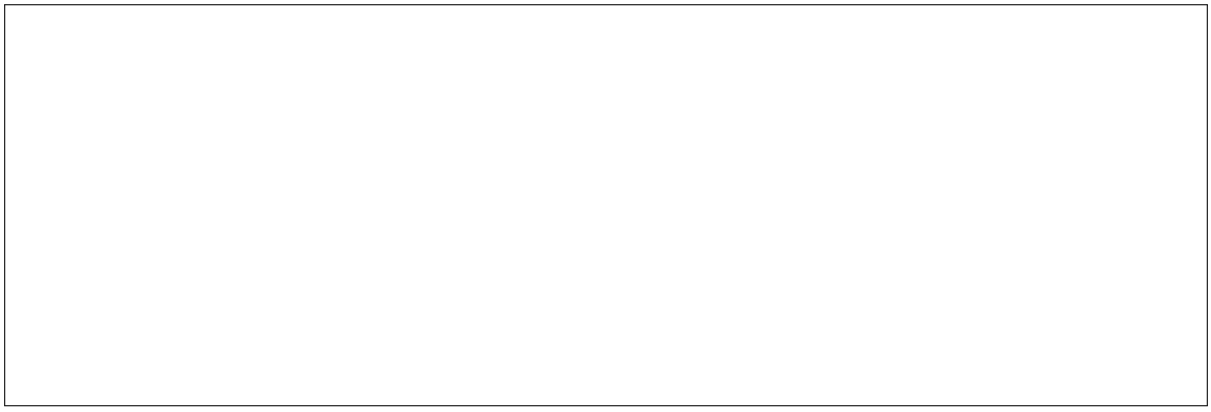

Figure 1: Learning curves (MacroAcc vs. steps) in cross-task and LOFO evaluations.

Table 3: Final MacroAcc (%) and Worst-dataset Acc (%) on cross-task and LOFO tests.

Method	MacroAcc ↑	Worst-dataset ↑
Static Uniform	00.0	00.0
Easy→Hard	00.0	00.0
Myopic Greedy	00.0	00.0
Bandit (LinUCB)	00.0	00.0
PBT/BOHB	00.0	00.0
PPO Teacher (ours)	**00.0**	**00.0**

5.5 Out-of-Distribution Transfer (Placeholders)

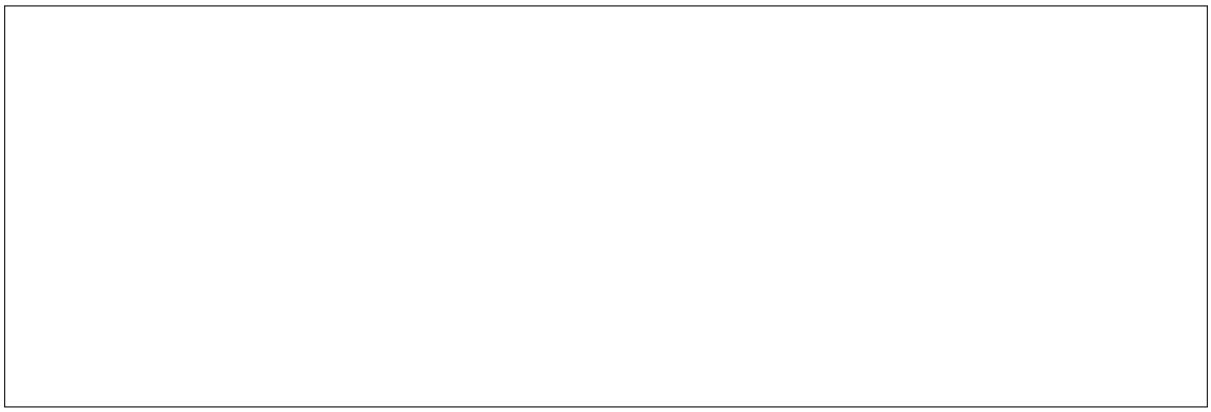

Figure 2: Zero-shot large-model transfer (e.g., ViT detection) and OOD regions (higher noise/imbalance).

Table 4: Zero-shot and few-shot MacroAcc on OOD grid and held-out task families.

Method	Zero-shot ↑	Few-shot (10 eps) ↑
Static Uniform	00.0	00.0
Bandit (Thompson)	00.0	00.0
PPO Teacher (ours)	**00.0**	**00.0**

5.6 Ablations (Placeholders)

- **State sufficiency:** remove trends; ablate dataset-embedding features or model/progress blocks (e.g., drop ECE or gradient stats); replace with alternative scalars; switch MLP→GRU.
- **Action parameterization:** continuous simplex vs. sparse top-k; effect of discretization for ℓ, u.
- **Encoder invariance:** shared per-dataset block + pooling vs. flat concatenation.
- **Horizon/budget:** vary H and B_0; measure stability and sample efficiency.

6 Discussion

We unify three desiderata for RL-based curriculum teaching in the many-to-one, group-sensitive setting: *Markovity*, *number-of-datasets invariance*, and *task/architecture invariance*. By formalizing hidden-state Markovity under SGD and using a *grouped* observation (dataset-level embeddings + model-complexity + training-progress), we obtain an observation that is strictly Markov when information-preserving and empirically *approximately* Markov otherwise (Def. 1).

DeepSets provides the route to invariance: treating datasets as a set and sharing per-dataset encoders with *mean* pooling guarantees order and cardinality robustness without scale drift as N varies. Coupling this with shared per-dataset action heads and normalization yields mixture policies that seamlessly accommodate unseen N and permutations. Crucially, controlling mixture, usage, and step size enables the teacher to *protect minority datasets* by adaptively reallocating budget—an operational complement to distributionally robust objectives.

Implications. A compact, task-agnostic grouped observation suffices to control learning dynamics across task families and architectures while preserving Markovity and N-invariance. This supports training a single teacher on inexpensive episodes and transferring it to large-scale mixtures where small, high-stakes datasets must not be overshadowed.

Limitations. The grouped state is intentionally coarse, foregoing task-specific diagnostics (e.g., error typing in detection) that may accelerate learning in specialized domains. Its efficacy depends on stable computation of NLL and confidence summaries (sequence normalization matters). Extreme regimes (e.g., severe class imbalance with heavy augmentation) may benefit from adding task-agnostic action heads (e.g., augmentation intensity tiers), which we leave to future work.

7 Conclusion

We presented a Markov, number-of-datasets invariant, and task/architecture-invariant formulation for RL-based curriculum teaching, aimed at the practical setting where small, critical datasets are trained alongside large auxiliaries. Our grouped state (dataset embeddings + model-complexity + training-progress) and set-based encoders make the problem well-posed for RL and scalable across any N. By jointly allocating mixture, learning rate, and usage, the teacher steers training toward balanced outcomes—echoing the fairness intuition that small groups should not be ignored—without changing the base loss.

Reproducibility. We will release code, configs, and scripts to reproduce all experiments (including CI-friendly runs), together with fixed probes and seeds.

References

Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *Proceedings of the 26th International Conference on Machine Learning (ICML)*, 2009.

M. P. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2010.

T. Schaul, J. Quan, I. Antonoglou, and D. Silver. Prioritized experience replay. In *International Conference on Learning Representations (ICLR)*, 2016.

M. Jaderberg et al. Population based training of neural networks. *arXiv:1711.09846*, 2017.

J. Snoek, H. Larochelle, and R. P. Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2012.

S. Falkner, A. Klein, and F. Hutter. BOHB: Robust and efficient hyperparameter optimization at scale. In *International Conference on Machine Learning (ICML)*, 2018.

C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation. In *International Conference on Machine Learning (ICML)*, 2017.

Y. Duan et al. RL^2: Fast reinforcement learning via slow reinforcement learning. *arXiv:1611.02779*, 2016.

M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. Salakhutdinov, and A. Smola. Deep Sets. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

J. Lee, Y. Lee, J. Kim, A. R. Kosiorek, S. Choi, and Y. W. Teh. Set Transformer: A Framework for Attention-based Permutation-Invariant Neural Networks. In *International Conference on Machine Learning (ICML)*, 2019.

Fan et al. Learning to Teach. *arXiv:1805.03643*, 2018.

Wu et al. Learning to Teach with Dynamic Loss Functions. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

A. Kendall, Y. Gal, and R. Cipolla. Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich. GradNorm: Gradient Normalization for Adaptive Loss Balancing in Deep Multitask Networks. *arXiv:1711.02257*, 2018.

S. Narvekar, J. Sinapov, M. Leonetti, and P. Stone. Curriculum Learning for Reinforcement Learning: A Survey. *arXiv:2003.04960*, 2020.

T. B. Hashimoto, M. Srivastava, H. Namkoong, and P. Liang. Fairness Without Demographics in Repeated Loss Minimization. In *International Conference on Machine Learning (ICML)*, 2018.

S. Sagawa, P. W. Koh, T. B. Hashimoto, and P. Liang. Distributionally Robust Neural Networks for Group Shifts: On the Importance of Regularization for Worst-Group Performance. In *International Conference on Learning Representations (ICLR)*, 2020.