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 MAESTRO (*Markovian, Architecture-Agnostic, Equitable Scheduling for Task-Robust Optimization*), 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 number of datasets 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 numbers and permutations of datasets, 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, MAESTRO achieves strong sample efficiency and improves balanced metrics (macro metrics) 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 any number of datasets, so that small, specialized datasets are not eclipsed

by large, easy ones. Here, "macro" denotes the equal-weight average of per-dataset accuracies, irrespective of dataset size. We refer to our teacher as MAESTRO (*Markovian, Architecture-Agnostic, Equitable Scheduling for Task-Robust Optimization*).

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 number of datasets 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 number of datasets. Additional heads control learning rate and usage, allowing the teacher to modulate *both* exposure and step size. We adopt a CMDP view over budget. We denote this unified design by MAESTRO.

Contributions. (1) We introduce MAESTRO, 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 number of datasets 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 metrics) over uniform sampling, myopic/bandit schedulers, and heuristic curricula. Conceptually, MAESTRO 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 number of datasets; 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 study a teacher–student interaction in which a policy π_ϕ controls the training of a student f_θ on N datasets $\{\mathcal{D}_i\}_{i=1}^N$ over a horizon H with an initial example budget B_0. The goal is to maximize a balanced end metric—macro-accuracy across datasets—so that small, high-stakes datasets are not eclipsed by large auxiliaries. Formally, with per-dataset validation accuracies $\mathrm{Acc}_{i,t}$, we write $\mathrm{MacroAcc}_t = \frac{1}{N}\sum_{i=1}^N \mathrm{Acc}_{i,t}$ and optimize either a terminal reward $r_t = \mathbf{1}\{t = H\}\mathrm{MacroAcc}_H$ or the incremental difference $r_t = \mathrm{MacroAcc}_{t+1} - \mathrm{MacroAcc}_t$ under a budget constraint.

3.1 A Markov Teacher–Student MDP

We cast curriculum control as an MDP with observable state S_t, action a_t, and reward r_t. The process is Markov if

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

The *hidden* state of training is $x_t \triangleq (\theta_t, B_t, \tau)$, where θ_t are the student parameters, B_t the remaining budget, and τ episode-specific task parameters. At each decision point the teacher selects

$$a_t = (w_t, \eta_t, u_t),$$

namely per-dataset mixture weights $w_t \in \Delta^{N-1}$, a learning rate η_t, and a usage fraction $u_t \in (0, 1)$. The student then draws $m_t = \lfloor u_t B_t \rfloor$ examples with *replacement* 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 η_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. With-replacement sampling ensures that (x_{t+1}, r_t) depends on history only through (x_t, a_t); the hidden process is therefore Markov. If sampling without replacement is desired, one augments x_t with the remaining per-dataset counts to retain Markovity.

Budget as a CMDP. The consumable cost $c_t = m_t$ induces a constrained MDP:

$$\begin{aligned} \max_{\pi} \quad & \mathbb{E}\left[\sum_{t=1}^{H} r_t\right] \\ \text{s.t.} \quad & \mathbb{E}\left[\sum_{t=1}^{H} c_t\right] \le B_0 \,. \end{aligned} \tag{3}$$

We optimize this via the standard Lagrangian

$$\mathcal{L}(\lambda) \;=\; \mathbb{E}\left[\sum_{t=1}^{H} \big(r_t - \lambda\, c_t\big)\right], \tag{4}$$

with per-episode dual updates $\lambda \leftarrow \big[\lambda + \eta_\lambda \big(\sum_{t=1}^{H} c_t - B_0\big)\big]_+$.

3.2 A Compact, Grouped Observation

To control stochastic gradient dynamics effectively, the teacher must observe just enough about (i) what the *data* currently elicits from the student, (ii) how the *update rule* is configured, and (iii) where the run sits in its *training trajectory*. We therefore expose a grouped, task/architecture-agnostic state

$$S_t \;=\; \big(g_t^{\text{data}},\; g_t^{\text{model}},\; g_t^{\text{progress}}\big) \;\in\; \mathbb{R}^{d_{\text{data}}} \times \mathbb{R}^{d_m} \times \mathbb{R}^{d_p}. \tag{5}$$

Here d_{data} is an architectural choice determined by the DeepSets readout, while d_m and d_p are fixed by our engineered feature sets; the precise probes, invariance mechanism, and action parameterization are detailed in the sections that follow.

Dataset block (set-valued summary). For each dataset i, a fixed probe produces a compact, task-agnostic descriptor $z_{t,i} \in \mathbb{R}^8$ that tracks (i) *difficulty and dispersion* of the loss through a level-and-spread pair, (ii) the model's *predictive uncertainty and calibration* on the probe, (iii) the *scale and agreement* of gradients between this dataset and the recent global direction, (iv) *diversity* of example representations in an embedding space, and (v) dataset *size*. Concretely, we instantiate these signals with the mean and interquartile range of NLL, the mean predictive entropy together with a five-bin expected calibration error (ECE-5), $\mathrm{ECE}_5 = \sum_{b=1}^{5} \frac{|\mathcal{B}_b|}{n} |\mathrm{acc}(\mathcal{B}_b) - \mathrm{conf}(\mathcal{B}_b)|$, the log-EMA gradient norm and its cosine similarity to a running global gradient, a log effective rank $\log r_{\mathrm{eff}}(\Sigma_i)$ of the probe embedding covariance Σ_i (with $r_{\mathrm{eff}}(\Sigma) = \frac{(\sum_j \lambda_j)^2}{\sum_j \lambda_j^2}$ for eigenvalues $\{\lambda_j\}$), and $\log |\mathcal{D}_i|$. Here n denotes the size of the probe set used to compute calibration bins. These quantities are label-lean, architecture-agnostic, and robustly normalized (median/IQR EMAs), and together they reveal when a dataset is hard, miscalibrated, conflicting, or redundant—precisely the factors a curriculum should adapt to. To obtain a permutation- and cardinality-invariant global summary we apply a DeepSets encoder with mean pooling:

$$g_t^{\mathrm{data}} \;=\; \rho\!\left(\frac{1}{N}\sum_{i=1}^{N} \phi(z_{t,i})\right), \tag{6}$$

where ϕ and ρ are shared neural networks. The per-dataset descriptors themselves are not part of the fixed-length state fed to the value function; they are consumed inside the policy to emit per-dataset actions while preserving invariance. Let $g_t \triangleq \frac{1}{K_t}\sum_{k=1}^{K_t} \nabla_\theta \mathcal{L}(\theta_t; \mathcal{B}_{t,k})$ be the average per-step gradient over the K_t mini-batches drawn at decision time t. Let $\bar{g}_t = \mathrm{EMA}_\beta(g_t)$ be its exponential moving average. For each dataset i, define $g_{t,i} \triangleq \nabla_\theta \mathcal{L}(\theta_t; \mathcal{P}_{t,i})$, the gradient on a fixed small probe mini-batch $\mathcal{P}_{t,i} \subset \mathcal{D}_i$. Collecting these descriptors yields the dataset-level summary vector used throughout the policy and value networks:

$$\begin{aligned} g_t^{\mathrm{data}} &= \rho\!\left(\frac{1}{N}\sum_{i=1}^{N} \phi\big(z_{t,i}\big)\right), \\ z_{t,i} &= \Big[\, \overline{\mathrm{NLL}}_{t,i},\ \mathrm{IQR}(\mathrm{NLL})_{t,i},\ \overline{H}_{t,i},\ \mathrm{ECE}_{5,t,i}, \\ &\qquad \log \|g_{t,i}\|_2,\ \cos\big(g_{t,i}, \bar{g}_t\big),\ \log r_{\mathrm{eff}}(\Sigma_{t,i}),\ \log |\mathcal{D}_i| \,\Big]^\top \in \mathbb{R}^8. \end{aligned} \tag{7}$$

Model-complexity block. We summarize the model with six scalars $g_t^{\mathrm{model}} \in \mathbb{R}^6$ chosen to be optimizer-independent, architecture-agnostic, and inexpensive to compute while still capturing capacity and geometry that influence learning dynamics. Specifically, $\log|\theta_t|$ (total parameter count) serves as a robust proxy for representational capacity across architectures; $\log \mathrm{FLOPs}_{\mathrm{fwd+bwd}}$ per sample reflects the computational scale of one SGD step and correlates with effective curvature/noise regimes; D_t (the longest parameterized path length) approximates effective depth, which affects gradient propagation and the stability of updates; $\mathrm{med}_\ell \log n_\ell$ (a width proxy with $n_\ell = \#W_\ell$, i.e., the number of scalar weights in layer ℓ) captures typical layer scale without drifting with the number of layers; sparsity $s_t = \frac{1}{|\theta_t|}\sum \mathbf{1}\{|w| < \tau_t\}$ (with a robust threshold τ_t set from the interquartile range of weight magnitudes) reflects implicit regularization/compression; and skip density ρ_{skip} (residual/skip edges per parameterized layer) quantifies shortcut connectivity that improves gradient flow and reduces effective depth. We use logarithms and medians to stabilize these descriptors under heavy-tailed weight distributions and to ensure comparability across CNNs, Transformers, and MLPs. In parallel, we capture the model's evolving characteristics through a

lightweight architectural summary:

$$g_t^{\text{model}} = \Big[\, \log|\theta_t|,\ \log \text{FLOPs}_{\text{fwd+bwd}},\ D_t,\ \text{med}_\ell \log n_\ell,\ s_t,\ \rho_{\text{skip},t} \Big]^\top \in \mathbb{R}^6 \tag{8}$$

Training-progress block. Finally, the progress block summarizes both the state of optimization and where the run sits in the overall training trajectory. It contains eleven scalars $g_t^{\text{progress}} \in \mathbb{R}^{11}$ capturing not only normalized time and budget—t/H and B_t/B_0—but also the instantaneous optimizer state that directly affects learning dynamics: the current learning rate, momentum-like coefficient (μ for SGD or β_1 for Adam), and the log weight decay. The remaining features describe empirical progress: the validation loss level and generalization gap (train minus validation EMA), the short-horizon slope of the validation loss to detect regime shifts, a scale-free update-size proxy $\text{UPR}_t = \|\Delta\theta_t\|_2/\|\theta_t\|_2$, the cosine similarity between successive gradients, and a log-EMA of the global gradient norm. Together these quantities expose the current stage of training, the stability of updates, and how optimization hyperparameters interact with observed progress—precisely the information a teacher needs to schedule exposure and adjust step size adaptively. Finally, to reflect the optimizer's trajectory and the learner's temporal position, we define a set of progress-oriented features:

$$\begin{aligned} g_t^{\text{progress}} = \Big[& t/H,\ B_t/B_0,\ \eta_t,\ \mu_t,\ \log \lambda_t^{\text{wd}},\ \mathcal{L}_t^{\text{val}}, \\ & \big(\mathcal{L}_t^{\text{train}} - \mathcal{L}_t^{\text{val}}\big),\ \text{slope}_t^{\text{val}},\ \text{UPR}_t,\ \cos\big(g_t, g_{t-1}\big),\ \log \widetilde{G}_t \Big]^\top \in \mathbb{R}^{11}, \\ \widetilde{G}_t = \text{EMA}_\alpha\big(\, & \|g_t\|_2 \,\big). \end{aligned} \tag{9}$$

Here, $\Delta\theta_t \triangleq \theta_t - \theta_{t-1}$ denotes the previous segment update used in UPR_t (with $\text{UPR}_1 = 0$). This ensures UPR_t is available when forming S_t prior to action selection.

By convention we treat $(\eta_t, \mu_t, \lambda_t^{\text{wd}})$ inside g_t^{progress} as the *currently active* optimizer hyperparameters before the teacher issues a_t; the action may overwrite η_t (and thus the effective step-size behaviour) for the subsequent update segment.

Action parametrization and invariance. Although S_t has constant length, the policy must output a valid mixture over an arbitrary number of datasets. We realize this by applying a shared head to each encoded element, producing per-dataset logits

$$\alpha_{t,i} = h_{\text{mix}}\big(\phi(z_{t,i}),\, g_t^{\text{data}},\, g_t^{\text{model}},\, g_t^{\text{progress}}\big), \qquad w_{t,i} = \frac{\exp(\alpha_{t,i})}{\sum_{j=1}^N \exp(\alpha_{t,j})}, \tag{10}$$

and separate heads for η_t and u_t. The DeepSets encoder in (6) and the shared per-element head in (10) make the policy permutation-equivariant and stable as N varies.

Final state space. In our instantiation, the per-dataset descriptor is $z_{t,i} \in \mathbb{R}^8$, the model block is $g_t^{\text{model}} \in \mathbb{R}^6$, and the progress block is $g_t^{\text{progress}} \in \mathbb{R}^{11}$. Consequently

$$S_t \;=\; \big(\rho(\tfrac{1}{N}\sum_i \phi(z_{t,i})),\, g_t^{\text{model}},\, g_t^{\text{progress}}\big) \;\in\; \mathbb{R}^{d_{\text{data}}} \times \mathbb{R}^6 \times \mathbb{R}^{11},$$

with d_{data} fixed by the readout ρ and independent of N; in this instantiation $d_m = 6$ and $d_p = 11$. This grouped, compact observation is strictly Markov when information-preserving and, empirically, approximately Markov otherwise; it is expressly designed to capture the levers most predictive of one-step dynamics under SGD while remaining invariant to the number and ordering of datasets.

Definition 1 (MAESTRO teacher)**.** *Let* $\{\mathcal{D}_i\}_{i=1}^N$ *be datasets,* B_0 *an initial example budget, and* H *a horizon.* MAESTRO *is the policy* π_ϕ *that, at decision time* t*, observes the grouped state* $S_t = \left(g_t^{\text{data}}, g_t^{\text{model}}, g_t^{\text{progress}}\right)$ *defined in §3.2, and emits actions* $a_t = (w_t, \eta_t, u_t)$ *where* $w_t \in \Delta^{N-1}$ *are per-dataset mixture weights produced by a permutation-equivariant head applied to* $\{\phi(z_{t,i})\}_i$*,* η_t *is a bounded learning rate, and* $u_t \in (0, 1)$ *is a usage fraction. The student updates by sampling* $m_t = \lfloor u_t B_t \rfloor$ *examples* with replacement *from* $\sum_i w_{t,i}\mathcal{D}_i$ *and applying* $K_t = \lceil m_t/b \rceil$ *SGD steps (batch size* b*) at step size* η_t*. The episode is optimized as a constrained MDP with cost* $c_t = m_t$ *and reward defined from* MacroAcc *as in §3.1.*

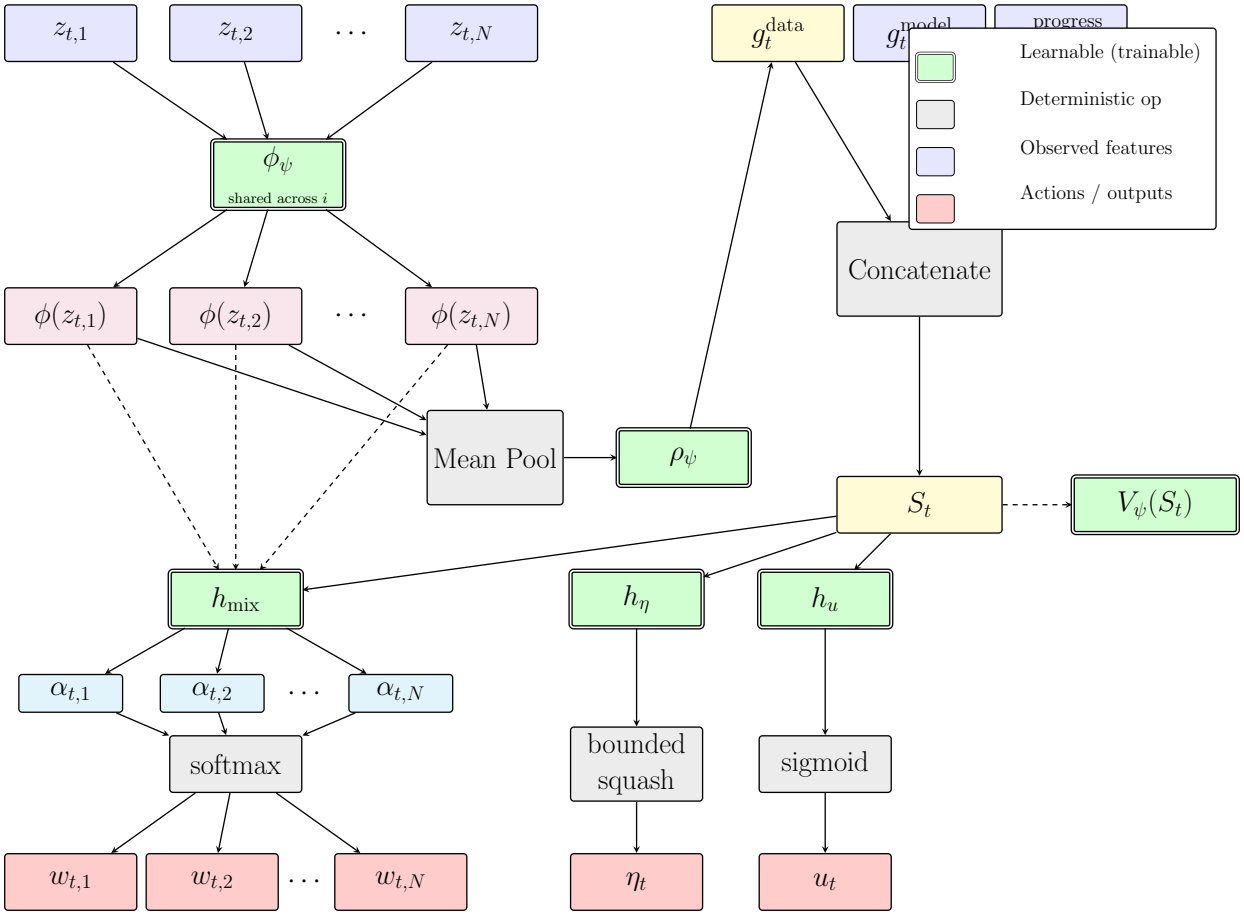

Figure 1: **Architecture and learnable components of the MAESTRO policy network.** Per-dataset descriptors $\{z_{t,i}\}_{i=1}^N$ are encoded through a shared per-dataset encoder ϕ_ψ and mean-pooled via ρ to form the data summary g_t^{data}. Combined with g_t^{model} and g_t^{progress}, this yields the state S_t. The shared mixture head h_{mix} is applied to each dataset embedding $\phi(z_{t,i})$ along with the global state (dashed arrows) to produce per-dataset logits $\{\alpha_{t,i}\}_{i=1}^N$, which are normalized via softmax to obtain mixture weights $w_t \in \Delta^{N-1}$. Separate heads emit the learning rate η_t (bounded via squashing to $[\eta_{\min}, \eta_{\max}]$) and usage u_t (sigmoid to $(0, 1)$), ensuring scalability to any number of datasets. Double-bordered green blocks denote *learnable* components (trained via PPO): ϕ, ρ, h_{mix}, h_η, h_u, and V_ψ. Gray blocks indicate deterministic operations. Inputs $z_{t,i}$, g_t^{model}, and g_t^{progress} are observed, while outputs w_t, η_t, and u_t are the action components. The mixture head is shared across datasets, with dashed arrows showing its dependence on per-dataset embeddings.

Definition 2 (Approximate Markovity of the grouped state)**.** *Fix a predictor class $\mathcal{F}$ and thresholds $(\delta, \varepsilon) \in (0,1) \times [0,1)$. We say the grouped observation S_t is* (δ, ε)-approximately Markov *for the training dynamics if there exists $\hat{F} \in \mathcal{F}$ such that the one-step prediction achieves $R^2\Big(S_{t+1};\, \hat{F}(S_t, a_t)\Big) \geq \delta$ and the marginal improvement from adding a short history is small: $\Delta R^2 \equiv R^2\Big(S_{t+1};\, \hat{F}(S_t, a_t, S_{t-1}, a_{t-1})\Big) - R^2\Big(S_{t+1};\, \hat{F}(S_t, a_t)\Big) \leq \varepsilon$. In experiments we report R^2 and ΔR^2 for a fixed $\mathcal{F}$ (MLP) as diagnostics of approximate Markovity.*

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 η_t (bounded via squashing to $[\eta_{\min}, \eta_{\max}]$), and (iii) usage $u_t \in (0,1)$ (sigmoid). A value head $V_\psi(S_t)$ shares the invariant encoder and globals. This architecture instantiates MAESTRO as defined in Def. 1.

Action regularizers and barriers. To prevent collapse and boundary hugging, we add entropy and barrier regularizers to the PPO loss: $\mathcal{L}_{\text{ent}} = \beta_{\text{mix}} H(w_t) + \beta_u H(u_t)$, $\mathcal{L}_{\text{bar}} = \kappa\big[-\log(\eta_t - \eta_{\min}) - \log(\eta_{\max} - \eta_t)\big] + \kappa'\big[-\log u_t - \log(1-u_t)\big]$, 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 η_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 $\{\alpha_{t,i}\}$, η_t, u_t; set $w_t = \text{softmax}(\{\alpha_{t,i}\})$
      $m_t = \lfloor u_t B_t \rfloor$; $K_t = \lceil m_t / b \rceil$
      Draw mini-batches *with replacement* from mixture w_t; apply K_t SGD steps at step size η_t
      $B_{t+1} = B_t - m_t$
      Compute $r_t \leftarrow \text{MacroAcc}_{t+1} - \text{MacroAcc}_t$
      Observe S_{t+1}
    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), 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] and $\text{AP}_{\text{S/M/L}}$.

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. 2). 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 ↑	Permutation σ
3	2	00.0	0.00
3	7	00.0	0.00
5	10	00.0	0.00

5.4 Main Results (Placeholders)

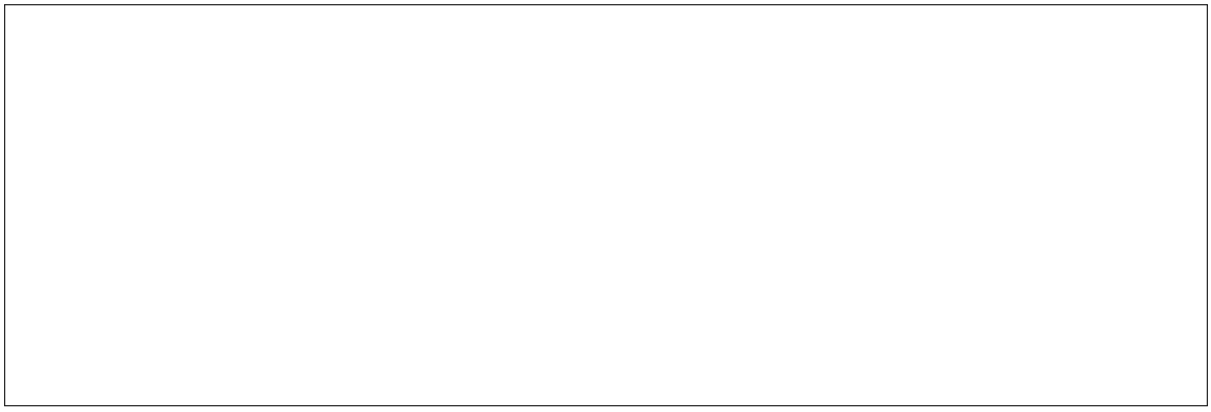

Figure 2: Learning curves (MacroAcc vs. steps) in cross-task and LOFO evaluations.

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

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

5.5 Out-of-Distribution Transfer (Placeholders)

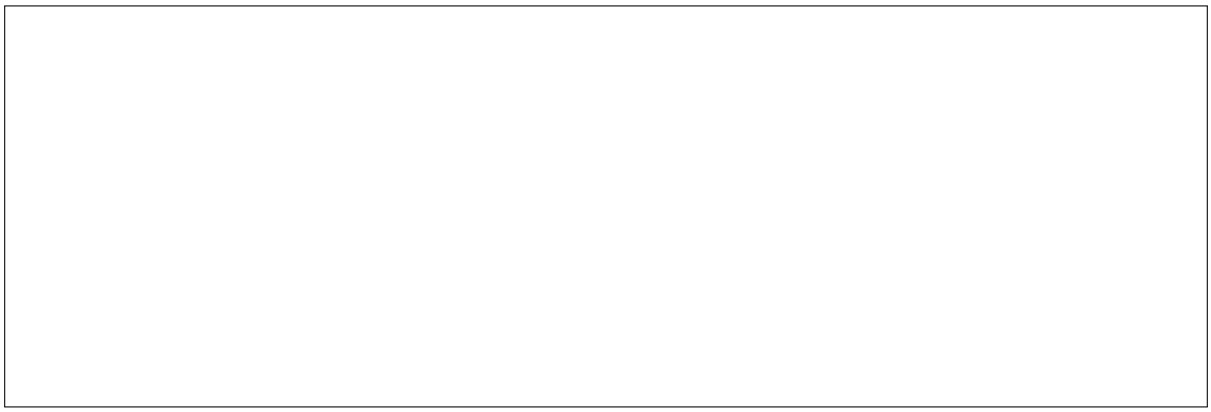

Figure 3: Zero-shot large-model transfer (e.g., ViT detection) and OOD regions (higher noise/imbalance).

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 η, 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. 2).

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 MAESTRO 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 number of datasets. 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.

Name. We call this teacher MAESTRO: *Markovian, Architecture-Agnostic, Equitable Scheduling for Task-Robust Optimization*. **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.