

Convex Dataset Selection for Post-Training via Portfolio Optimization

Han Zhao

10/18/2026

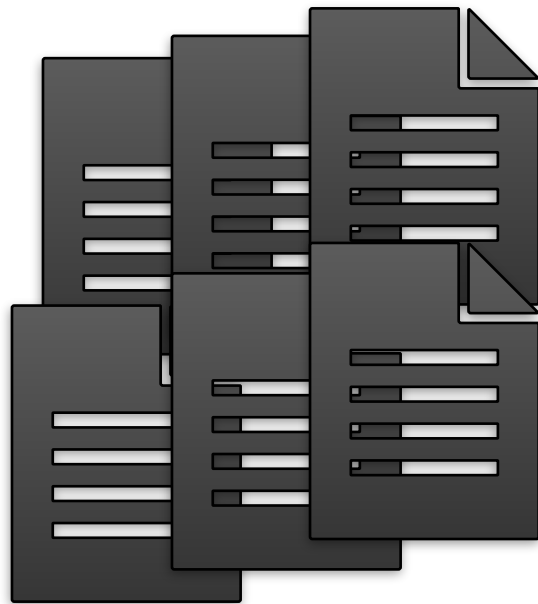
The 2nd International Conference on AI Scientists (ICAIS 2026)
Zhongguancun Academy, Beijing, China

Assistant Professor, Amazon Scholar
hanzhao@illinois.edu
Siebel School of Computing and Data Science
University of Illinois Urbana-Champaign

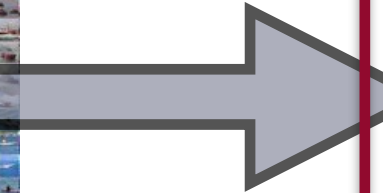


Typical Machine Learning Pipeline

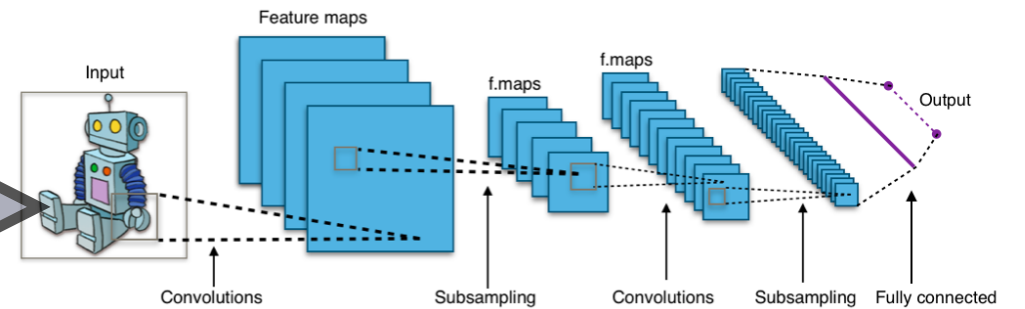
Data



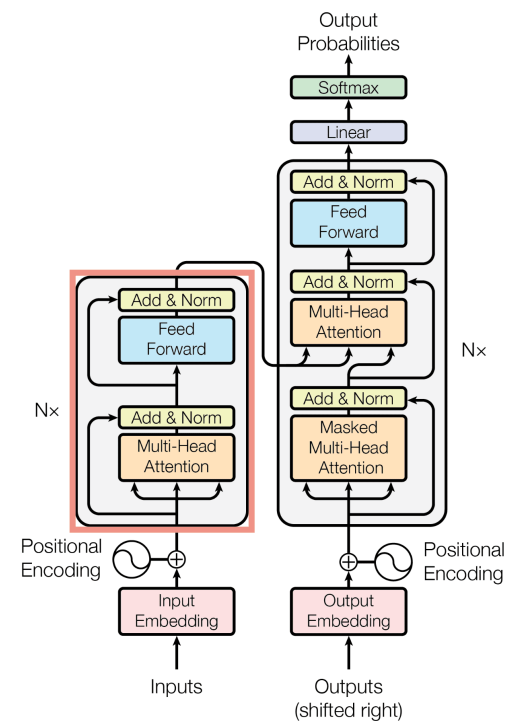
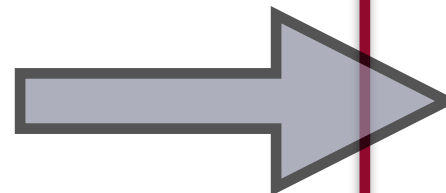
2013



Models



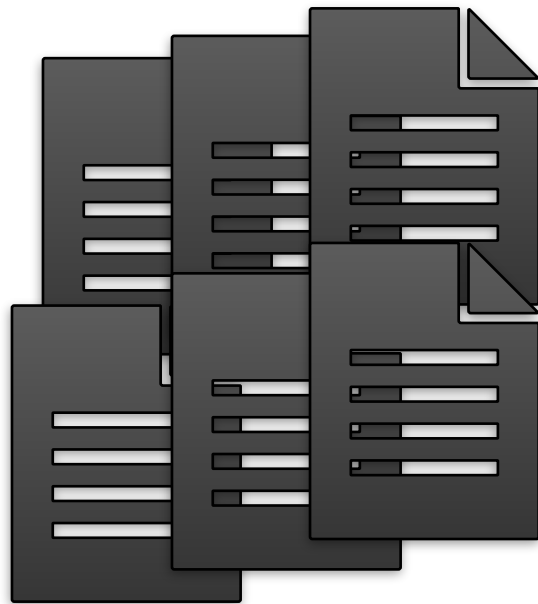
2022



Typically: Our focus is on the trained model

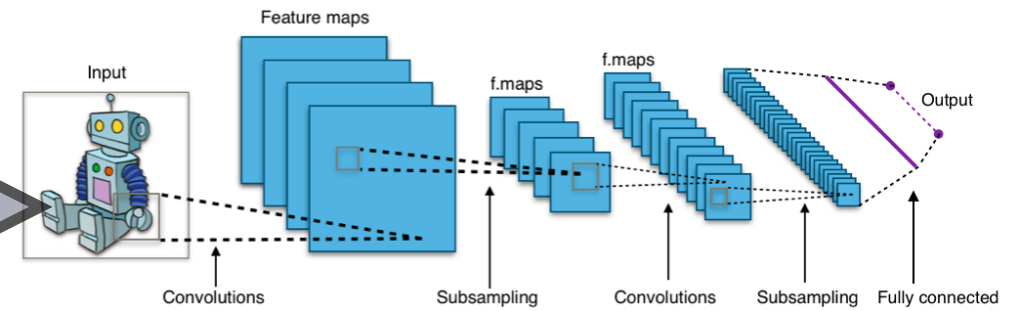
Typical Machine Learning Pipeline

Data

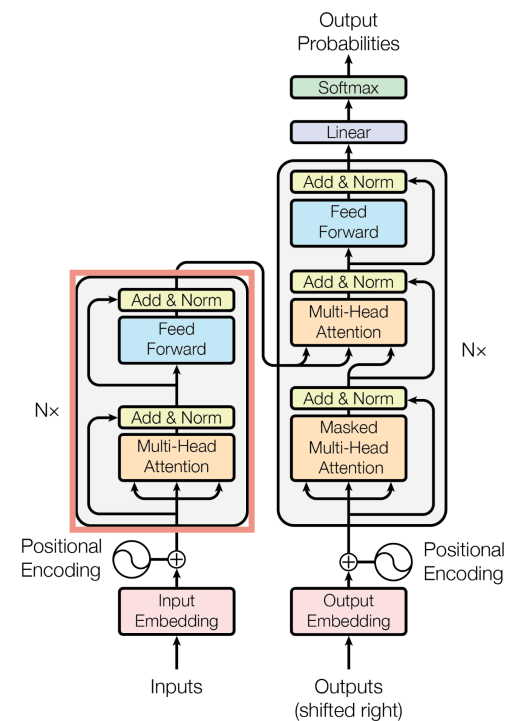


Models

2013



2022



But: It is the data that matters (a lot) too!

Data: the new scarce resource

Post-training data increasingly decides what a model can do, and good data is finite and costly.

~300T

tokens of quality public human text; projected to be fully used between 2026 and 2032

\$60M / year

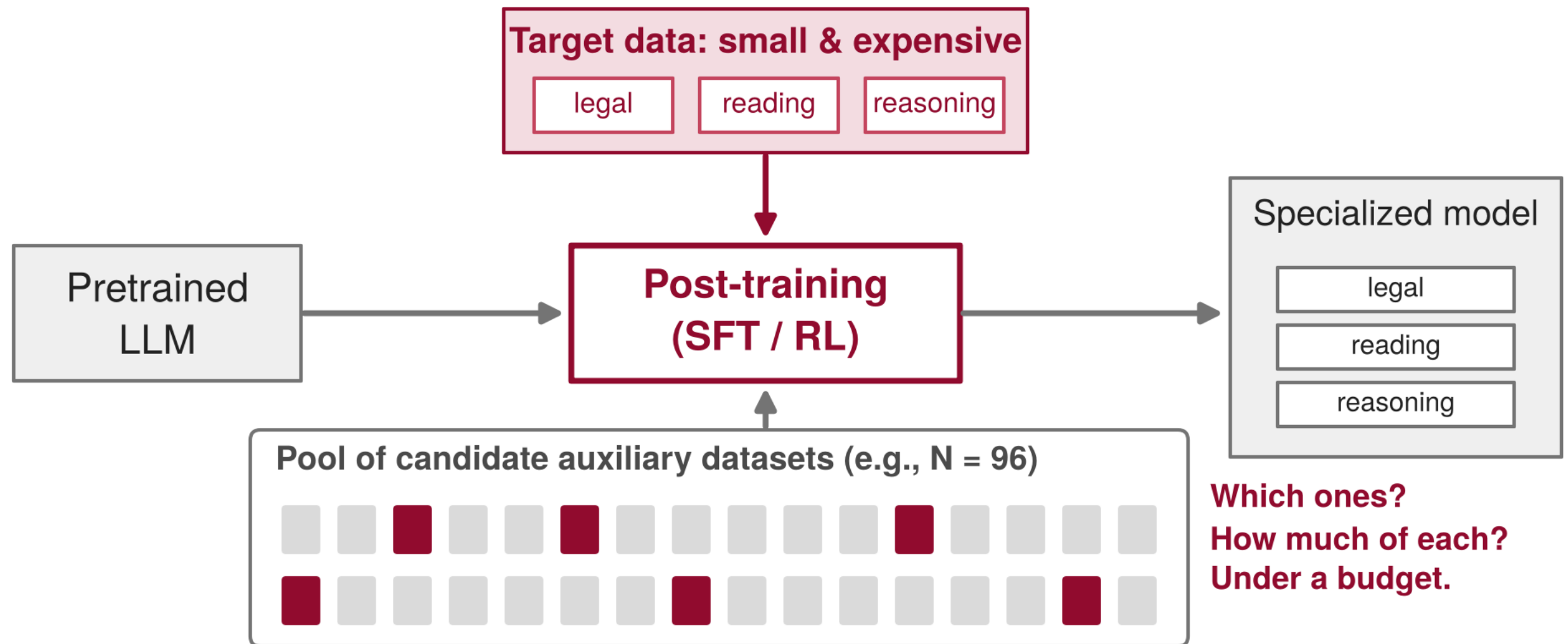
Reddit's reported AI data-licensing deal with Google (2024); News Corp–OpenAI: ~\$250M over 5 years

Experts

Target tasks (law, medicine, science) need scarce expert annotation

Not “how much data?” but “which data, and how much of each?”

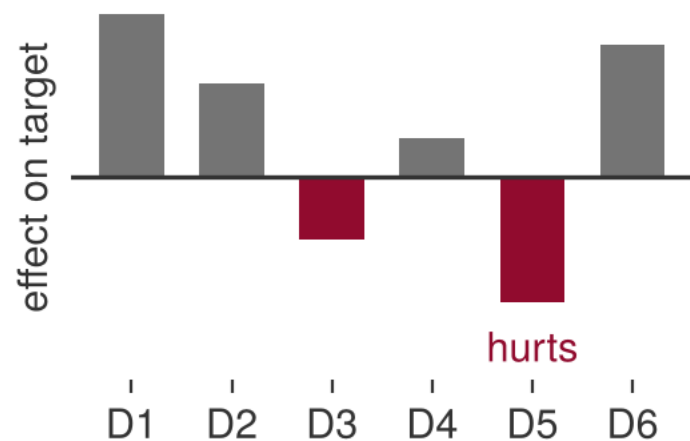
The post-training data decision



Given a few target examples and a fixed budget: which auxiliary datasets should we train on, and in what proportions?

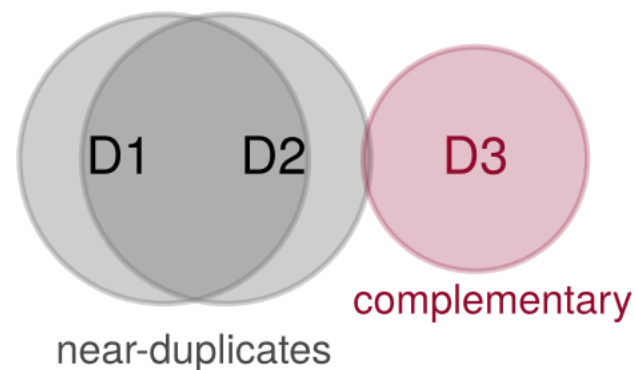
- Our study: 8 target tasks (legal, reading comprehension, logical reasoning, commonsense, ...), 96 candidate datasets, models from 0.6B to 70B parameters

Why is choosing data hard?



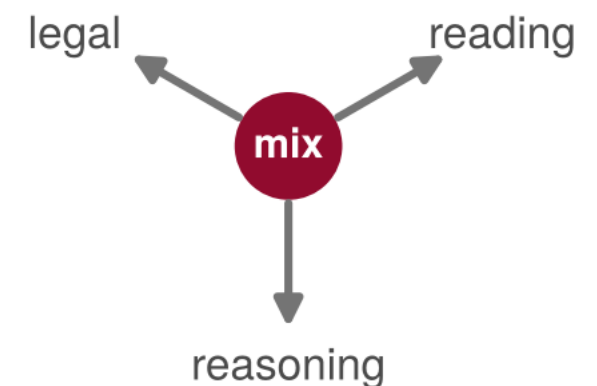
Help or hurt

Transfer is unpredictable: some datasets improve the target, others degrade it



Overlap

Two individually great datasets can be redundant together; value depends on the rest of the mix



Many goals

One shared mixture must serve several capabilities that can conflict

Brute force is hopeless: 96 datasets $\rightarrow 2^{96} \approx 8 \times 10^{28}$ subsets, each needing a training run

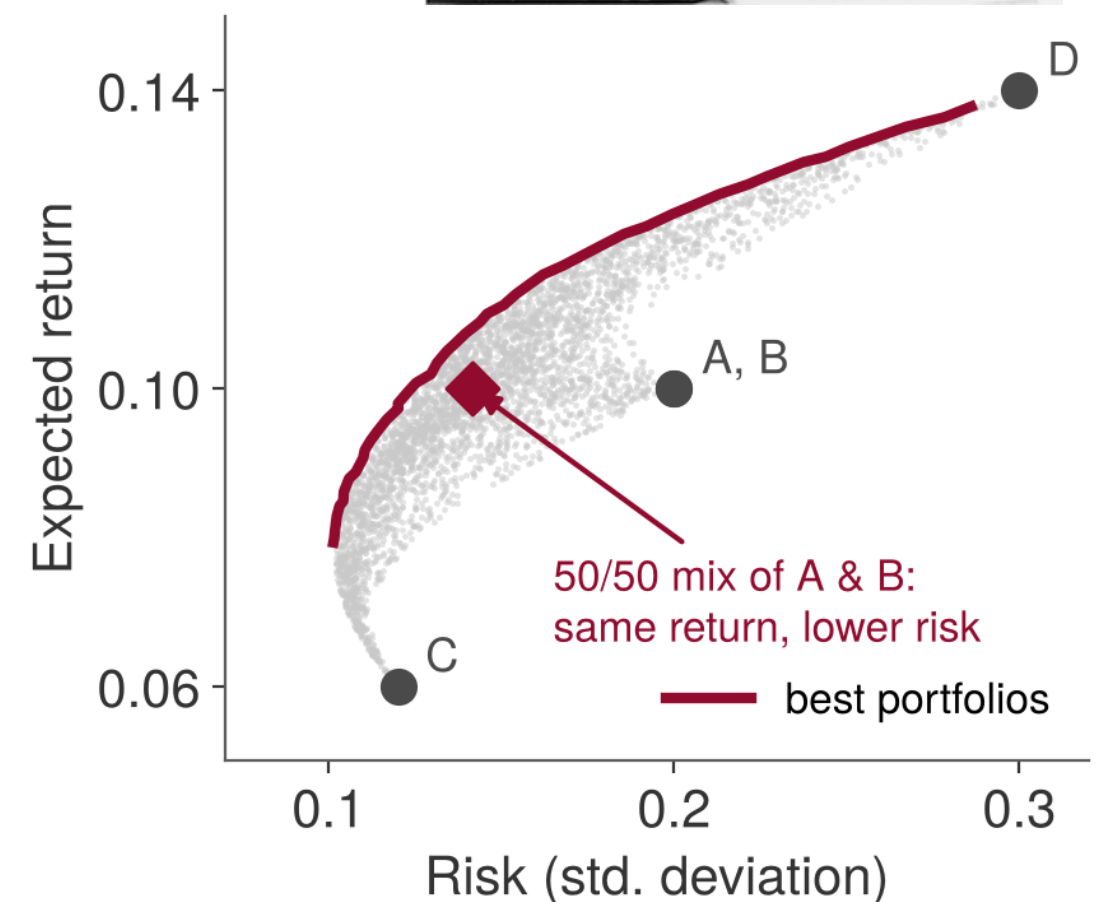
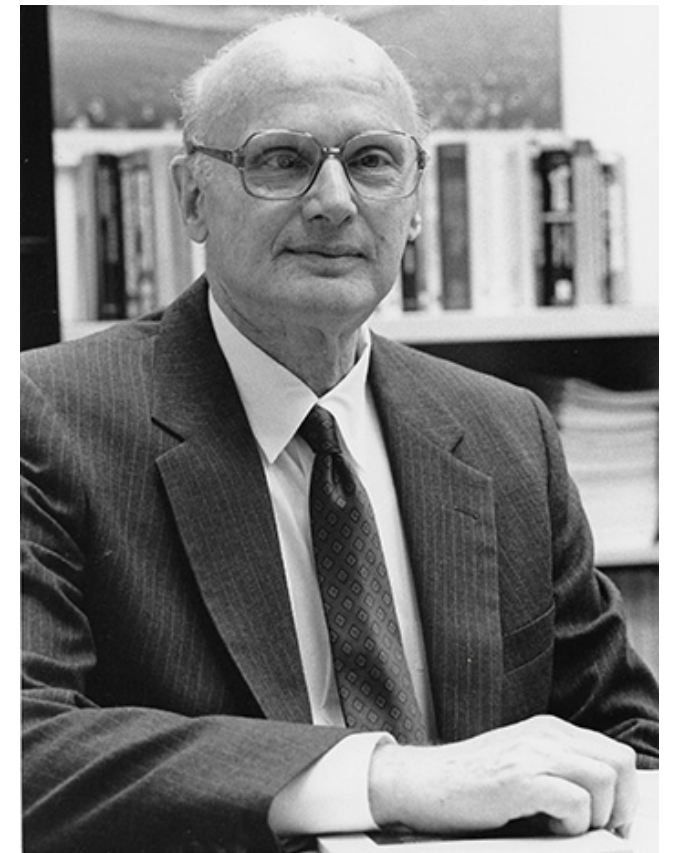
We need a principled and cheap way to value datasets jointly

An idea from finance: diversify

Markowitz portfolio theory (1952; Nobel Prize in Economics, 1990)

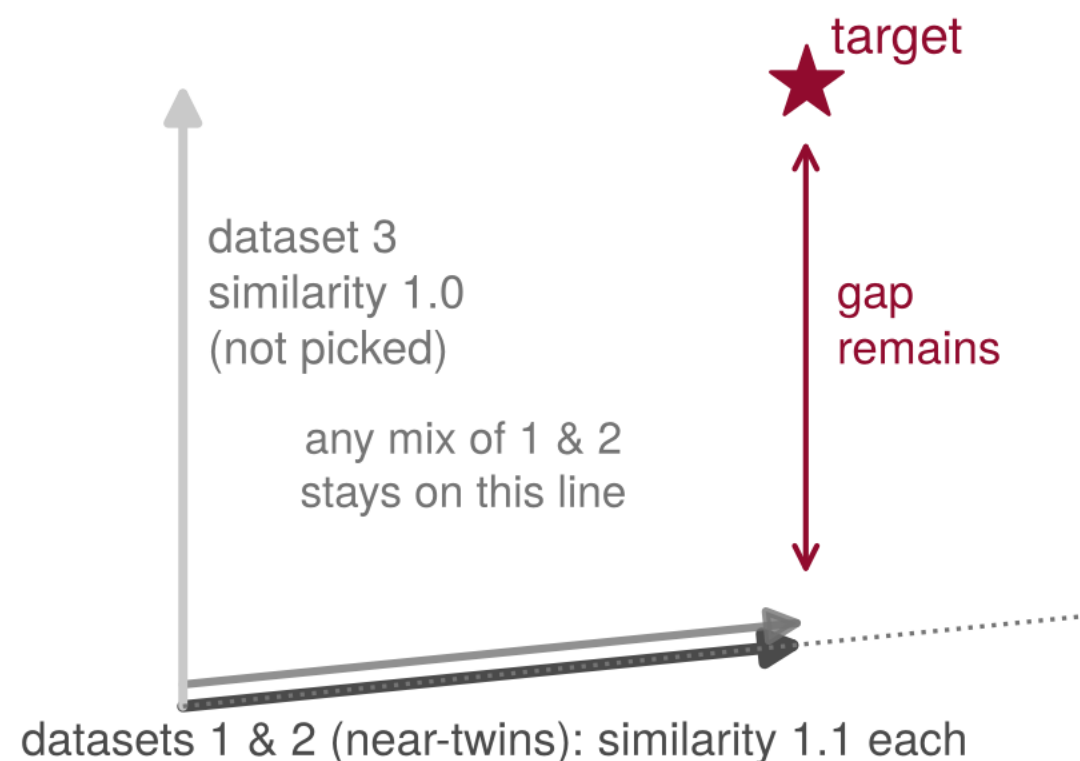
$$\max_w \underbrace{\mu^\top w}_{\text{expected return}} - \frac{\gamma}{2} \underbrace{w^\top \Sigma w}_{\text{risk}}$$

- Assets have uncertain returns, and their risks are correlated
- Choose weights w that balance expected return against risk
- Two assets, each with 10% return and variance 0.04, uncorrelated: a 50/50 mix keeps the 10% return and halves the variance (0.02)



Most similar $\neq$ most valuable

Greedy: pick the most similar



Portfolio: pick complementary



Summarize each dataset by its gradient: the direction in which it would move the model

- Greedy ranking by similarity picks the two near-duplicates (score 1.1 each) over the complementary dataset (1.0), and never reaches the target
- The portfolio view drops the duplicate and adds the complementary one: $1 \times \text{dataset 1} + 0.9 \times \text{dataset 3} = \text{target}$

One training step is a portfolio problem

Target improvement after one training step on the mixture w :

$$L_{\text{tar}}(\theta_0) - L_{\text{tar}}(\theta_0 - \eta \Phi w) \approx \eta \underbrace{\beta^\top w}_{\text{return}} - \frac{c\eta^2}{2} \underbrace{w^\top K w}_{\text{risk}}$$

$\beta_i = \langle g_i, g_{\text{tar}} \rangle$ alignment of dataset i with the target

$K_{ij} = \langle g_i, g_j \rangle$ overlap between datasets i and j

→ a convex quadratic program: kernel mean matching in gradient space

$$\min_{\|w\|_1 \leq B} \frac{1}{2} w^\top K w - \lambda \beta^\top w \iff \min_{\|w\|_1 \leq B} \|\Phi w - \lambda g_{\text{tar}}\|_2^2$$

- Intuition: make the gradient of the data mixture match the gradient of the target
- Convex: global optimum in seconds; needs only gradients, no retraining
- Same quadratic form as Markowitz, with return = alignment and risk = overlap

Data selection = portfolio construction

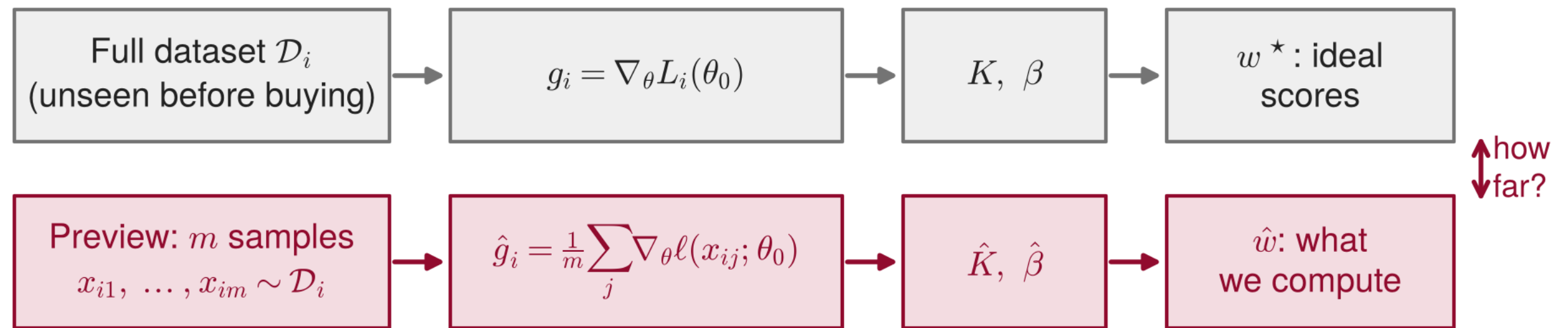
	Finance (Markowitz, 1952)	Data selection (post-training)
Asset i	R_i (random return)	$g_i = \nabla_{\theta} L_i(\theta_0)$ (gradient of $\mathcal{D}_i$)
Weights	$\mathbf{1}^T w = 1, \quad w \geq 0$	$w \in \mathcal{W}_B = \{w : \ w\ _1 \leq B\}$
Portfolio	$R_w = \sum_i w_i R_i$	$\Phi w = \sum_i w_i g_i$ (update direction)
Return	$\mathbb{E}[R_w] = \mu^T w, \quad \mu_i = \mathbb{E}[R_i]$	$\beta^T w, \quad \beta_i = \langle g_i, g_{\text{tar}} \rangle$
Risk	$\text{Var}(R_w) = w^T \Sigma w, \quad \Sigma_{ij} = \text{Cov}(R_i, R_j)$	$w^T K w, \quad K_{ij} = \langle g_i, g_j \rangle$
Objective	$\max_w \mu^T w - \frac{\gamma}{2} w^T \Sigma w$	$\max_{w \in \mathcal{W}_B} \lambda \beta^T w - \frac{1}{2} w^T K w$

One training step gives exactly this form (Taylor expansion, Hessian $\approx cI_d$):

$$\mu = \eta \beta, \quad \Sigma = c \eta^2 K \geq 0$$

Same convex program: return = alignment, risk = redundancy

Theory: valuing data from a small preview



Theorem (informal; Zeng et al., ICML 2026)

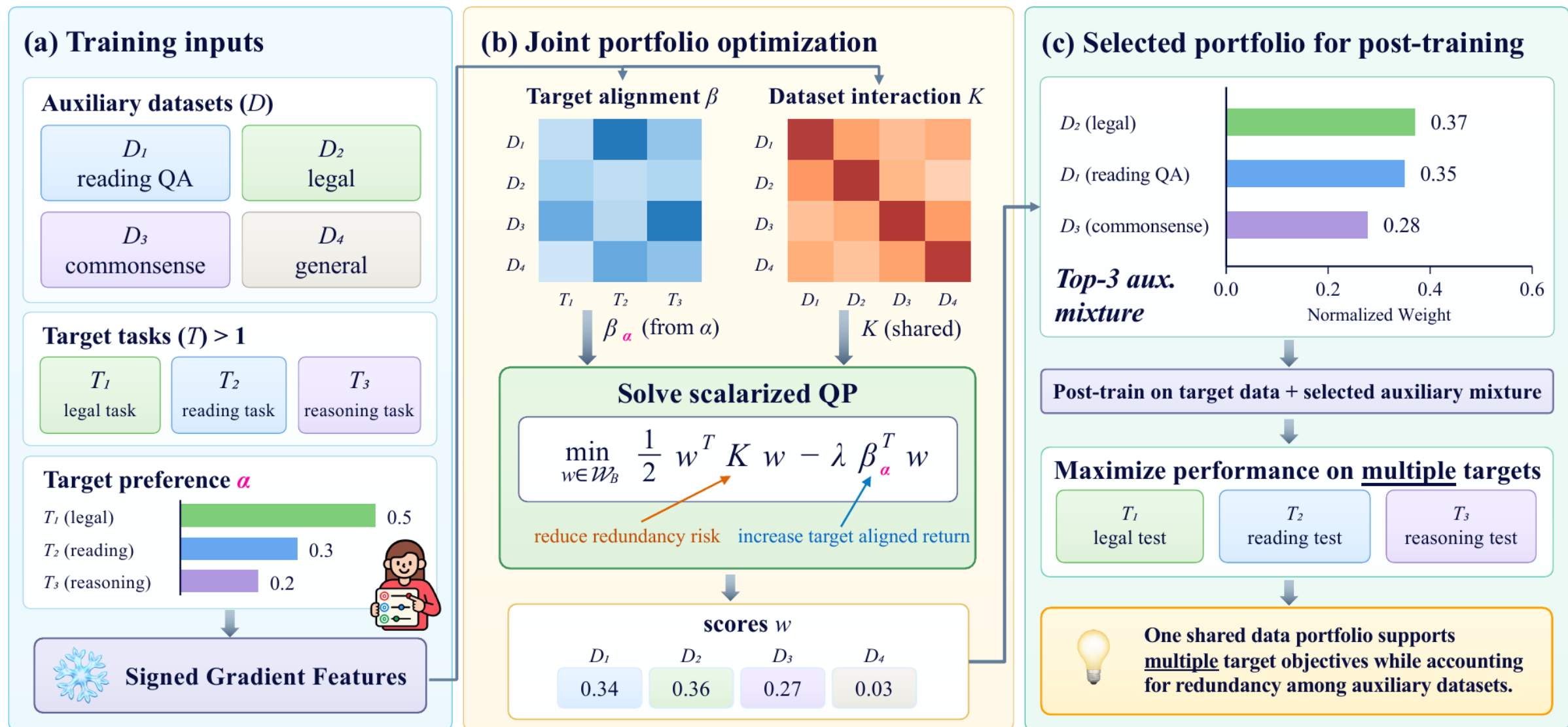
If $\|\nabla_{\theta} \ell(x; \theta_0)\|_2 \leq C$ and $\lambda_{\min}(K) \geq \mu > 0$, then w.p. $\geq 1 - \delta$:

$$\|\hat{w} - w^*\|_2 = \tilde{O}\left(\frac{kN}{\mu\sqrt{m}}\right)$$

- Error decays at the Monte Carlo rate $1/\sqrt{m}$: a small preview suffices
- Harder when datasets are nearly redundant (small μ) or k, N are large
- Extends to many targets (uniformly over α) and to signed gradients

Many targets, one data portfolio

Post-training targets several capabilities at once, with user priorities α



$$\beta_\alpha = \sum_{r=1}^M \alpha_r \beta_r \quad \Rightarrow \quad \min_{\|w\|_1 \leq B} \frac{1}{2} w^T K w - \lambda \beta_\alpha^T w$$

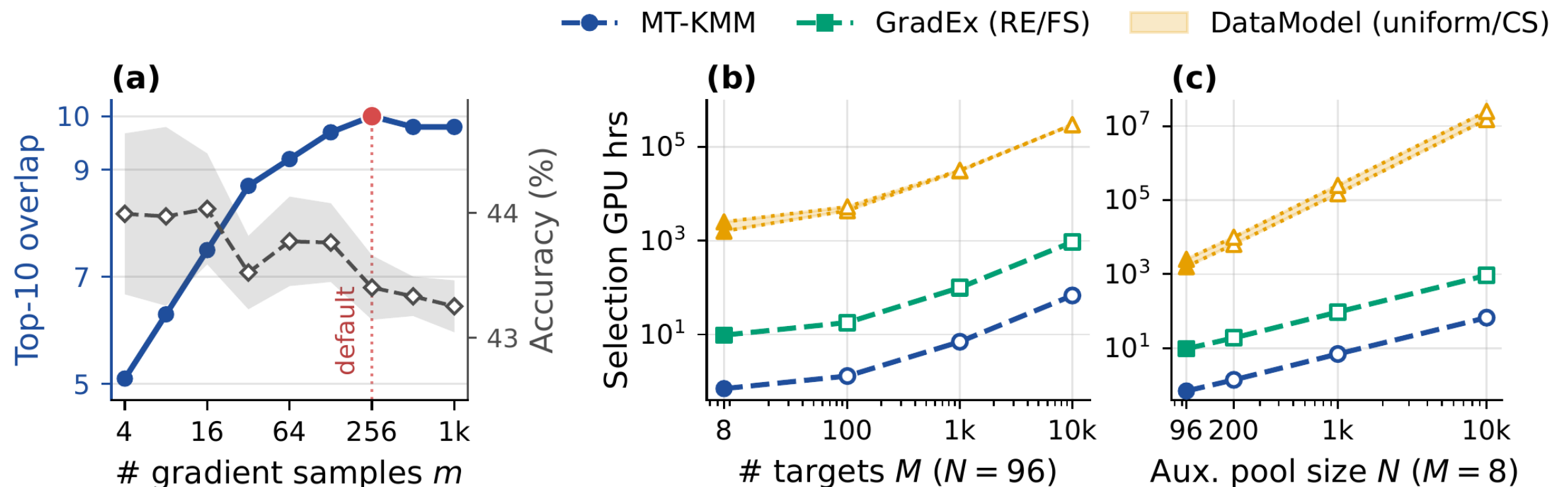
- One redundancy matrix K , shared across all priorities: change α and re-solve a tiny convex program
- For any $\alpha > 0$, the solution is Pareto-optimal across the targets

Making it scale

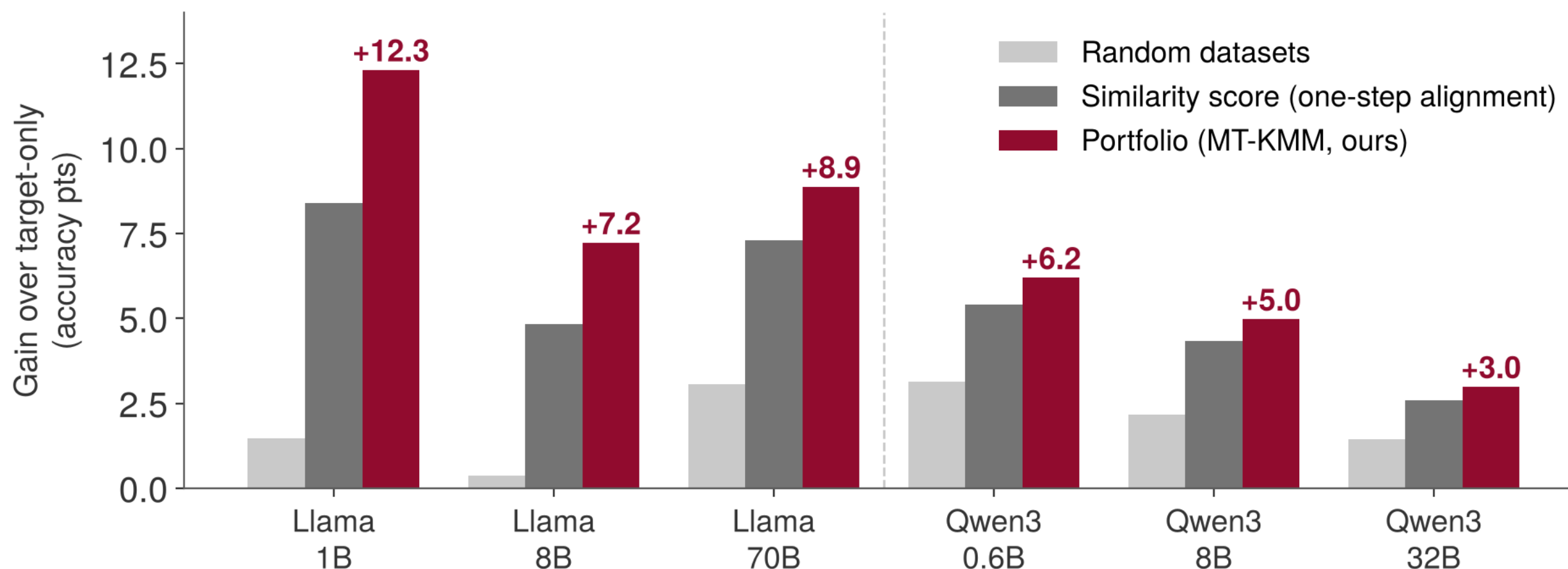
Signed gradients, computed once on a small model

$$\text{Adam's 1st step: } -\eta \frac{\hat{m}_1}{\sqrt{\hat{v}_1}} \approx -\eta \text{sign}(g) \quad \implies \quad \phi_i = \frac{1}{\sqrt{d}} \text{sign}(g_i)$$

- Matches how LLMs are trained (Adam); removes loss-scale and length biases
- Computed once on a $\leq 1\text{B}$ model (256 samples per dataset), reused up to 70B
- Theory: scores converge at rate $\tilde{O}(1/\sqrt{m})$ in the preview size m
- Cost: $O(N + M)$ gradient passes, $\sim 14\times$ cheaper than GradEx

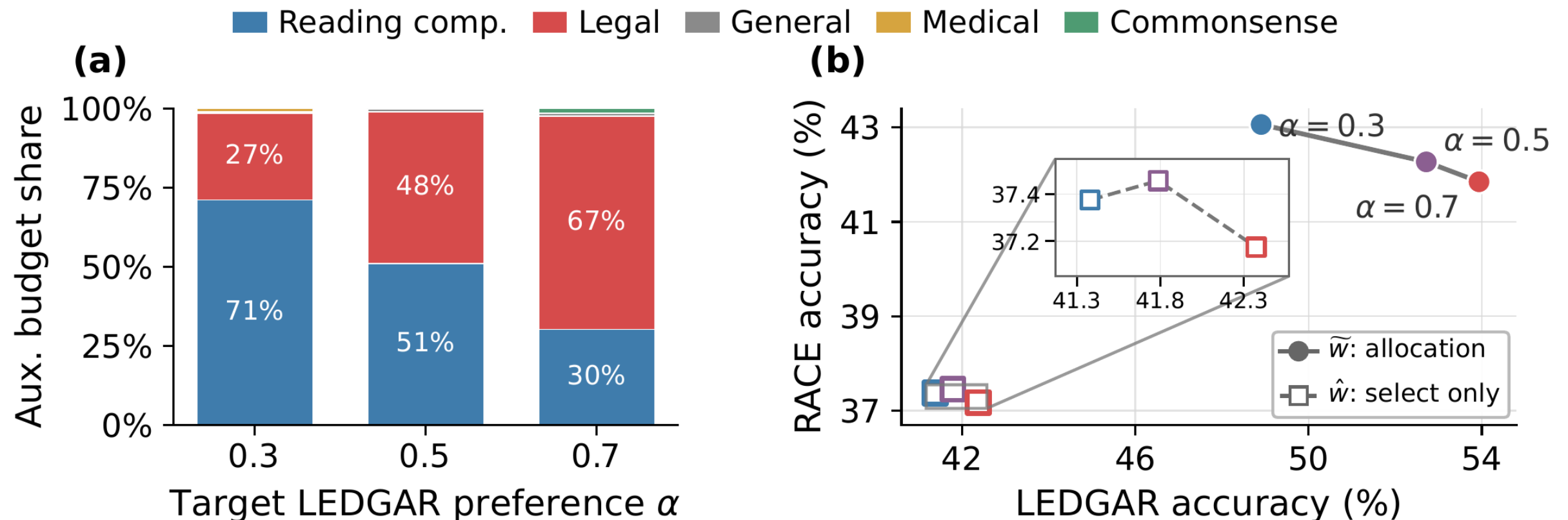


Results: best at every scale



- Highest mean accuracy over 8 targets at every scale: +12.3 points over target-only training on Llama-1B, +8.9 on Llama-70B
- Selection computed once on the smallest model of each family, then reused
- Also best for RL post-training (GRPO, 4 targets): 42.3% vs. 40.8% for the next-best method

Steering with preferences



- Raising the legal priority α from 0.3 to 0.7 shifts the budget from reading-comprehension to legal data (27% $\rightarrow$ 67%)
- Accuracy follows the knob: legal (LEDGAR) +5.0 points, reading (RACE) -1.2 points
- Both ingredients matter: ignoring redundancy costs 1.4–2.9 points; equal weighting costs 2.1–4.4 points

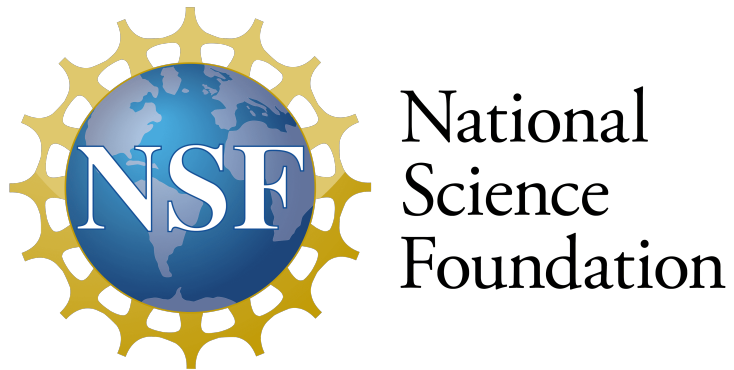
Takeaways

Choosing training data is portfolio construction

- A dataset's value = alignment with the target – redundancy with the rest of the mix
- Gradients turn this into a convex program: global optimum in seconds, no retraining
- One shared redundancy matrix serves any priorities over many targets (Pareto-optimal)
- Cheap and transferable: computed on a 1B model, used up to 70B, for SFT and RL

Beyond LLMs: the same lens applies wherever scientists decide which data to collect, buy, or simulate

Thanks!



Joint work with Siqi (Cindy) Zeng and collaborators



Reference:

1. Convex Dataset Valuation for Post-Training, Zeng et al., ICML'26 (arXiv: 2605.16704)
2. Convex Multi-Target Dataset Selection for Post-Training via Portfolio Optimization, under review
3. Code: github.com/uiuctml/convex_data_valuation