Scaling Laws for SAE Training Data

Anonymous Author(s)
Affiliation
Address
email

Abstract

Sparse autoencoder (SAE) training is often bottlenecked by activation storage. We show that the stored activation buffer can be far smaller than the total SAE training budget: quality depends mainly on the number of *unique* activations, not on whether every training token is fresh. After a short diversity-limited regime, additional fresh activations provide little benefit; replaying the same buffer preserves both reconstruction and interpretability performance. We capture this diversity–repetition tradeoff with a data-constrained scaling law and validate it across dictionary sizes, token budgets, Llama-3.1 and Qwen3 models, early/middle/late layers, BatchTopK/TopK/JumpReLU SAEs, and downstream metrics including EV, reconstruction MSE, feature absorption, automated interpretability, and sparse probing. The resulting buffer-sizing rule is simple: choose the smallest activation buffer that reaches the quality plateau, then use replay to spend the remaining training budget. In our experiments, this reduces activation storage by 8–$64\times$ with negligible quality loss.

1 Introduction

Sparse autoencoders (SAEs) have become a standard tool for decomposing neural network representations into interpretable features [Cunningham et al., 2023, Gao et al., 2024, Lieberum et al., 2024]. As SAE training scales to larger language models and wider feature dictionaries, the storage cost of activation datasets becomes a practical bottleneck. State-of-the-art SAEs are trained on tens to hundreds of billions of model activations, and the most resource-intensive settings require materializing a large activation corpus on disk before training begins.

A natural question is whether this cost can be reduced without sacrificing quality. Prior work has not studied the distinction between the total number of SAE training tokens and the number of *unique* activations in the buffer: all training tokens are typically assumed equivalent, whether they represent fresh activations or repeated replays of an earlier activation. We show that these two quantities have qualitatively different effects on SAE quality.

We vary total training budget, dictionary size, and the fraction of activations that are unique. Holding total optimization tokens fixed, we sweep from full-unique training to many epochs over a small replay buffer, isolating activation diversity from optimization compute. Across these settings, SAE quality improves with the number of unique activations but quickly reaches a plateau: beyond a configuration-dependent threshold, additional fresh activations provide negligible benefit, so the same training budget can be spent by replaying a much smaller buffer.

We then fit a phenomenological scaling law adapted from the data-constrained framework of Muennighoff et al. [2023]. The law models SAE quality as a joint function of dictionary size and an *effective* activation count that discounts repeated examples. It describes the 96-run main grid with $\mathrm{RMSE} = 4.8 \times 10^{-3}$ EV, interpolates reliably across held-out training budgets, and extrapolates to held-out dictionary sizes when the held-out dictionary is sufficiently trained.

Submitted to 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Do not distribute.

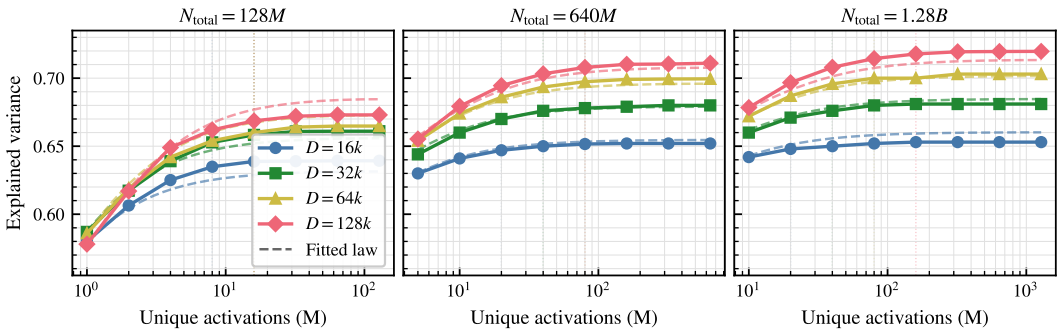

Figure 1: Explained variance as a function of unique activation count under fixed total SAE training budgets. Within each panel, total optimization tokens are fixed; moving right increases activation diversity while decreasing the number of epochs over the buffer. EV improves sharply in the low-diversity regime and then saturates.

We also test whether the diversity–repetition tradeoff is specific to the main BatchTopK setting. The same curve shape transfers across Llama-3.1 layers, Qwen3 model sizes, and TopK/JumpReLU variants after re-estimating only level parameters. SAEBench evaluations on the 640M layer-15 sweep show that reconstruction MSE, feature absorption, automated interpretability, and sparse probing saturate at thresholds comparable to EV. Together, these results support a practical buffer-sizing rule: choose the smallest replay buffer whose quality matches the full-buffer run, then spend the remaining training budget on replay. In our grid, this reduces activation storage by $\mathbf{8}$–$\mathbf{64}\times$ with negligible quality loss.

Contributions.

- We show that SAE quality saturates in the number of unique activations: at fixed total training budget, fresh activations help only up to a configuration-dependent plateau, after which replaying the same buffer is sufficient.
- We show that saturation appears across all evaluated quality metrics, including explained variance, reconstruction MSE, feature absorption, automated interpretability, and sparse probing.
- We validate the saturation pattern across dictionary sizes, token budgets, Llama-3.1 layers, Qwen3 model sizes, and BatchTopK/TopK/JumpReLU SAEs.
- We fit a closed-form scaling law for the diversity–repetition tradeoff, explaining the 96-run main grid with RMSE $= 4.8 \times 10^{-3}$ EV and favoring a dictionary-dependent repetition scale by AIC.
- We turn the plateau thresholds into a practical buffer-sizing rule: choose the smallest replay buffer that reaches the quality plateau, then spend the remaining training budget on replay. In our experiments, this reduces activation storage by $\mathbf{8}$–$\mathbf{64}\times$ with negligible reconstruction or interpretability loss.

2 Related Work

Sparse autoencoders for mechanistic interpretability. Cunningham et al. [2023] introduced sparse autoencoders as a method for decomposing language model residual streams into sparse, interpretable features. Subsequent work has scaled SAE training significantly: Gao et al. [2024] studied scaling laws for SAE with different sparsity, data budgets and layers, Templeton et al. [2024] scaled SAEs to large Anthropic models, and Lieberum et al. [2024] released Gemma Scope, a suite of open SAEs at every layer and width of Gemma 2. Automated interpretation of SAE features has been studied in Bills et al. [2023] and Paulo et al. [2024]. SAEBench [Karvonen et al., 2024] provides a comprehensive benchmark for evaluating SAE quality across interpretability-relevant downstream tasks. None of this work studies the distinction between unique activations and total training tokens.

Data-constrained and compute-optimal scaling laws. Scaling laws for language model training were systematized by Hoffmann et al. [2022], who derived compute-optimal model-size and

Table 1: Saturation intervals for each dictionary size and total training budget ($\epsilon = 0.005$).

Budget	Dict. size	Plateau EV	Saturation interval	Storage reduction
128M	16k	0.6393	8M–16M	8–16×
128M	32k	0.6610	16M–32M	4–8×
128M	64k	0.6648	16M–32M	4–8×
128M	128k	0.6730	16M–32M	4–8×
640M	16k	0.6520	20M–40M	16–32×
640M	32k	0.6800	40M–80M	8–16×
640M	64k	0.6995	80M–160M	4–8×
640M	128k	0.7110	80M–160M	4–8×
1.28B	16k	0.6530	20M–40M	32–64×
1.28B	32k	0.6810	40M–80M	16–32×
1.28B	64k	0.7030	80M–160M	8–16×
1.28B	128k	0.7197	160M–320M	4–8×

token-count tradeoffs (Chinchilla). Muennighoff et al. [2023] extended this framework to the data-constrained regime, where the text corpus is finite and data repetition is unavoidable. They introduced the concept of effective training tokens to capture diminishing returns from repeated data, and derived scaling laws under data constraints. Our work adapts their effective-token framework to SAE training, where the relevant constraint is the activation buffer rather than the text corpus.

3 Setup

Base models and activations. The main experiments use residual-stream activations from layer 15 of Llama-3.1 8B. We sample sequences from FineWeb [Penedo et al., 2024] and construct activation replay buffers at a fixed layer. Cross-layer validation additionally uses layers 5 and 25 of Llama-3.1 8B, and cross-model validation uses comparable mid-depth layers of Qwen3-1.7B, Qwen3-4B, and Qwen3-8B.

SAE architecture. The main grid trains BatchTopK sparse autoencoders [Gao et al., 2024] with $k = 32$ active features per token and dictionary sizes $D \in \{16\text{k}, 32\text{k}, 64\text{k}, 128\text{k}\}$. We test whether the observed saturation pattern transfers beyond this setting using standard TopK SAEs for cross-layer and cross-model validation, and BatchTopK/TopK/JumpReLU comparisons at $D = 32\text{k}$.

Diversity–repetition grid. We train with total token budgets $N_{\text{total}} \in \{128\text{M}, 640\text{M}, 1.28\text{B}\}$. For each budget, we vary the number of unique activations U by sweeping

$$U/N_{\text{total}} \in \{1/128, 1/64, 1/32, 1/16, 1/8, 1/4, 1/2, 1\}.$$

Thus each curve holds total optimization tokens fixed while moving from full-unique training ($U = N_{\text{total}}$) to 128 replay epochs over a small buffer ($U = N_{\text{total}}/128$). This isolates activation diversity from total optimization compute.

Evaluation. We evaluate each SAE on a held-out activation set (not included in any training buffer) and report explained variance (EV) as the primary quality metric. We define the saturation interval as the adjacent pair of unique-activation grid points bracketing the quality plateau: the lower endpoint is the smallest observed buffer size whose EV is within $\epsilon = 0.005$ of the maximum observed EV for that configuration, and the upper endpoint is the next larger grid point. Section 4.6 additionally evaluates downstream SAEBench metrics.

Training hyperparameters. Unless otherwise stated, all SAEs use the same optimizer, schedule, batch size, and activation normalization. The held-out EV set contains 10^6 FineWeb tokens with no overlap with any training buffer. Full training hyperparameters are given in Appendix A.

4 Results: The Diversity–Repetition Tradeoff

We first study how explained variance depends on the number of unique activations when the total SAE training budget is held fixed. For each total budget N_{total}, we vary the unique activation count

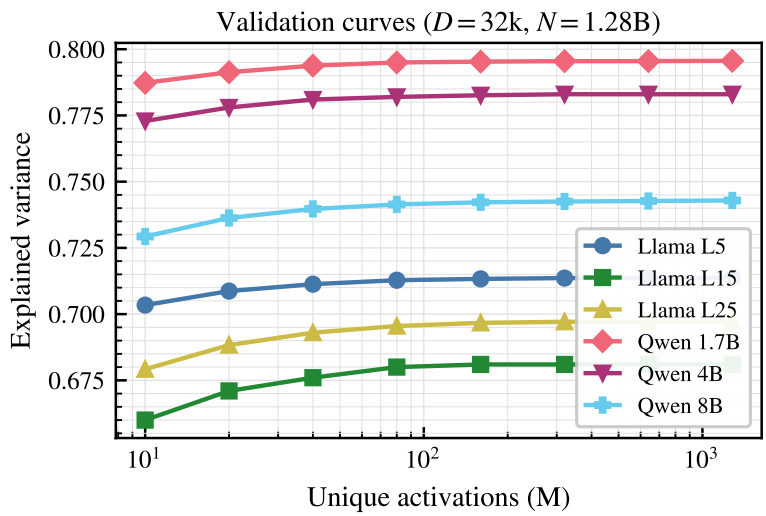

Figure 2: Cross-layer and cross-model validation ($D = 32$k, $N = 1.28$B). Llama-3.1 8B layer sweeps use BatchTopK SAEs; Qwen3 model-size sweeps use TopK SAEs. All curves show the same diversity-saturation pattern despite changes in layer, model size, and SAE activation function.

N_{unique} over a logarithmic grid and train for

$$E = \frac{N_{\text{total}}}{N_{\text{unique}}} \tag{1}$$

epochs. Thus, all points within a curve use the same number of total SAE optimization tokens, but differ in how much activation diversity is available.

4.1 Unique Activations Improve EV Under a Fixed Token Budget

Figure 1 shows explained variance as a function of unique activation count. At fixed total token budget, increasing the number of unique activations improves reconstruction quality substantially in the low-diversity regime. Equivalently, training for many epochs over a very small activation buffer underperforms training for fewer epochs over a larger and more diverse buffer.

For the 128M-token budget, the effect is large across all dictionary sizes. For example, increasing the unique activation count from 1M to the full 128M budget improves EV from 0.5801 to 0.6393 for the 16k dictionary, and from 0.5870 to 0.6610 for the 32k dictionary. The 64k dictionary shows the same pattern, improving from 0.5860 at 1M unique activations to 0.6648 at 128M unique activations.

The same qualitative behavior appears at the 1.28B-token budget. Although absolute EV values are higher, the low-diversity regime remains clearly suboptimal: repeated training over a small activation buffer does not fully substitute for broader activation coverage.

4.2 Explained Variance Saturates in Unique Activation Count

The benefit of additional unique activations is not linear. Across all dictionary sizes, EV increases rapidly at small buffer sizes but eventually reaches a plateau. This implies that full unique-token storage is often unnecessary: once the activation buffer is sufficiently diverse, additional unique activations provide negligible reconstruction benefit.

We define saturation as reaching EV within $\epsilon = 0.005$ of the maximum observed EV on the corresponding curve. Because the unique-activation grid is logarithmic, we report the saturation threshold as an interval between adjacent grid points: the smallest observed buffer size on the plateau and the next larger grid point.

Two patterns emerge from Table 1. First, at the largest budget the saturation interval scales approximately linearly with dictionary size: 20M–40M for 16k, 40M–80M for 32k, 80M–160M for 64k, and 160M–320M for 128k. Second, once the total budget comfortably exceeds the dictionary-specific

Table 2: Activation-function transfer at $D = 32\text{k}$, $N_{\text{total}} = 640\text{M}$, Llama-3.1 8B layer 15. BatchTopK and TopK are matched at $k = 32$; JumpReLU converges to $L_0 \approx 100$ and is therefore less sparse. All three variants saturate at 40M unique activations, showing that the diversity–repetition tradeoff is not specific to BatchTopK.

Activation	Sparsity	Plateau EV	u_{sat}	Storage reduction
BatchTopK	$k = 32$	0.6800	40M	16×
TopK	$k = 32$	0.6807	40M	16×
JumpReLU	$L_0 \approx 100$	0.7634	40M	16×

saturation interval, the interval depends only weakly on N_{total}. For example, the 16k, 32k, and 64k dictionaries have the same saturation intervals at 640M and 1.28B. The implied storage reduction therefore grows with total budget, reaching $\mathbf{64}\times$ for the 16k dictionary at 1.28B tokens.

4.3 Larger Dictionaries Reach Higher Plateaus but Require More Diversity

Dictionary size affects both the best observed EV and the saturation interval. At the 1.28B-token budget, doubling D raises the best observed EV by roughly 2–3 EV points and shifts the saturation interval upward by roughly $2\times$ (Table 1). At smaller budgets the plateau gain shrinks: the 128k dictionary at 640M reaches only 0.711 EV versus 0.720 at 1.28B, suggesting that 0.711 is an undertrained plateau rather than the dictionary's true asymptote. Larger dictionaries are therefore simultaneously more capable and more data-hungry, and a single-axis "total tokens" view of SAE training conflates these two effects. Equivalently, since $E = N_{\text{total}}/U$, the low-diversity regime in Figure 1 is a high-replay regime: training for many epochs over a small buffer underperforms training for fewer epochs over a larger, more diverse buffer.

4.4 Cross-Layer and Cross-Model Validation

We next test whether the diversity–repetition tradeoff persists beyond the main layer-15 Llama setup. These experiments are designed as validation checks rather than a second full scaling grid: all use $D = 32\text{k}$ and $N_{\text{total}} = 1.28\text{B}$ with the same unique-fraction protocol.

Layer generalization. To isolate the effect of layer within the same model and SAE architecture, we train BatchTopK SAEs on Llama-3.1 8B residual-stream activations from layers 5, 15, and 25. The plateau EV differs across layers (0.713 at layer 5, 0.681 at layer 15, and 0.697 at layer 25), reflecting layer-specific representation structure, but all three layers show the same pattern: low-diversity buffers underperform, and quality improves with unique activations before plateauing.

Model-size generalization. We also test whether the pattern depends on base-model size by training SAEs on Qwen3-1.7B, Qwen3-4B, and Qwen3-8B activations from comparable mid-depth layers. To keep this validation sweep lightweight, these Qwen runs use standard TopK rather than BatchTopK. This means the Qwen comparison changes both base model and SAE activation function, so it should be read as a combined stress test rather than an isolated model-family ablation. Plateau EV ranges from 0.743 for Qwen3-8B to 0.796 for Qwen3-1.7B, but all three model sizes exhibit the same diversity-saturation structure.

Together, these validation runs show that the saturation pattern is not specific to Llama layer 15. It appears across early, middle, and late Llama layers with BatchTopK, and across Qwen model sizes under TopK. Figure 2 shows the six validation curves.

4.5 Activation-Function Transfer

A natural concern is that the saturation phenomenon could be specific to BatchTopK. We therefore run a clean activation-function comparison on the main setup: Llama-3.1 8B layer 15, $D = 32\text{k}$, and $N_{\text{total}} = 640\text{M}$. We compare BatchTopK, standard TopK [Gao et al., 2024], and JumpReLU [Lieberum et al., 2024] over the same unique-activation grid, $u \in \{5, 10, 20, 40, 80, 160, 320, 640\}$M. BatchTopK and TopK both use $k = 32$ active features per token; JumpReLU uses a learned threshold and converges to an average $L_0 \approx 100$, making it less sparse and therefore not directly comparable in absolute EV.

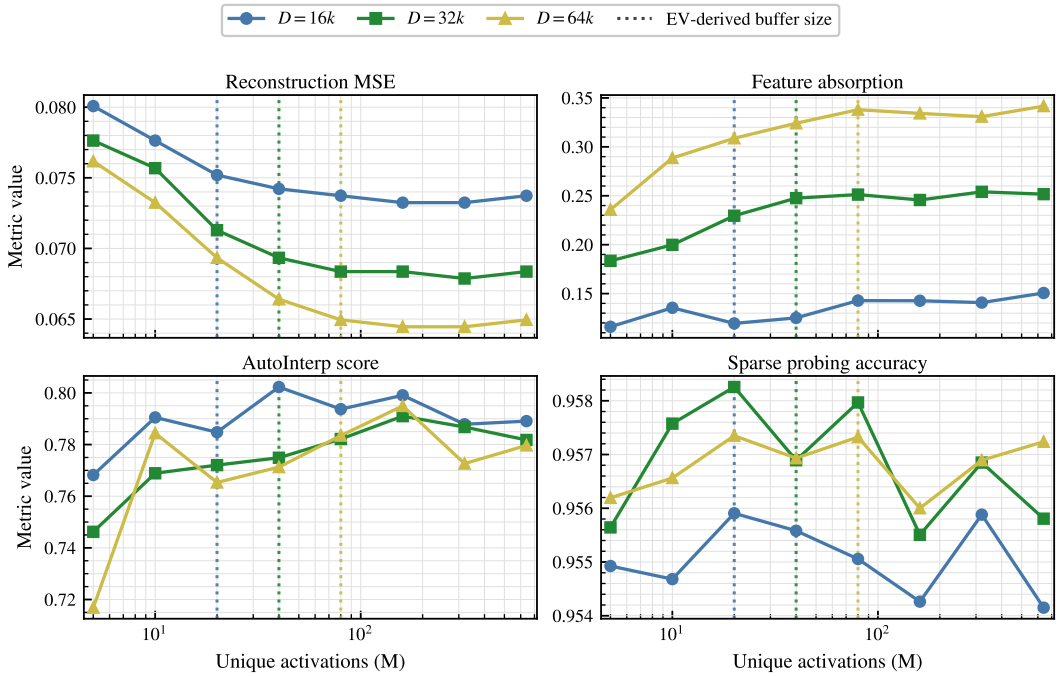

Figure 3: SAEBench metrics versus unique-activation count for the 640M-token, layer-15 BatchTopK sweep. Dotted vertical lines mark the EV-derived replay-buffer size for each dictionary. Reconstruction MSE closely follows the EV plateau; feature absorption and automated interpretability are noisier and fluctuate within a narrow range beyond the EV-derived buffer scale; sparse probing is largely insensitive to replay-buffer size. No evaluated downstream metric shows a collapse at the EV-derived buffer scale.

Saturation pattern. All three activation functions show the same qualitative behavior: EV improves rapidly with unique-activation count, then plateaus. At $D = 32$k, all three variants reach saturation at 40M unique activations (Table 2), despite different gating mechanisms and different plateau EV values. This suggests that the diversity–repetition tradeoff is a property of the training-data regime, not an artifact of BatchTopK.

Plateau levels. At matched $k = 32$, TopK reaches 0.681 EV, essentially identical to BatchTopK (0.680). JumpReLU reaches a higher plateau (0.763), but it also uses many more active latents per token ($L_0 \approx 100$), so the higher EV should not be interpreted as a pure gating-mechanism advantage. We therefore use this experiment to test transfer of the saturation pattern, not to rank activation functions by reconstruction quality.

4.6 Downstream Interpretability Tracks EV Saturation

The buffer-sizing rule is derived from explained variance, but SAE usefulness is not determined by EV alone. We therefore ask whether choosing the replay buffer from the EV saturation interval creates a hidden cost in downstream metrics. We evaluate 24 checkpoints from the 640M-token, layer-15 BatchTopK sweep on SAEBench: eight unique-activation settings for each of $D \in \{16\text{k}, 32\text{k}, 64\text{k}\}$. In Figure 3, dotted vertical lines mark the EV-derived replay-buffer size for each dictionary.

Comparison to the EV buffer scale. Reconstruction MSE follows the EV trend most closely: it improves rapidly in the low-diversity regime and is already near its best value at the EV-derived buffer size. Feature absorption generally improves with more unique activations, especially for larger dictionaries, but changes smoothly rather than exhibiting a sharp threshold. Automated interpretability is noisy, though its broad trend is consistent with better feature quality at larger buffers. Sparse-probing accuracy is nearly flat across the sweep, suggesting that this metric is largely insensitive to replay-buffer size in our setting.

Implications. The EV-derived buffer is not necessarily the point that maximizes every downstream metric on the grid. Some interpretability metrics fluctuate after the EV saturation scale rather than improving monotonically, so their apparent optima are less stable than the reconstruction plateau. Moreover, the absolute and relative changes beyond the EV-derived buffer are small. We therefore use SAEBench as a stress test rather than as a separate buffer-selection rule. Across the evaluated metrics, reducing the buffer to the EV saturation scale does not produce a hidden downstream collapse: reconstruction quality is preserved, feature absorption and automated interpretability remain in the same narrow performance band, and sparse probing is largely unaffected. This supports using EV saturation as a practical proxy for activation-buffer sizing.

4.7 Effective Activation Count

When a replay buffer contains U unique activations and is used for N_{total} training tokens, each activation is replayed on average N_{total}/U times. We define the repetition count as

$$R = \frac{N_{\text{total}}}{U} - 1. \tag{2}$$

Following Muennighoff et al. [2023], we convert (U, N_{total}) into an effective activation count

$$N_{\text{eff}}(U, N_{\text{total}}, R^*) = U\left[1 + R^*\left(1 - e^{-R/R^*}\right)\right], \tag{3}$$

where R^* controls how quickly repeated activations lose value. When $R \ll R^*$, replay is nearly as useful as fresh data and $N_{\text{eff}} \approx N_{\text{total}}$. When $R \gg R^*$, additional replay contributes little and N_{eff} grows only as a constant multiple of U.

4.8 Joint Scaling Law

We model explained variance as

$$\text{EV}(D, N_{\text{eff}}) = E_\infty - AD^{-\alpha} - BN_{\text{eff}}^{-\beta}. \tag{4}$$

The first penalty captures finite dictionary capacity: larger dictionaries reduce the gap to the infinite-dictionary limit. The second captures finite effective data: larger and less-repeated activation buffers reduce the data-limited error.

A constant repetition scale R^* is too restrictive, because larger dictionaries appear to become diversity-limited after fewer replay passes. We therefore also fit a dictionary-dependent scale

$$R^*(D) = cD^{-\gamma}. \tag{5}$$

This lets the value of replay decrease faster for wider dictionaries, matching the empirical observation that larger dictionaries require more activation diversity.

4.9 Fitting and Model Comparison

We fit on the 96-point main grid: three total budgets, four dictionary sizes, and eight unique-activation fractions for Llama-3.1 8B layer 15 BatchTopK SAEs. We compare the constant-R^* and dictionary-dependent-$R^*(D)$ models using AIC; fitting details are given in Appendix B.

The dictionary-dependent model fits substantially better: it reduces the fit error to $\text{RMSE} = 4.8 \times 10^{-3}$ EV and is strongly preferred by AIC, indicating that dictionary size and activation diversity interact rather than contributing independently. Full fitted parameters are reported in Appendix B.

Figure 4 (left) shows the law collapse: after removing the capacity term (plotting $\text{EV} + AD^{-\alpha}$ against $N_{\text{eff}}(D)$), all 96 data points fall onto a single power-law curve. The scatter plot (right) confirms agreement between predicted and observed EV at $\text{RMSE} = 4.8 \times 10^{-3}$.

4.10 Validation

Figure 4 summarizes the fit. After correcting EV by the fitted dictionary-capacity term, points from all dictionary sizes and budgets collapse onto a single curve as a function of N_{eff}, showing that the law captures the 96-run grid with small residual error.

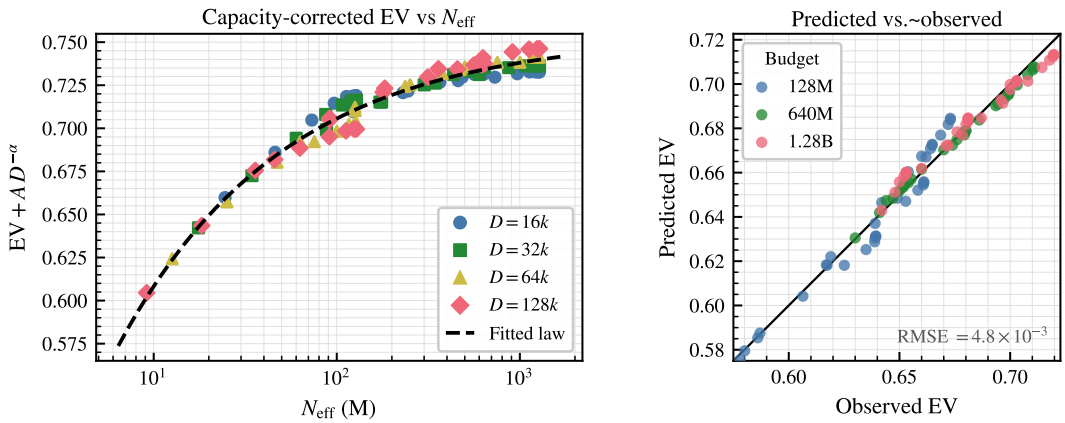

Figure 4: Capacity-corrected EV ($\text{EV} + AD^{-\alpha}$) versus effective activation count N_{eff} under the extended $R^*(D)$ model. Data points from all four dictionary sizes and three total budgets collapse onto a single power-law curve, confirming the functional form.

Held-out training budget. We fit the law on the two extreme budgets, 128M and 1.28B tokens, and predict the intermediate 640M budget. The interpolation is reliable: test RMSE is 3.1×10^{-3} EV, and the predicted saturation intervals match the observed intervals up to one grid step.

Held-out dictionary size. We also fit on the three smaller dictionaries and predict the held-out 128k dictionary. Extrapolation succeeds at 1.28B tokens, where the 128k dictionary is sufficiently trained, but degrades at smaller budgets. This is expected: when a large dictionary is undertrained, the observed best EV is not its true asymptotic plateau, so a law fit on well-trained smaller dictionaries cannot infer that undertraining regime.

Shape transfer. Finally, we test whether the fitted curve shape transfers beyond the main grid. Holding the nonlinear shape fixed, we re-estimate level parameters for the cross-layer and cross-model validation curves. The resulting fits track Llama layer variation and Qwen model-size variation well, suggesting that the diversity–repetition curve shape is reusable even when the absolute EV level changes.

5 Practical Implications: Activation Buffer Sizing

The saturation intervals give a direct recipe for choosing activation-buffer sizes. Instead of storing one fresh activation for every training token, practitioners can store a replay buffer just large enough to reach the quality plateau, then spend the remaining training budget by replaying that buffer.

Concretely, for a target dictionary size D and total SAE training budget N_{total}, choose a replay buffer size U_{buf} from the corresponding saturation interval. In Table 5, we use the lower endpoint of each interval: the smallest observed buffer size whose EV is within $\epsilon = 0.005$ of the best point on that curve. The number of replay epochs is then

$$E = \frac{N_{\text{total}}}{U_{\text{buf}}}, \tag{6}$$

and the storage reduction relative to storing all activations is

$$\text{StorageReduction} = \frac{N_{\text{total}}}{U_{\text{buf}}}. \tag{7}$$

Using this rule, every configuration in our grid admits at least an $8\times$ reduction in activation storage, and the largest-budget smaller-dictionary settings reach up to $64\times$ (Appendix D). In absolute terms, this turns multi-terabyte activation corpora into replay buffers of a few hundred gigabytes or less. For example, a Llama-3.1 8B residual stream stored in bf16 requires 8 KiB per activation; at $N_{\text{total}} = 1.28\text{B}$ and $D = 16\text{k}$, full-buffer storage would require 9.5 TB, while the recommended replay buffer requires only 153 GB.

The reduction grows with total budget once the budget exceeds the dictionary's activation-diversity requirement. For smaller dictionaries, the saturation interval is reached early, so increasing N_{total} mostly increases replay and therefore storage savings. For larger dictionaries at small budgets, the model can remain capacity- or optimization-limited, so the savings grow more slowly until the total budget is large enough to reveal the dictionary's full plateau.

Training recipe. In practice, choose the target dictionary size and total budget, estimate the saturation interval from a pilot sweep or the fitted law, set U_{buf} to the smallest buffer on the quality plateau, and train for $N_{\text{total}}/U_{\text{buf}}$ replay epochs.

6 Conclusion

SAE training quality is governed not only by the total number of optimization tokens, but by the number of unique activations those tokens cover. At fixed training budget, small replay buffers underperform diverse ones, but the benefit of fresh activations quickly saturates. Once this plateau is reached, replaying a much smaller activation buffer preserves reconstruction quality and does not produce a hidden collapse in the downstream SAEBench metrics we evaluate.

We model this diversity–repetition tradeoff with a data-constrained scaling law that combines dictionary size, total training budget, and effective activation count. The law fits the 96-run main grid, interpolates held-out budgets, and captures the same curve shape across Llama layers, Qwen model sizes, and SAE activation variants. Translating the observed saturation intervals into buffer-sizing recommendations yields 8–$64\times$ activation-storage reductions in our experiments.

The practical message is simple: SAE training does not require storing a fresh activation for every optimization token. Practitioners can choose the smallest replay buffer that reaches the quality plateau, then spend the remaining budget on replay or recomputation. This turns activation storage from a scaling bottleneck into a tunable resource, and makes larger SAE sweeps substantially cheaper to run.

Limitations. Our main grid uses one text domain and one base architecture, Llama-3.1 8B; cross-layer and cross-model experiments support shape transfer but do not establish universal exponents. Held-out validation shows that the law interpolates across budgets more reliably than it extrapolates to undertrained larger dictionaries. Saturation intervals are reported at a fixed tolerance $\epsilon = 0.005$ EV, so stricter quality targets may require larger buffers. Activation-function transfer is tested at a single dictionary size, and SAEBench evaluations are limited to the layer-15 BatchTopK sweep. Finally, while storing a replay buffer and recomputing the same activations are equivalent training schedules, we do not evaluate all possible online sampling policies.

Future work. Extending the sweep to more base models, domains, modalities, and SAE objectives would test how universal the fitted exponents are. It would also be useful to study adaptive buffer construction: choosing the replay set online, refreshing it selectively, or allocating activation diversity unevenly across layers and training stages.

References

Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. Language models can explain neurons in language models. *OpenAI blog*, 2023.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*, 2023.

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Tow, Hoagy Cunningham, Tom Conerly, Adly Templeton, Trenton Bricken, et al. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093*, 2024.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Adam Karvonen, Joseph Bloom, Arthur Conmy, Samuel Marks, Can Rager, Curt Tigges, Nikhil Vyas, Jett Wang, et al. SAEBench: A comprehensive benchmark for sparse autoencoder evaluation. *arXiv preprint arXiv:2412.06881*, 2024.

Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, János Kramár, Rohin Shah, Tom Henighan, Neel Nanda, and Juyeon Bhattacharyya. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2. *arXiv preprint arXiv:2408.05147*, 2024.

Niklas Muennighoff, Alexander M. Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A. Raffel. Scaling data-constrained language models. *arXiv preprint arXiv:2305.16264*, 2023.

Gonçalo Paulo, Alex Mallen, Caden Juang, and Nora Belrose. Automatically interpreting millions of features in large language models. *arXiv preprint arXiv:2410.13928*, 2024.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The fineweb datasets: Decanting the web for the finest text data at scale. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL `https://openreview.net/forum?id=n6SCkn2QaG`.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, et al. Scaling and evaluating sparse autoencoders. *Anthropic, accessed*, 2024.

A Additional Experimental Details

Training hyperparameters. Unless otherwise stated, SAEs are trained with AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.99$, weight decay 0.05), peak learning rate 3×10^{-4}, cosine decay to 10^{-6}, and a 10% linear warmup. Batch size is 4096 activations and gradients are clipped to unit norm. For TopK-style models, k is linearly annealed from 128 to the target value over the first 10% of training. BatchTopK runs use an auxiliary dead-latent penalty of $1/32$ and straight-through estimator bandwidth 0.001. All activations are unit-normalized before being passed to the SAE.

B Scaling-Law Fitting Details

For fixed nonlinear parameters, the law in Eq. 4 is linear in (E_∞, A, B). We therefore search over the nonlinear parameters and solve for the linear parameters by ordinary least squares at each candidate point, then refine the best solution with local coordinate descent. For the constant-R^* baseline, the nonlinear parameters are (α, β, R^*); for the dictionary-dependent model they are $(\alpha, \beta, c, \gamma)$. We fit both models on the 96-point main grid and compare them using the Akaike Information Criterion (AIC).

Table 3: Fitted parameters for the constant-R^* and extended $R^*(D)$ models, fit on all 96 data points from the main grid. The extended model is decisively preferred by AIC.

Model	E_∞	A	α	B	β	R^* / c	γ	ΔAIC
Const. R^*	0.738	0.387	0.688	0.432	0.466	13.7	—	0
Extended $R^*(D)$	0.760	0.328	0.486	0.440	0.477	103.3	0.526	-77.3

C SAEBench Threshold Table

Table 4 reports per-metric saturation thresholds for the SAEBench sweep. For each metric, we report the smallest unique-activation count within 5% of that metric's observed range from the best value on the corresponding curve. These thresholds are descriptive only; the main buffer-sizing rule uses the EV-derived saturation intervals.

Table 4: SAEBench saturation thresholds at $N_{\text{total}} = 640\text{M}$, Llama-3.1 8B layer 15, BatchTopK. The EV row is reproduced from Table 1.

Quantity	$D = 16\text{k}$	$D = 32\text{k}$	$D = 64\text{k}$
Explained variance (main)	20M	40M	80M
Reconstruction MSE	80M	80M	80M
Feature-absorption fraction	80M	40M	80M
Automated interpretability score	40M	160M	160M
SAE sparse-probing accuracy	20M	20M	20M

D Storage-Reduction Recommendations

Table 5: Recommended activation-buffer sizes and implied storage reductions ($\epsilon = 0.005$). Epochs $= N_{\text{total}}/u_{\text{sat}}$.

Budget	Dict. size	u_{sat}	Epochs	Storage reduction
128M	16k	8M	16	16×
128M	32k	16M	8	8×
128M	64k	16M	8	8×
128M	128k	16M	8	8×
640M	16k	20M	32	32×
640M	32k	40M	16	16×
640M	64k	80M	8	8×
640M	128k	80M	8	8×
1.28B	16k	20M	64	64×
1.28B	32k	40M	32	32×
1.28B	64k	80M	16	16×
1.28B	128k	160M	8	8×

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the central empirical finding, validation scope, scaling-law contribution, and storage-saving implication. The limitations section further qualifies the generality of the claims.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper includes a limitations paragraph discussing the single main architecture and text domain, extrapolation limits, fixed tolerance choice, limited activation-function sweep, and limited SAEBench scope.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.

- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper proposes and fits a phenomenological scaling law but does not present formal theoretical results, theorems, or proofs.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The setup, results, and appendices specify the models, data source, layers, activation grid, training budgets, architectures, evaluation metrics, fitting procedure, and hyperparameters needed to reproduce the main experimental design.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed

instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: The paper provides experimental and plotting details but does not yet provide a public anonymized code and data release as part of the manuscript.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: The setup section and Appendix A describe the data source, held-out evaluation set, model layers, SAE architectures, token budgets, unique-activation grid, optimizer, schedule, batch size, and normalization.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We do not report error bars or repeated-seed confidence intervals, because the main grid consists of large single training runs and repeating the full sweep would be computationally expensive. We instead report dense sweeps over dictionary size, token budget, and activation diversity, plus held-out and transfer validations.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: The paper describes the training scale and activation-storage requirements, but does not provide a complete accounting of GPU type, memory, wall-clock time, or total compute for each experiment.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The work is an empirical study of SAE training efficiency using public models and public text data, and we are not aware of any deviation from the NeurIPS Code of Ethics.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: The work is foundational research on reducing activation-storage costs for sparse autoencoder training and does not introduce a deployed system or direct application pathway. Its main expected impact is improved accessibility of interpretability experiments through lower storage costs.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release a new high-risk pretrained model, image generator, or scraped dataset requiring special safeguards.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [No]

    Justification: The paper cites the main existing assets used, including base models, FineWeb, SAEBench, and prior SAE methods, but it does not explicitly enumerate all asset licenses and terms of use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not introduce or release a new dataset, benchmark, or public model asset; it reports experimental results from SAE training runs.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or research with human subjects, so IRB approval is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not an important, original, or non-standard component of the research methodology; any use for writing, editing, or formatting does not affect the scientific method or results.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.