Layer-wise Diagnostic Probing to Enhance Selectivity in Machine Unlearning

Anudeep Vurity · Zhisheng Yan · Massimiliano Albanese

George Mason University, Fairfax, Virginia, USA

Class-level unlearning

Existing unlearning applies a **global update** to the whole model.

Nobody asks **which layers hold the forget data.**

The cost of not knowing

Updates land everywhere, **including layers that had nothing to do with the forgotten class**. That is where retention damage comes from.

What we add

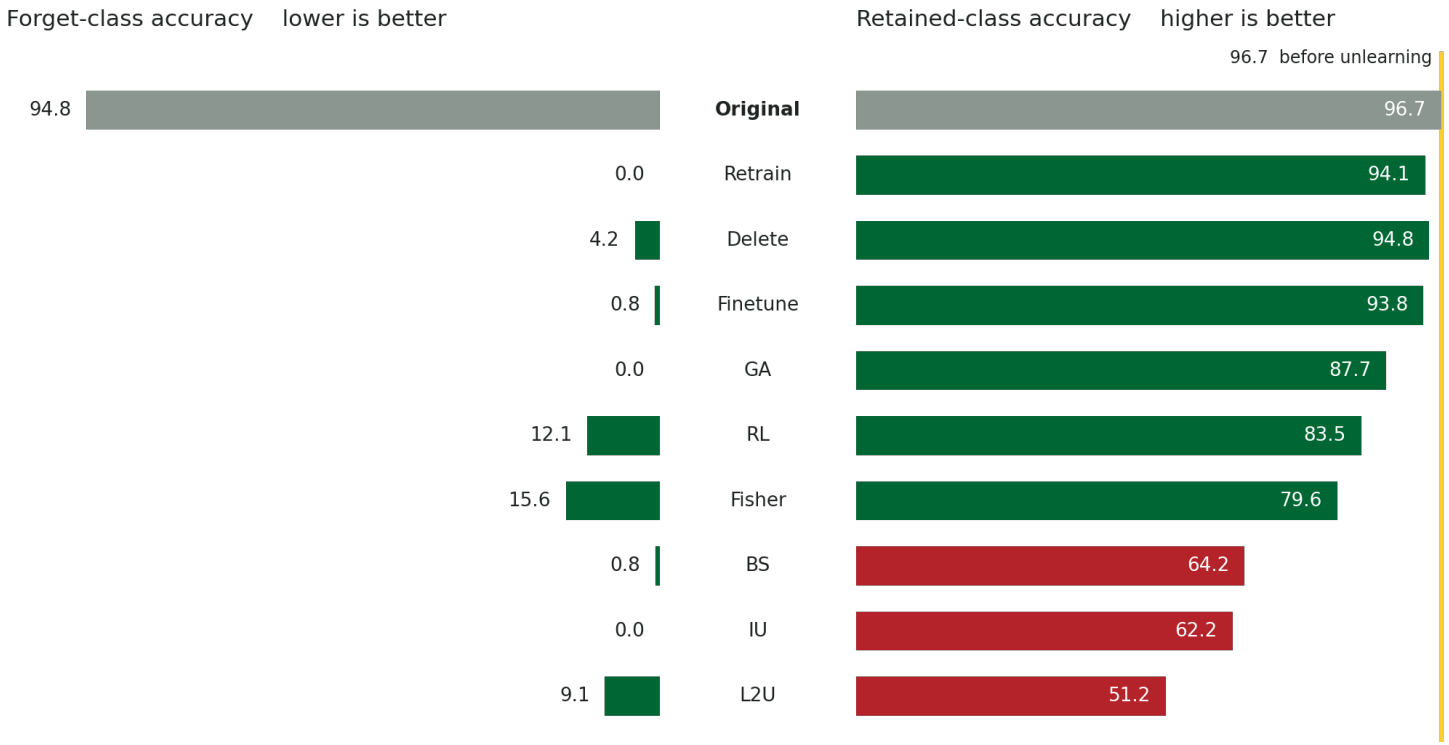
A lightweight probe on each layer that scores how strongly that layer responds to the forget set.

Our Focus

Unlearning aimed at the layers that matter and freezes the layers that do not.

MOTIVATION

Unlearning is applied uniformly to a model that stores the forget class unevenly



CIFAR-10, ResNet-18. All eight methods reach near-zero forget accuracy; retention spans 51.2 to 94.8.

The gap

- Forget-class influence varies by depth.
- Residual paths spread that influence; VGG concentrates it late.
- Every method updates all parameters with **no layer-level estimate**.
- No whole-network measure of layer-level forget sensitivity exists.

THE QUESTIONS

Three questions this paper answers

RQ1 Where does memorization of the forget class actually live in the architecture?

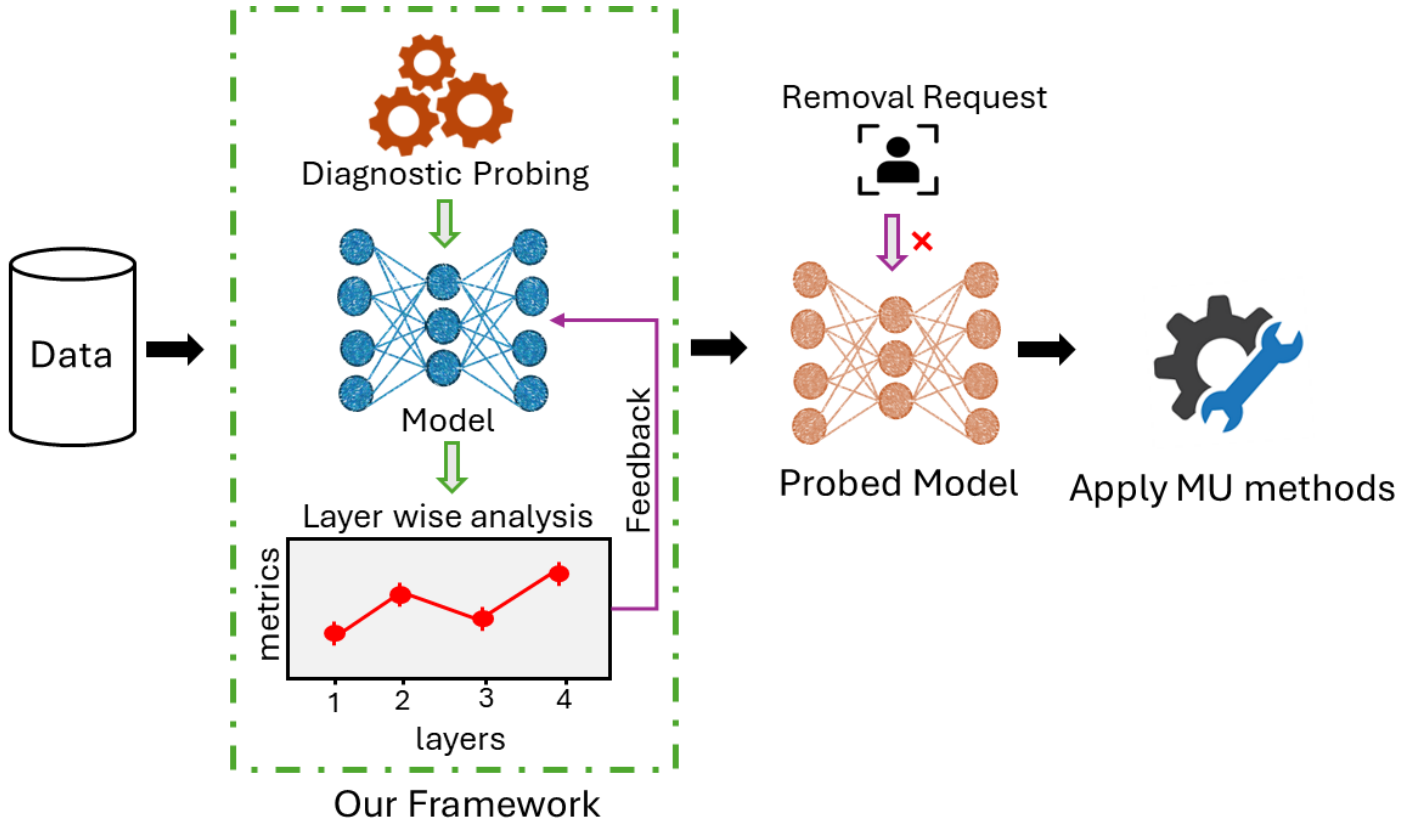
RQ2 Can layer-level diagnostic signals make forgetting more selective without hurting generalization?

RQ3 Does adding the probe improve existing unlearning methods on forgetting and retention?

The results answer them in order: RQ1 on the layer maps, RQ2 on the latent space, RQ3 on the baseline tables.

THE IDEA

Diagnose first, then unlearn



The probe runs once on a frozen model. It costs 1.35 MFLOPs and 126 seconds, and adds nothing to the unlearning step itself.

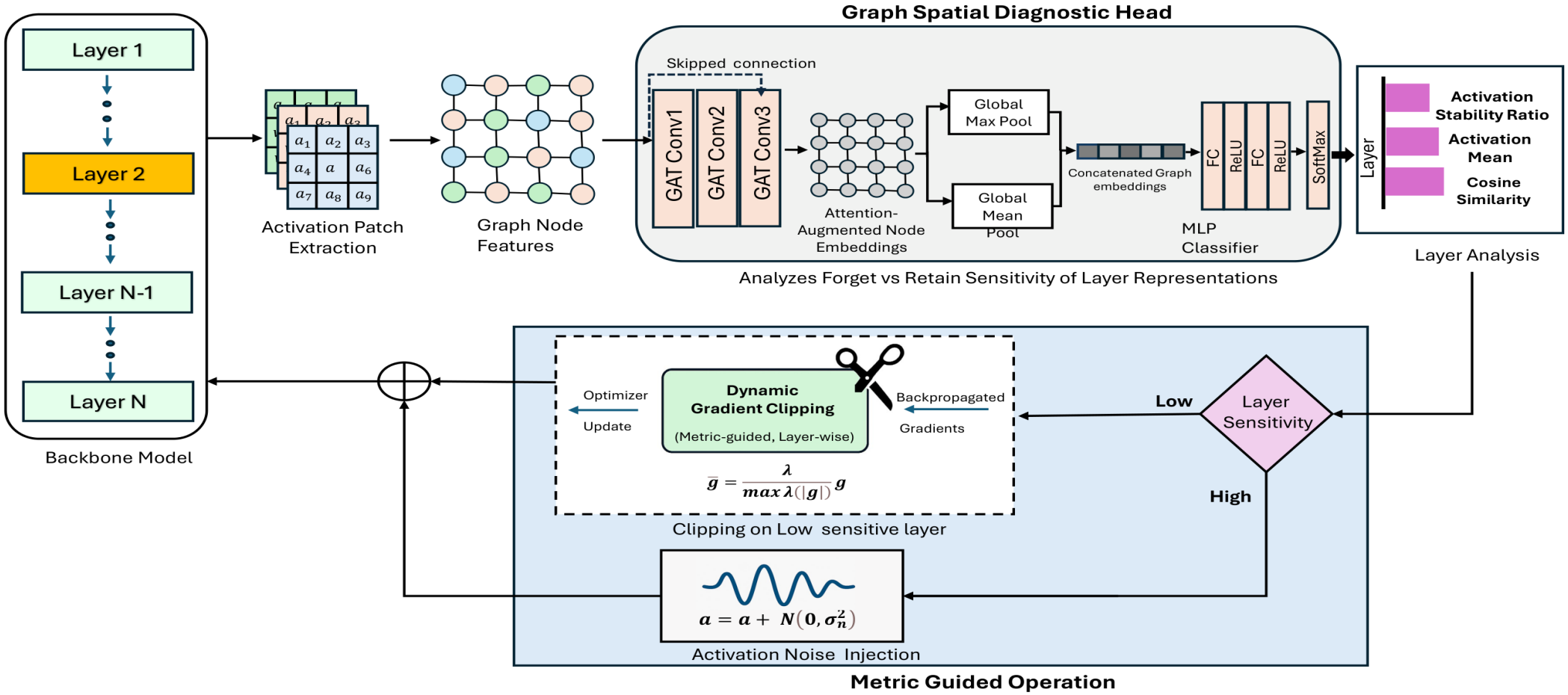
ProGraDx

Probing Graph Diagnostics for Unlearning

- Attach a probe to every intermediate layer of a frozen backbone.
- Each probe is trained to tell forget-set activations from retain-set activations.
- Score each layer, label it high or low sensitivity.
- Feed those labels back to steer the unlearning update.

FRAMEWORK

ProGraDx end to end



Top: activations become graph nodes, a three-layer GAT scores the layer. Bottom: high-sensitivity layers get activation noise, low-sensitivity layers get gradient clipping.

THE Layer METRIC

Activation Stability Ratio

Step 1: the influence map

$$I_l = ReLU(a_l \odot g_l)$$

Activation times gradient, rectified. Keeps only what pushes the loss in the relevant direction.

Step 2: pool, then take mean over spread

$$ASR_l = \mu_l / (\sigma_l + \varepsilon)$$

High ASR means a focused, consistent influence pattern.

Step 3: label the layer

High sensitivity if ASR is above the 65th percentile across all layers. Low sensitivity otherwise.

The 65th percentile is tuned empirically. The ablation later shows why it lands there.

ASR drives the decision

- Cosine similarity and activation mean are also computed.
- Those two are used as additional interpretation tools.

ASR threshold

HIGH ASR

Inject activation noise

$$\tilde{a}_l = a_l + \epsilon, \quad \epsilon \sim N(0, \sigma_l^2 \alpha^2)$$

- Gaussian noise scaled to the layer's own activation spread, with α = 0.25.
- Disrupts memorized patterns without deleting a single weight.
- Applied only to layers the probe flagged.

LOW ASR

Clip the gradient

$$\nabla L_l \leftarrow clip(\nabla L_l, -\gamma, +\gamma), \quad \gamma = 0.001$$

- Holds these layers close to the original model.
- Protects the general features that have nothing to do with the forgotten class.
- No clipping on high-sensitivity layers.

SETUP

Experimental setup

3 datasets

CIFAR-10 (single-class forgetting)
CIFAR-100 (20 forget classes)
Tiny ImageNet

Class-centric forgetting, same splits for every method.

3 architectures

ResNet-18, residual connections
VGG-16, deep and sequential, no skips
ViT-S/16, patch-wise self-attention

Chosen to vary depth and inductive bias.

8 baselines

Finetune, Gradient Ascent, Random Label, Fisher Forgetting, Influence Unlearning, L2U, Boundary Shrink, DELETE.

Plus Original and Retrain as reference points.

Every baseline runs twice: as published, then with the probe attached. No baseline hyperparameter tuning, official implementations at default settings, so the only difference is the probe.

RESULT

CIFAR-10, single-class forgetting

Each metric is shown twice: the published method, then the same method with the probe. Green means the probe helped, red means it hurt.

Method	Acc_ft ↓ better		Acc_rt ↑ better		MIA ↓ better	
	base	+ probe	base	+ probe	base	+ probe
Original	94.83	78.20	96.71	90.54	99.36	90.54
Retrain	0	–	94.06	–	0	–
Finetune	0.81	**0**	93.77	**93.18**	0	0
GA	0	0	87.65	**88.14**	0.07	**0**
RL	12.05	**16.41**	83.51	**87.60**	7.76	**5.14**
L2U	9.10	**0.10**	51.21	**52.16**	38.14	**0**
BS	0.80	**0**	64.18	**65.16**	0.48	**0**
IU	0	0	62.24	**65.14**	0	0
Fisher	15.56	**8.56**	79.56	**79.61**	45.15	**31.15**
Delete	4.15	**2.00**	94.76	**93.64**	0.41	**0.11**

MIA falls or stays at zero for all eight methods. L2U drops from 38.14 to 0, Fisher from 45.15 to 31.15. Random Label is the one method whose forgetting gets worse.

RESULT

CIFAR-100, twenty forget classes

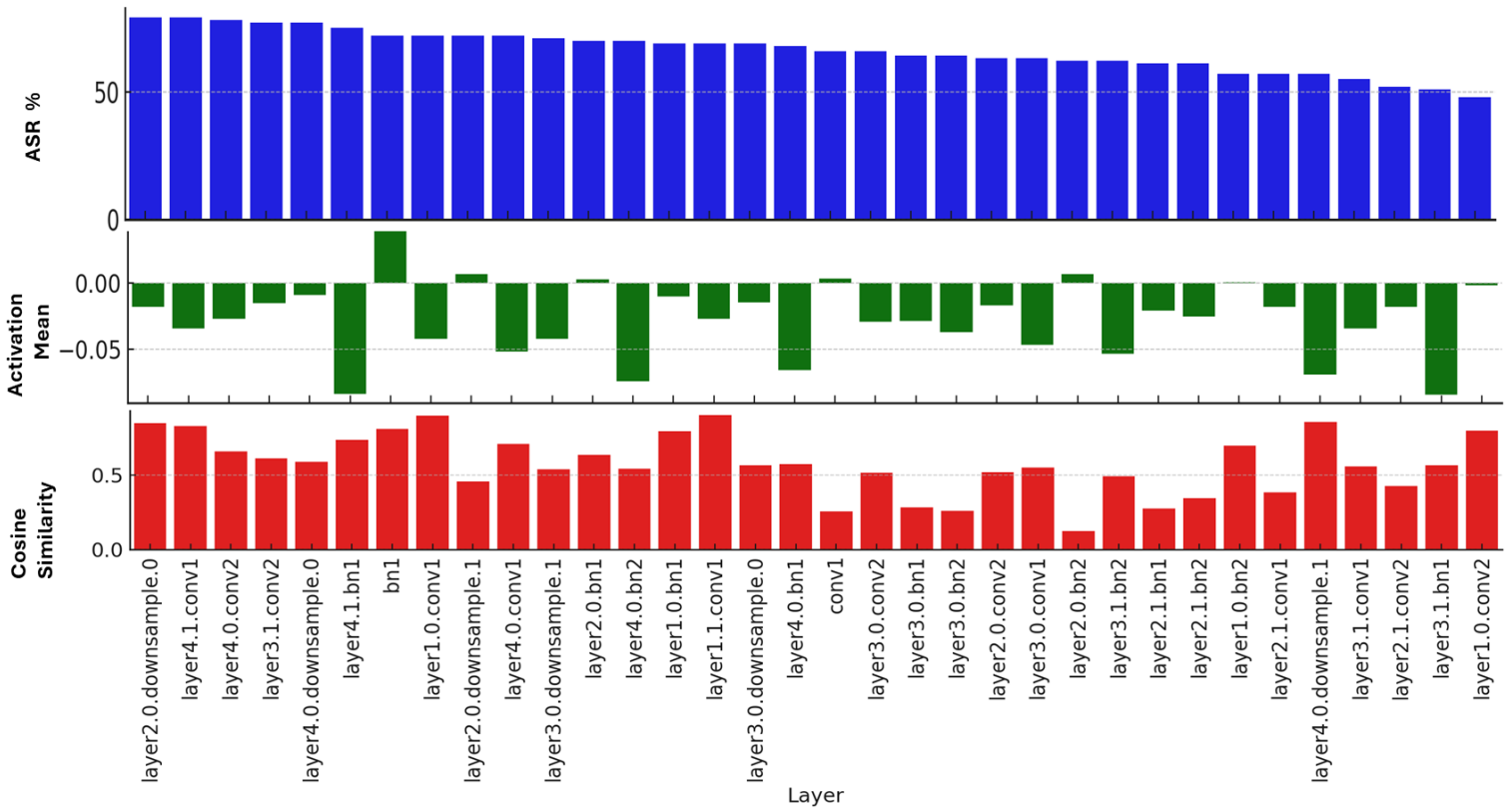
Same layout. Twenty classes removed at once, so this is the harder and more realistic setting.

Method	Acc_ft ↓ better		Acc_rt ↑ better		MIA ↓ better	
	base	+ probe	base	+ probe	base	+ probe
Original	77.97	71.41	79.01	79.51	100	94.54
Retrain	0	–	79.88	–	0	–
Finetune	0.77	**0**	79.97	**79.88**	0	0
GA	0	0	91.58	**91.77**	0	0
RL	0.52	**0**	60.84	**60.47**	0.06	**0**
L2U	20.14	**12.45**	62.41	**62.17**	37.54	**15.42**
BS	0.31	**0.24**	80.45	**81.24**	0	0
IU	3.21	**1.11**	67.57	**67.49**	0.37	**0.25**
Fisher	0.14	**0**	72.04	**71.94**	0.47	**0**
Delete	2.05	**1.48**	74.01	**74.45**	0.41	0.41

Forget-class accuracy improves for seven of eight methods and MIA never gets worse. Retention moves by less than a point either way, which is the point: the probe buys forgetting without paying for it in retention.

RESULT

Where the forget class actually lives



ResNet-18. Layers sorted by ASR. Middle panel is activation mean, bottom is cosine similarity.

What it shows

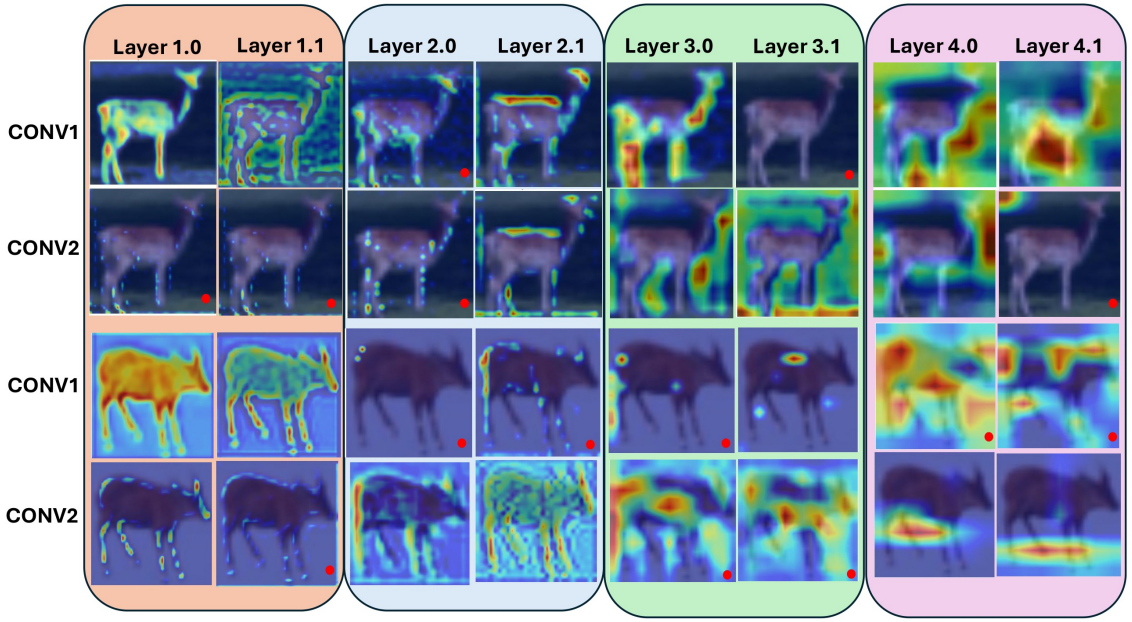
- A set of layers carries most of the forget signal.
- Most activation means are negative: these layers suppress the forget class rather than amplify it.
- Cosine similarity flags layer1.1.conv1 and layer4.0.downsample.1 as most aligned with forget features.

Answering RQ1: forgetting is not distributed evenly. ResNet spreads it through residual connections, VGG piles it into the final layers.

RESULT 5

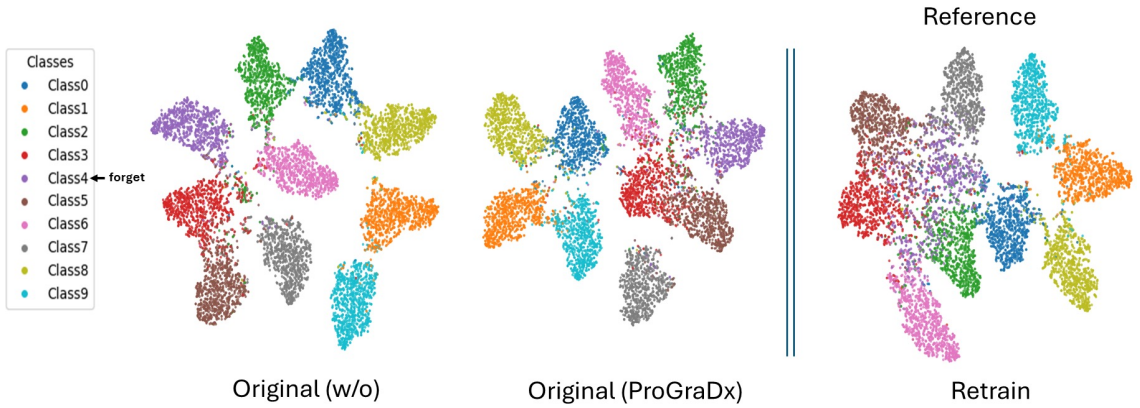
Two views of the same effect

Activation influence, ResNet-18



Red dots mark low-sensitivity layers: diffuse, weak responses, so they get clipped. Unmarked layers show tight object-aligned structure and get the noise.

Latent space, CIFAR-10 Class 4



Left: Class 4 forms a tight purple cluster. Middle: after ProGraDx it disperses. Right: Retrain, which never saw Class 4, is fully entangled. The other nine clusters stay intact throughout.

Answering RQ2: the disruption is targeted. Forget-class structure breaks down while retained-class structure survives.

Probe training and cost

Parameter	Value
Optimizer	Adam
Learning rate	0.001
Batch size	32
Max epochs	100 (converges in 20 to 30)
Early stopping	Patience 5 on validation loss
Loss	Cross-entropy
GAT layers	3, hidden dim 64, 2 attention heads each
Pooling	Global mean + global max, concatenated
MLP hidden dim	64
Output classes	2 (high / low sensitivity)
Graph connectivity	Fully connected

Cost

1.35 MFLOPs

126 seconds for the full probing pass, including activation extraction.

CONCLUSION

Diagnose then decide what to remove

- ProGraDx is the first diagnostic probing framework for machine unlearning.
- A GAT reads each layer's activations and gradients and scores its sensitivity to the forget set.
- The probe alone, on a frozen model, cuts forget-class accuracy by 16.6 points and membership inference from 99.4 to 90.5 on CIFAR-10.
- Added to eight baselines, it improves privacy across the board at 1.35 MFLOPs and no unlearning-time overhead is added.

Thank you

Questions welcome

avurity@gmu.edu

GEORGE MASON UNIVERSITY