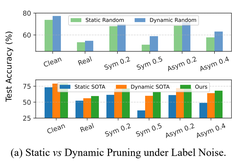
Beyond Loss Values: Robust Dynamic Pruning via Loss Trajectory Alignment

Huaiyuan Qin Muli Yang Gabriel James Goenawan Hongyuan Zhu
Institute of Advanced Intelligence and Computing (IAIC), A*STAR, Singapore

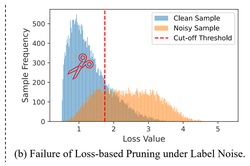
Can Dynamic Data Pruning Survive Label Noise?

1. Data pruning meets the noisy reality

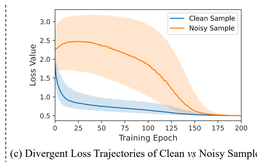
- **Data pruning** reduces training cost by discarding low-utility samples. ***Static*** coreset selection fixes the subset *before* training — once a noisy sample is selected, it stays in the loop **permanently**.
- Instead, ***Dynamic*** pruning (InfoBatch, SeTa) re-selects the subset *every epoch* and is inherently more robust: even *random* dynamic pruning beats random static pruning under all noise types.
- Modern web-scale datasets are unavoidably noisy: web crawlers, weak image–text supervision, semantic ambiguity ⇒ pruning **must** stay reliable under noise.



(a) Static *vs* Dynamic Pruning under Label Noise.



(b) Failure of Loss-based Pruning under Label Noise.

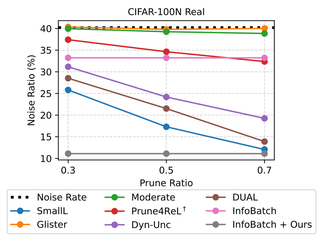


(c) Divergent Loss Trajectories of Clean *vs* Noisy Samples.

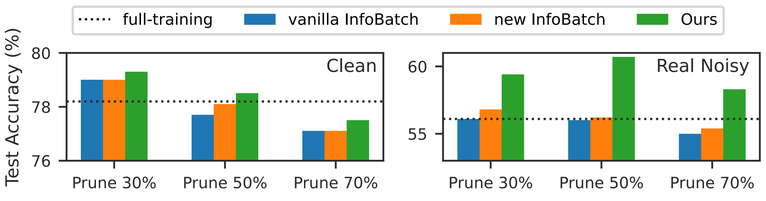
- Panel (a) Dynamic > static under every noise type; our **AlignPrune** sets a new state of the art over both dynamic (InfoBatch) and static-robust (Prune4ReL) baselines.

2. But loss-based ranking fails under noise ✗

- SOTA dynamic pruning methods rank each sample by its **per-epoch loss value**.
- Under label noise this signal **inverts**: noisy samples have *high* loss and get **kept or prioritized**, while low-loss *clean* samples are pruned (Panel (b) above).
- Retained noise **accumulates** as pruning progresses, degrading the model — the loss threshold is unreliable exactly when it matters.

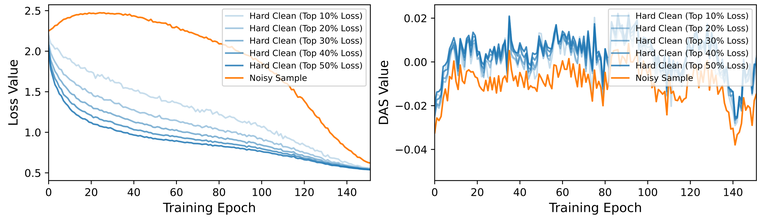


- Measured on CIFAR-100N (Real noise): baselines retain a subset nearly as **noisy as the raw data**; AlignPrune keeps it **far cleaner**, increasingly so under stronger pruning.
- The failure is **not a tuning artifact**: even a noise-*calibrated* InfoBatch barely reaches full training on Real noise — DAS ranking **clears it at every prune ratio**:



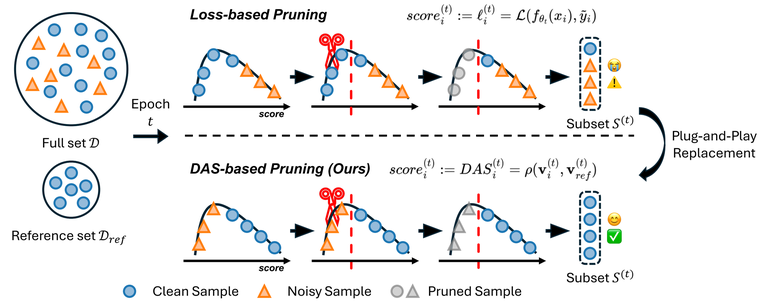
3. Our insight: trajectories separate clean from noisy ✓

- **Clean** samples follow a **smooth, monotonic loss decay** that mirrors the model's overall learning progress.
- **Noisy** samples exhibit **erratic, non-monotonic** loss patterns (Panel (c) above).
- Even **hard-but-clean** samples (also high loss!) keep clean-like *dynamics* ⇒ a trajectory-level signal separates them where a loss threshold cannot:



AlignPrune: Rank by Trajectory Alignment

1. Dynamic Alignment Score (DAS)



- Keep each sample's loss over the last N epochs as a **loss trajectory** $\mathbf{v}_i^{(t)}$; compute the average over a small **clean reference set** as $\mathbf{v}_{ref}^{(t)}$.
- Score each sample by **correlation with the clean reference trend**:

$$DAS_i^{(t)} = \rho\left(\mathbf{v}_i^{(t)},\ \mathbf{v}_{ref}^{(t)}\right), \qquad \mathbf{v}_i^{(t)} = [\ell_i^{(t-N+1)}, \ldots, \ell_i^{(t)}]$$

- **Positive DAS** = synchronized with clean learning dynamics ⇒ likely clean, **keep**; **Negative DAS** = conflicting dynamics ⇒ likely noisy, **prune**.

2. AlignPrune: A plug-and-play module

- Drop-in replacement of the ranking score in existing dynamic pruners: $score_i^{(t)} := DAS_i^{(t)}$, with **no change** to the model architecture, loss, or training pipeline of InfoBatch / SeTa.
- Preserves InfoBatch's **unbiased gradient expectation** (rescaling kept intact).
- Per-sample trajectories live in a window-N **memory bank**; DAS is updated once per epoch with vectorized ops ⇒ negligible overhead.
- Insensitive to hyper-parameters: window $N = 5$–25 all work; cosine ≈ Pearson; higher prune probability r helps under noise.

3. Better accuracy, Better efficiency ✓

Efficiency on CIFAR-100N (ResNet-18, 50% prune, Real noise, 200 epochs)

	Static	Dynamic	SeTa	InfoBatch	**Ours**	Full Data
Acc (%) ↑	51.8	54.1	55.7	56.0	**60.7**	56.1
Time (mins) ↓	24.7	24.4	25.8	23.9	**22.5**	46.7

- **Above full-data accuracy** (60.7 vs. 56.1) at **2.1× lower time cost** (22.5 vs. 46.7 min) — and **+4.7** over the strongest pruning baseline.

4. Scaling to ImageNet-1K, CNNs & ViTs ✓

ImageNet-1K across modern architectures (Δ vs. full training)

	Model Architecture →	ConvNeXt		DeiT		Swin		Mean
	Pruning Method ↓	Tiny	Base	Small	Base	Tiny	Base	Δ
Clean	Full-training	73.3	75.6	71.9	77.8	73.5	76.9	–
Clean	InfoBatch	73.2 −0.1	75.4 −0.2	71.5 −0.4	**77.8** +0.0	**73.6** +0.1	77.1 +0.2	−0.1
Clean	**InfoBatch + Ours**	**73.4** +0.1	**75.7** +0.1	**72.1** +0.2	**77.8** +0.0	**73.6** +0.1	**77.2** +0.3	**+0.1**
Noisy	Full-training	66.4	67.3	64.7	68.9	65.0	67.8	–
Noisy	InfoBatch	65.8 −0.6	66.8 −0.5	64.1 −0.6	68.1 −0.8	64.3 −0.7	66.9 −0.9	−0.7
Noisy	**InfoBatch + Ours**	**67.2** +0.8	**67.8** +0.5	**65.5** +0.8	**69.4** +0.5	**65.6** +0.6	**68.2** +0.4	**+0.6**

- Under noise, InfoBatch falls **below** full training on *every* architecture (mean −0.7); AlignPrune **recovers above it** (mean **+0.6**) while staying lossless on clean data.

Robust SOTA Across Noisy Benchmarks

1. CIFAR-100N / CIFAR-10N: consistent gains ✓

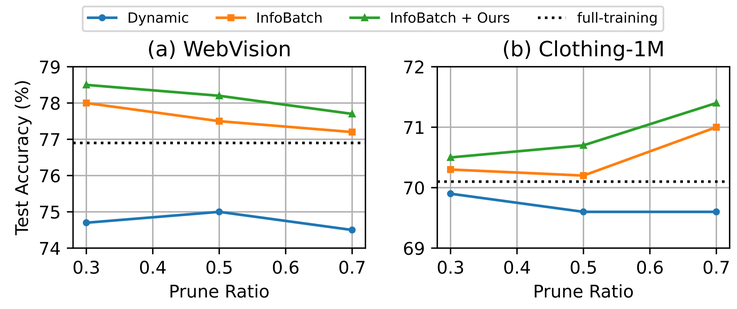
CIFAR-100N (ResNet-18), prune ratio ∼50%

Noisy Type → / Pruning Method ↓	Clean	Real	Symmetric			Asymmetric		Mean
			0.2	0.5	0.8	0.2	0.4	Δ
Full-training	78.2	56.1	71.4	58.6	39.8	72.4	63.3	–
Static Random	72.1 −6.1	51.8 −4.3	64.0 −7.4	47.2 −11.5	22.9 −16.9	65.3 −7.0	54.6 −8.7	−8.8
Dynamic Random	75.3 −2.9	54.1 −2.0	70.4 −1.0	59.5 +0.9	40.7 +0.9	70.1 −2.3	64.8 +1.6	−0.7
InfoBatch	77.7 −0.5	56.0 −0.1	71.3 −0.1	60.5 +1.9	42.2 +2.4	71.8 −0.6	65.2 +2.0	+0.7
InfoBatch + Ours	78.5 +0.3	60.7 +4.6	71.6 +0.3	62.0 +3.4	42.6 +2.8	72.6 +0.3	68.6 +5.3	**+2.4**
SeTa	77.5 −0.7	55.7 −0.4	70.7 −0.7	60.0 +1.4	40.5 +0.7	71.9 −0.5	64.5 +1.2	+0.2
SeTa + Ours	78.4 +0.2	56.3 +0.1	71.2 −0.2	61.0 +2.4	41.9 +2.1	72.2 −0.2	66.0 +2.7	**+1.0**

CIFAR-10N (ResNet-18), prune ratio ∼50%

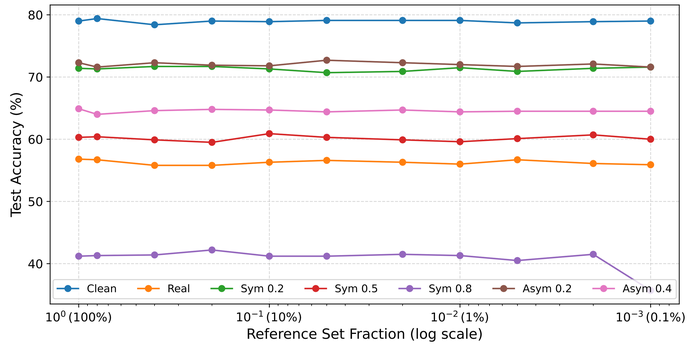
Noisy Type → / Pruning Method ↓	Clean	Real		Symmetric			Asymmetric		Mean
		Real-A	Real-W	0.2	0.5	0.8	0.2	0.4	Δ
Full-training	95.6	90.7	78.3	91.3	85.4	65.6	90.6	85.8	–
Static Random	93.3 −2.3	88.6 −2.1	72.4 −5.9	87.4 −3.9	77.5 −8.0	51.4 −14.2	85.8 −4.9	77.9 −7.9	−6.1
Dynamic Random	94.5 −1.1	89.2 −1.5	76.1 −2.3	88.9 −2.4	83.7 −1.8	63.3 −2.3	90.6 +0.0	83.7 −2.1	−1.7
InfoBatch	95.0 −0.6	90.5 −0.2	77.7 −0.6	92.0 +0.7	87.5 +2.1	64.3 −1.3	92.0 +1.3	87.5 +1.7	+0.4
InfoBatch + Ours	95.0 −0.6	91.3 +0.6	82.4 +4.1	92.8 +1.5	87.6 +2.1	64.5 −1.1	92.1 +1.5	89.8 +4.0	**+1.5**
SeTa	94.8 −0.8	89.4 −1.3	78.5 +0.2	91.3 +0.0	86.0 +0.5	53.0 −12.7	91.7 +1.1	87.9 +2.1	−1.4
SeTa + Ours	94.9 −0.7	89.7 −1.0	79.3 +1.0	91.9 +0.6	87.6 +2.2	60.2 −5.5	92.2 +1.6	88.5 +2.7	**+0.1**

2. Real-world noise at scale ✓



- **WebVision** (from-scratch) and **Clothing-1M** (fine-tuning): **+0.5** average over InfoBatch across all pruning ratios.

3. Barely needs clean data ✓



- Shrinking the clean reference set from 100% to **0.1% (just 10 images)** barely moves accuracy across all noise types.
- **Pseudo-clean** reference sets estimated from the *noisy* data itself (SmallL/Moderate) match truly clean ones ⇒ **no clean labels required in practice**.

Paper: arXiv:2604.07306
Code: github.com/leonqin430/AlignPrune