Prioritized Training on Points that are Learnable, Worth Learning, and Not Yet Learned

Sören Mindermann, Jan Brauner*, Muhammed Razzak*, Mrinank Sharma*, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan Gomez, Adrien Morisot, Sebastian Farquhar, Yarin Gal*

Paper & code

We all know model training can be very slow ...

... but lots of computation and time is wasted on redundant and noisy points.

Skip points that are already learnt, not learnable, or not worth learning to accelerate training.

* image from *Pleiss, Geoff, et al. "Identifying mislabeled data using the area under the margin ranking." Advances in Neural Information Processing Systems 33 (2020)*

Online Batch Selection

$\mathcal{D}$

Draw large batch with uniform sampling

$$\mathcal{B} = \{(x_i, y_i)\}_{i=1}^{n_B}$$

Create small batch b made of the ***highest scoring*** *points in large batch $\mathcal{B}$ according to a* ***selection function***

Key contribution

Reducible Holdout Loss

A simple and principled selection function that identifies points that most reduce generalisation loss

$$b = \{(x_i, y_i)\}_{i=1}^{n_b}$$

Train on b...

... Repeat

Reducible Holdout Loss

A principled approach for (approximately) choosing points that would most improve the generalisation loss if trained on, *without* needing to train on those points.

Points with high training loss are not yet learnt ...

... but they could have high loss *because* they are noisy (unlearnable), or not worth learning (outlier, low density points)

Noisy and less relevant/outlier points are **hard to predict using a holdout set**, and thus have **high IL**

$$\underset{(x,y)\in B_t}{\arg\max} \; \underbrace{L[y \mid x; \mathcal{D}_{\mathrm{t}}]}_{\text{training loss}} - \underbrace{L[y \mid x; \mathcal{D}_{\mathrm{ho}}]}_{\text{irreducible holdout loss (IL)}}$$

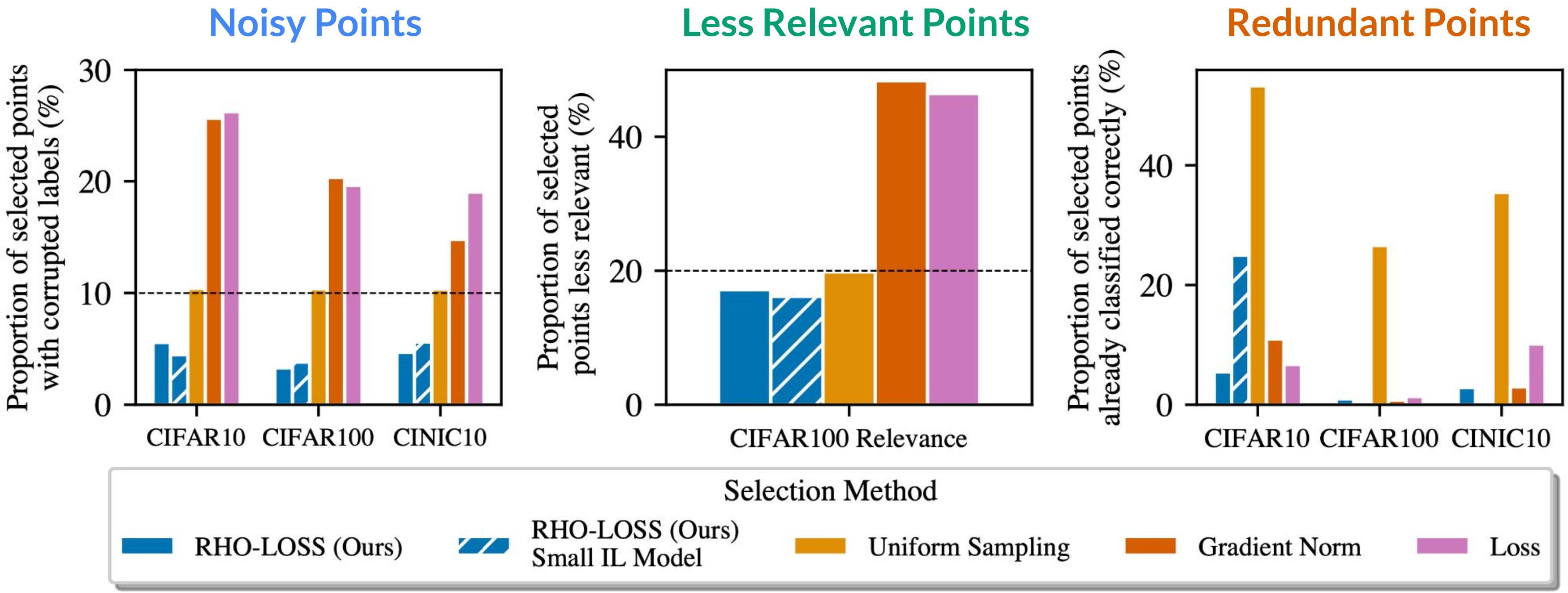
The Reducible Holdout Loss is …

… low for points that are already learnt (low training loss).
… low for points that are noisy (high training loss, high IL).
… low for points are are outliers / less relevant (high training loss, high IL).

… high for points that are learnable, worth learning, and not yet learnt!

$$\underset{(x,y)\in B_t}{\arg\max} \quad \overbrace{\underbrace{L[y \mid x; \mathcal{D}_{\mathrm{t}}]}_{\text{training loss}} \quad - \quad \underbrace{L[y \mid x; \mathcal{D}_{\mathrm{ho}}]}_{\text{irreducible holdout loss (IL)}}}^{\text{reducible holdout loss}}$$

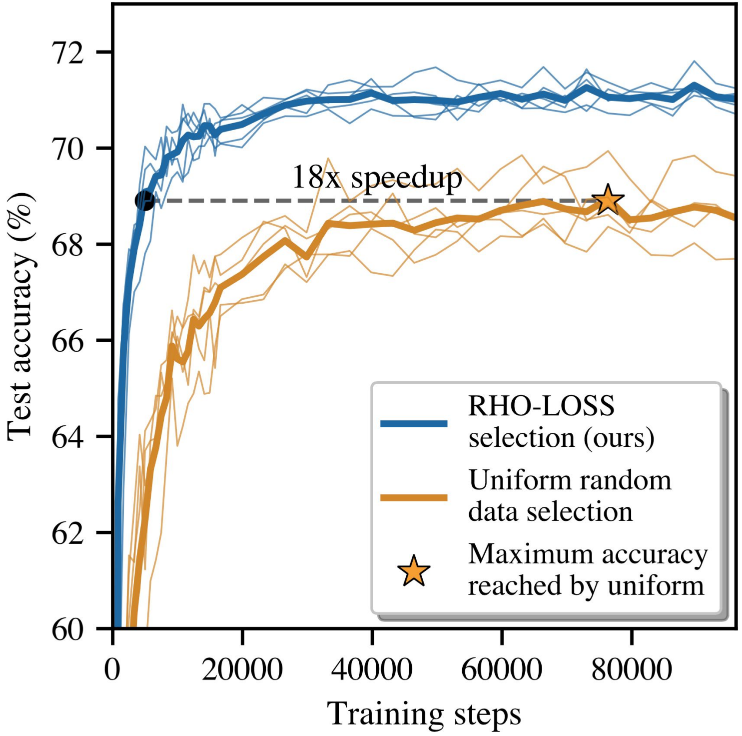
Indeed, RHO-LOSS prioritises points that are non-noisy, task-relevant and non-redundant



RHO-LOSS trains in 18x fewer steps on Clothing-1M

Clothing-1M is a large web-scraped image dataset, containing redundant and noisy data—our target application.

Further, RHO-LOSS speeds up training on a **wide range of datasets, hyperparameters, and architectures** (MLPs, CNNs, and BERT).



Thin lines: *ResNet-50, MobileNet v2, DenseNet 121, Inception v3, GoogleNet.* ***Bold lines:*** *mean across architectures.*

Thank you 😊

Come and speak to us! And check out the paper for more …

✨ re-using a single small IL model for accelerating training for multiple architectures.

✨ deriving RHO-LOSS as an efficient, cheap approximation to optimal selection, derived in the language of probabilistic modelling.

✨ explaining why prior approaches don't work robustly.

Paper & code