TopK Sparse Autoencoders Across Three Model Architectures: Dictionary Collapse, Dense-Feature Degeneracy, and the Limits of Activation-Pattern Feature Matching

J. Melton
American Code Labs
jmelton@americancode.org

July 2026
Preprint.

Abstract

We train TopK sparse autoencoders (SAEs) at matched sparsity ($k = 128$) and matched dictionary size (16,384) on mid-network residual stream activations from three model families—Llama-3.2-3B, Mistral-7B-v0.3 and Qwen2.5-3B—and report three findings, two of them negative.

First, TopK training without an auxiliary revival loss undergoes an abrupt *dictionary collapse* followed by partial recovery. In all three runs, dead features rise by 3.9× to 41× shortly after step 2,000, peaking between steps 5,200 and 5,800—a 600-step band shared across three source models with pre-collapse dead counts differing tenfold. Between 16% and 57% of collapsed features subsequently return over the remaining 45,000 steps. Reconstruction quality gives no warning: two models show a transient FVE dip that fully recovers, and Mistral's FVE *rises* through its collapse while three-quarters of its dictionary goes silent. Qwen ends at 57.4% dead.

Second, the surviving code is dominated by a small set of dense directions. With $k/D = 0.78\%$, a feature should fire on under 1% of tokens if activation were spread evenly; instead 16–25 features per model fire on more than half of all tokens, and the 200 most frequent features absorb 29–39% of total activation mass. Any procedure that selects features by activation frequency—a common first step in feature labelling and in feature-based classifiers—selects almost entirely from this degenerate head.

Third, cross-architecture feature matching by activation-pattern similarity finds essentially nothing. Under a chunk-permutation null, the best-of-16,384 match similarity between *unrelated* features averages 0.59–0.74 and has a 99th percentile of 0.94–0.98, so the conventional fixed threshold of 0.8 sits below the noise floor for every model pair. With one-to-one reciprocal matching, a closed-triangle requirement and a null-calibrated threshold, exactly one feature triple is shared across all three models—against a null that yields at least one 14% of the time. The conventional protocol applied to the same fingerprints yields 3,753. A planted-signal power analysis bounds what that null means: minimum detectable effect size tracks the calibrated threshold, reaching a correlation of 0.95–1.00, so the finding is "no triple shared at correlation ≥ 0.95" rather than "no shared structure". A positive control fixes the other bound. Two SAEs differing only in training seed—same model, same activations, same hyperparameters—agree on just 8% of their matchable features, which the procedure recovers at 108× the null. The method is therefore not blind, but 8% is its ceiling, and the three runs are indistinguishable by

loss, FVE, L_0 and dead-feature count while disagreeing about 92% of their dictionaries. Because the threshold must clear a null whose mean grows with dictionary size, that floor is structural to dictionary-scale matching rather than a consequence of our choices.

Results characterise the TopK objective at a single sparsity level; no comparison of training objectives is made, as only TopK autoencoders were trained (§5.1).

1 Introduction

Language models represent concepts in high-dimensional activation spaces that mix thousands of distinct semantic and syntactic features within individual neurons—the phenomenon Elhage et al. [5] term *superposition*. Sparse autoencoders address this by learning an overcomplete dictionary of directions such that any activation can be reconstructed as a sparse sum over dictionary elements. Bricken et al. [3] applied the approach at scale and found the resulting features highly interpretable.

The practical literature has converged on a small number of quality metrics—variance explained, L_0, dead-feature count—and on reporting them at the end of training. This paper is about what those metrics miss. We train TopK SAEs [6] under matched hyperparameters on three model families and instrument the training process rather than only its endpoint. Two pathologies emerge that a final-checkpoint report would not surface, and a third result—an attempt to match features across the three models—fails in a way that we think is instructive.

Why these three models. Llama-3.2-3B, Mistral-7B-v0.3 and Qwen2.5-3B come from three organizations, have different hidden dimensions (3,072 / 4,096 / 2,048), different attention schemes, and no publicly shared training data. Holding dictionary size fixed at 16,384 across them means the expansion ratio varies (5.33× / 4.0× / 8.0×). The model with the largest ratio ends with the worst dead-feature rate, though as §4.1 notes, ratio and source model are confounded across only three runs.

Contributions.

1. **Dictionary collapse is abrupt, consistently timed, partially reversible, and invisible to FVE.** We characterize a sharp transition peaking between steps 5,200 and 5,800 in all three models, in which dead features rise 3.9×–41× before 16–57% of them recover, while reconstruction quality gives no signal (§4.1).

2. **Quantification of dense-feature degeneracy.** We measure how much of the sparse code's activation mass is carried by its most frequent features, and show the concentration is severe enough to invalidate frequency-based feature selection (§4.2).

3. **A permutation null for cross-model feature matching, its detection floor, and a negative result.** We show that the standard protocol—fixed cosine threshold, nearest-neighbour matching—produces thousands of spurious matches, and that essentially nothing survives proper calibration (§4.4). We then measure what the calibrated procedure could have found: minimum detectable effect size equals the threshold, 0.95–1.00, so the method sees near-duplicates and little else (§4.6). Reporting that floor should be standard practice for cross-model matching claims in either direction.

4. **A positive control for the matching method, and a ceiling.** Two SAEs differing only in seed agree on 8% of matchable features at 108× the null (§4.5), so the near-zero cross-model count is not instrument failure—but no study of this kind can report more than about 8%, and standard quality metrics cannot distinguish dictionaries that disagree about 92% of their content.

5. **An honest account of what these numbers do not establish.** With all three SAEs trained over 205 epochs of their entire activation dump, no held-out reconstruction measurement is available for any of them. We report the one held-out number we do have—from a partial 1,000-step checkpoint—and explain why it is weak evidence (§4.3).

2 Related Work

2.1 Superposition and Dictionary Learning

Elhage et al. [5] established the theoretical basis for SAE-based feature extraction: a transformer with n neurons can represent $O(n^2)$ nearly orthogonal feature directions if those features are sparse. Dictionary learning recovers them by sparse factorization: given activations $X \in \mathbb{R}^{n \times d}$, find $D \in \mathbb{R}^{m \times d}$ (with $m \gg d$) and sparse $C \in \mathbb{R}^{n \times m}$ such that $X \approx CD$ and $\|c_i\|_0 \ll m$ for each row.

2.2 SAE Architecture Variants

Standard L1 SAE. Bricken et al. [3], Cunningham et al. [4] train an encoder–decoder pair to minimize

$$\mathcal{L} = \|x - \hat{x}\|_2^2 + \lambda \|z\|_1, \tag{1}$$

where $\hat{x} = W_{\text{dec}}\,\text{ReLU}(W_{\text{enc}}x + b_{\text{enc}}) + b_{\text{dec}}$. The L1 penalty encourages sparsity through a continuous relaxation.

TopK SAE. Gao et al. [6] replace the L1 penalty with a hard constraint:

$$z^{\text{TopK}} = \text{TopK}(\text{ReLU}(W_{\text{enc}}x + b_{\text{enc}})), \tag{2}$$

guaranteeing exactly k nonzero activations per input. Critically for this paper, Gao et al. [6] pair TopK with an *auxiliary loss* that periodically resurrects dead feature directions. Our implementation omits it, and §4.1 is in effect a measurement of what that omission costs.

Gated and JumpReLU SAEs. Rajamanoharan et al. [8] separate feature detection from magnitude estimation via a gating network; Rajamanoharan et al. [9] use a discontinuous activation with a learned threshold. Neither variant is trained here (§5.1).

2.3 Evaluation Methodologies

Prior evaluation approaches include linear probing [1, 2, 4], human interpretability ratings via the top-k activating contexts protocol [3], and activation steering fidelity [10, 11]. This paper uses none of them; our evaluation is confined to reconstruction quality, dictionary utilization, and cross-model feature correspondence.

Table 1: The three training runs, all completed to the same specification. "Epochs" is the number of passes over the source token set implied by steps × batch.

Model	Layer	d	Expansion	Source tokens	Steps	Epochs
Llama-3.2-3B	14	3,072	5.33×	500,000	50,000	205
Mistral-7B‡	16	4,096	4.0×	500,000	50,000	205
Qwen2.5-3B	18	2,048	8.0×	500,000	50,000	205

‡ 4-bit quantized source model.

3 Methods

3.1 Models, Layers, and Activation Collection

SAEs are trained on residual stream activations at the 50% depth midpoint of each model: **Llama-3.2-3B** (layer 14 of 28, $d = 3{,}072$), **Mistral-7B-v0.3** (layer 16 of 32, $d = 4{,}096$), and **Qwen2.5-3B** (layer 18 of 36, $d = 2{,}048$). Activations are collected from 500,000 tokens of WikiText-103 (`wikitext-103-raw-v1`, train split), streamed in the same order for every model, and stored as float16.

Mistral activations were extracted from 4-bit quantized weights (`mlx-community/Mistral-7B-v0.3-4bit`); the bf16 base model requires authentication we did not have. This is a confound we cannot rule out and flag wherever Mistral results appear.

3.2 SAE Configuration

All three SAEs are TopK with $k = 128$, dictionary size $D = 16{,}384$, peak learning rate 10^{-4} with cosine annealing, batch size 2,048 tokens, and seed 42. Because D is held fixed while d varies, the expansion ratio differs by model: 5.33× (Llama), 4.0× (Mistral), 8.0× (Qwen). The encoder is $z = \mathrm{TopK}(\mathrm{ReLU}(W_{\mathrm{enc}}(x - b_{\mathrm{dec}}) + b_{\mathrm{enc}}))$ and the decoder $\hat{x} = W_{\mathrm{dec}} z + b_{\mathrm{dec}}$, with decoder columns unit-normalized. The loss is plain MSE; there is no auxiliary dead-feature loss and no re-initialization scheme.

All three runs completed the same 50,000 steps over the same 500,000-token corpus (Table 1), so training length is not a confound for any cross-model comparison below. The remaining asymmetry is expansion ratio, which varies because dictionary size is held fixed while hidden dimension does not.

3.3 Reconstruction Metrics

We report fraction of variance explained, $\mathrm{FVE} = 1 - \mathrm{Var}(x - \hat{x}) / \mathrm{Var}(x)$, and MSE. MSE is activation-scale-dependent and is not comparable across models with different hidden dimensions and activation norms; FVE is.

An important measurement detail: during training, FVE is computed on the current batch. This is a noisy estimator, and reporting the final-step value—the natural thing to do—samples once from a wide distribution. Across the last ten logged steps the batch FVE spans 0.013 (Llama), 0.021 (Qwen) and 0.364 (Mistral). All FVE figures in §4 are instead computed over the full 500,000-token corpus.

Table 2: Dead-feature count (of 16,384) over training, from the rolling 5,000-step `dead_5k` window. All three models collapse within a 600-step band and then partially recover. Data: `data/sae-runs/*/training.jsonl`.

Model	Pre-collapse	Peak	at step	Final (50k)	Collapse	Recovered
Llama-3.2-3B	1,078	8,671	5,400	3,753	8.0×	57%
Qwen2.5-3B	2,893	11,242	5,200	9,401	3.9×	16%
Mistral-7B	292	12,082	5,800	5,272	41.4×	56%

3.4 Cross-Architecture Feature Matching

Features are compared across models by *functional* similarity: the correspondence of their activation patterns over a shared corpus. For each model we encode 500,000 tokens of WikiText-103, partition them into $N = 1{,}000$ non-overlapping chunks of 500 tokens, and average post-TopK feature activations within each chunk. Column j of the resulting $N \times D$ matrix is a 1,000-dimensional *fingerprint* of feature j. Fingerprints are centred before normalization, so the similarity statistic is a Pearson correlation across chunks rather than a cosine between non-negative vectors.

Three procedural choices distinguish this from the conventional protocol, and §4.4 shows each is necessary:

1. **Support filter.** Features active in fewer than 5% of chunks are excluded. A feature active in a handful of chunks correlates near 1 with any other feature active in the same chunks, by construction rather than by shared function.

2. **Reciprocal matching.** A match requires features i and j to be *mutual* nearest neighbours. This is one-to-one by construction. Taking the union of both directions' nearest neighbours—the common alternative—permits one feature to be the recorded partner of arbitrarily many, and permits match counts exceeding the dictionary size.

3. **Null-calibrated threshold.** τ is set to the 99th percentile of a permutation null in which chunk order is shuffled in one model, destroying cross-model correspondence while preserving each feature's marginal activation profile (20 replicates).

A feature triple is *three-way universal* only if all three pairwise edges independently satisfy these criteria—a closed triangle, not a chain through an intermediate model.

4 Results

4.1 Dictionary Collapse and Partial Recovery

All three runs undergo the same three-phase trajectory in dictionary utilization: a slow decline in dead features through roughly step 2,000, an abrupt collapse peaking between steps 5,200 and 5,800, and a long partial recovery through the remainder of training. Table 2 gives the shape.

The collapse is abrupt and its timing is consistent. Dead features are flat or slowly declining through step 2,000 in every run, then rise by between 3.9× and 41× within about 3,000 steps. The peak falls between steps 5,200 and 5,800 in all three models, despite pre-collapse dead counts differing by a factor of ten (292 to 2,893) and despite three different source models, hidden dimensions and expansion ratios. Whatever drives the transition is a property of the training procedure rather than of the model being decomposed.

Recovery is real, substantial, and slow. Collapse is not a ratchet, and the argument that it must be does not survive the data. That argument runs: a feature outside the top-k on every batch receives exactly zero encoder gradient, so it can never return. Its first half holds—W_{enc}'s row for a silent feature does stop updating—but the conclusion does not follow, because the encoder input is $x - b_{\text{dec}}$, and b_{dec} continues to receive gradient from the features that *are* firing. A silent feature's pre-activation therefore keeps moving even though its own weights do not, and it can drift back into contention. Empirically it does, for 16% to 57% of collapsed features, and the recovery is close to monotone—after the peak, Llama shows 30 increases across 224 logged points and Mistral 18 across 222.

The practical consequence is that a dead-feature count read at any single step between roughly 4,000 and 15,000 badly overstates permanent loss. Read the Llama run at step 13,000 and it records 5,278 dead; read it at convergence and the figure is 3,753. A single-checkpoint dead count is not a property of the trained SAE unless the checkpoint is the final one.

Reconstruction quality does not warn you, and for one model it improves through the collapse. Llama and Qwen both show an FVE dip at step 5,000 (0.986 to 0.775, and 0.982 to 0.841), but both are back above 0.98 by step 10,000 and stay there. Mistral shows no dip at all: its FVE *rises* from 0.933 at step 2,000 to 0.966 at step 5,000 while its dead count goes from 292 to 3,500 on its way to 12,082. A practitioner watching reconstruction quality on that run would have seen a healthy, monotonically improving curve while three-quarters of the dictionary went silent. Dictionary utilization has to be logged separately; it is not recoverable from the loss.

Expansion ratio does not straightforwardly predict severity. Qwen has the largest expansion ratio (8.0×) and the worst final dead rate (57.4%), which is the expected direction. But Mistral has the *smallest* ratio (4.0×) and the largest collapse magnitude (41×), and recovers to a better final state than Qwen. With three runs at three different ratios and no replicates, we can report the ordering but cannot attribute it: source model, activation norm distribution and expansion ratio are all confounded here.

4.2 Dense-Feature Degeneracy

Sparsity of the code does not imply that the code is evenly used. With $k = 128$ of $D = 16{,}384$, a feature would fire on 0.78% of tokens if activation were spread uniformly. Table 3 shows the actual distribution.

In every model, 1.2% of the dictionary carries between 29% and 39% of the sparse code, and 16 to 25 features fire on the majority of all tokens. A feature firing on 97% of tokens—the maximum we observe in both Llama and Qwen—is not a feature in any useful sense; it

Table 3: Concentration of activation mass, measured over the full 500,000-token corpus. "Top-200 mass" is the share of all feature activations accounted for by the 200 most frequently firing features—1.2% of the dictionary. Data: `data/sae-analysis/corrections.json`.

Model	Uniform expectation	Features firing >50% of tokens	Top-200 mass
Llama-3.2-3B	0.78%	25	30.9%
Qwen2.5-3B	0.78%	20	38.6%
Mistral-7B	0.78%	16	29.0%

is closer to a learned bias absorbed into the dictionary.

Training to convergence dilutes this but does not remove it. At 1,000 steps the top-200 share is 40% (Llama) and 57% (Mistral); at 50,000 steps it is 31% and 29%. The concentration falls as features specialise, and then stops falling: a third of the code still sits in 1.2% of the dictionary at convergence.

Consequence for feature selection. The standard first step in feature labelling is to take the top-N features by activation frequency and inspect their maximally activating contexts. Table 3 says that this procedure draws heavily from the degenerate head: of the 200 most frequent Llama features, 25 fire on more than half of all tokens, and those 200 features carry 31% of the entire code. In a companion paper [7] we built a feature-based safety classifier on exactly this selection and obtained ROC-AUC 0.564 with a 95% confidence interval spanning chance. Switching to differential-frequency labelling against a harm-relevant contrastive corpus (PKU-SafeRLHF + do-not-answer + toxic-chat) raised ROC-AUC to 0.6422 (95% CI [0.566, 0.720])—still modest but now clearly above chance, confirming that the degenerate feature head is the primary culprit rather than TopK SAEs in general. Selecting by *differential* activation between contrastive corpora, and imposing a density ceiling, are both necessary; neither is standard practice.

4.3 Held-Out Reconstruction

All three SAEs made 205 passes over their entire 500,000-token activation dump. There is no unseen data for any of them, and at that level of repetition an SAE can in principle memorize the corpus rather than learn a dictionary. Every FVE in the top half of Table 4 is in-sample, and we report them as the best available characterization, not as evidence of generalization.

One genuine held-out measurement exists, and only by accident. A 1,000-step Mistral checkpoint was trained on just the first 50,000 tokens of the dump, leaving 450,000 unseen. It reconstructs held-out activations at FVE 0.9252 against an in-sample 0.9273—a gap of 0.2 percentage points. Whatever else is wrong with a 1,000-step SAE, it is not overfitting its training corpus.

We retain that checkpoint (`checkpoint_final_PARTIAL_1k.npz`) precisely so the measurement stays reproducible, but we are careful about how much it carries. It comes from the least-trained model in the set, at a point where features have barely specialised; a converged SAE has far more capacity to memorize. It is suggestive that the in-sample figures

Table 4: Reconstruction over the full 500,000-token corpus. All three converged runs are in-sample: each made 205 passes over its entire dump. The rows marked $*$ come from a partial 1,000-step Mistral checkpoint that saw only the first 50,000 tokens, and are the only genuine held-out measurement available. Dead-feature counts here are features that fire nowhere in the corpus, a stricter criterion than the rolling window of Table 2.

Model	Split	MSE	FVE	Dead features	
Llama-3.2-3B	in-sample	0.00617	0.9895	3,344 (20.4%)	205 epochs
Qwen2.5-3B	in-sample	0.27532	0.9839	9,398 (57.4%)	205 epochs
Mistral-7B	in-sample	0.00195	0.9659	2,346 (14.3%)	205 epochs
Superseded 1,000-step Mistral checkpoint, retained for the held-out measurement:					
Mistral-7B*	in-sample	0.00415	0.9273	24 (0.1%)	41 epochs
Mistral-7B*	**held-out**	0.00426	**0.9252**	26 (0.2%)	unseen

Table 5: Matchable feature inventory over the 500,000-token corpus. "Dead" features fire nowhere; "low support" features fire in fewer than 50 of 1,000 chunks. Only the last column enters the matching analysis.

Model	Dictionary	Dead	Low support	**Matchable**
Llama-3.2-3B	16,384	3,344	913	**12,127** (74.0%)
Qwen2.5-3B	16,384	9,398	22	**6,964** (42.5%)
Mistral-7B	16,384	2,346	5,863	**8,175** (49.9%)

are not pure memorization artifacts, and no more than that. Reserving a held-out token split *before* training is the cheapest fix available to this work, and none of these three runs did it.

4.4 Cross-Architecture Feature Matching

Between 43% and 74% of each dictionary is usable (Table 5). The attrition traces directly to the collapse of §4.1: Qwen is 57% dead, and Mistral—which reaches 32% dead by step 50,000—additionally carries a large low-support tail. Neither is an artifact of insufficient training; all three runs completed 50,000 steps.

The null. Table 6 reports the chunk-permutation null. The number to attend to is the mean: two features from different models with no functional relationship, matched reciprocally over a 1,000-dimensional fingerprint, correlate at 0.59–0.74 on average.

A fixed threshold of $\tau = 0.8$—the conventional choice—is at or below that noise floor for two of three pairs. The problem is structural rather than a matter of picking a better number: a match is the best of ∼5,000–8,000 candidates, and the expectation of an extreme-value statistic grows with the size of the search and shrinks with fingerprint dimensionality. At a lower resolution we also tested (100 chunks over 50,000 tokens, a common configuration) the null 99th percentile reaches 0.98–0.995, leaving the statistic with no discriminative power whatsoever.

Table 6: Chunk-permutation null, 20 replicates. τ is set to the null 99th percentile. "Expected FP" is the number of matches a shuffled control produces above τ. Data: `data/sae-analysis/matching-v2/matching-report.json`.

Pair	Null mean	Null p99 ($= \tau$)	Null matches/replicate	Expected FP
Llama–Qwen	0.595	0.950	736	7.4
Mistral–Qwen	0.594	0.944	706	7.1
Llama–Mistral	0.744	0.980	872	8.7

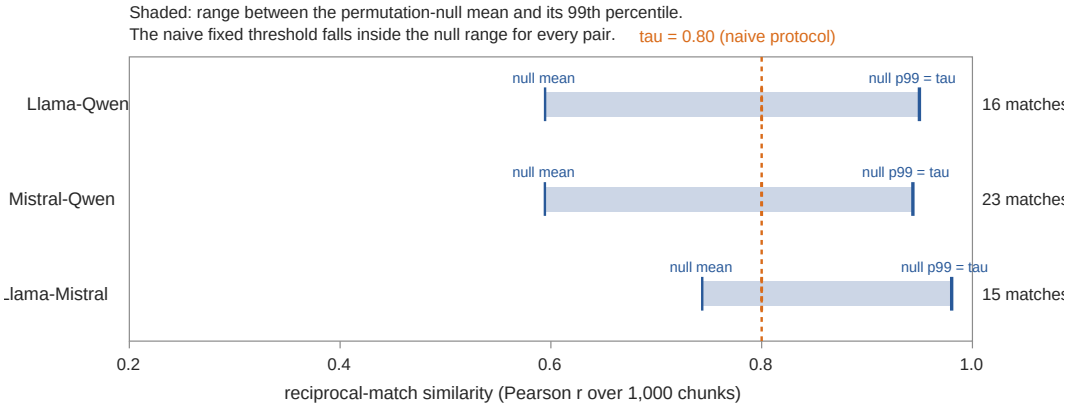

Figure 1: Permutation-null similarity ranges against the two candidate thresholds. The fixed $\tau = 0.80$ (dashed) sits between the mean and the 99th percentile of similarities produced by chunk-shuffled controls, for all three pairs. The calibrated threshold is the null 99th percentile; match counts at that threshold are shown at right. Data: `data/sae-analysis/matching-v2/matching-report.json`.

Figure 1 places the two candidate thresholds against the null directly. For every model pair the fixed $\tau = 0.8$ falls *inside* the null range—above the null mean, below the null 99th percentile—so it admits pairs a chunk-shuffled control produces just as readily.

Results. Table 7 gives the counts.

Exactly one feature triple matches across all three models, against a null expectation of 0.15 closed triangles per replicate. Under a Poisson model with that rate, $\Pr(\geq 1) = 0.14$; under the cosine metric the null rate is 0.55 and $\Pr(\geq 1) = 0.42$. One match is what this procedure produces from noise a substantial fraction of the time. Pairwise, 15–23 pairs survive per model pair against 7–9 expected by chance—an enrichment of 1.7–3.2×, or roughly 7–16 genuine pairs out of 7,000–12,000 matchable features (about 0.1%). At these counts Poisson uncertainty alone spans a factor of two, and we do not think the pairwise numbers support a quantitative claim.

The three corrections compound and none is individually sufficient. Reciprocal matching alone reduces the Llama–Mistral count from 12,174 to 40 at the *same* threshold, because the uncalibrated count recorded a handful of Mistral features many times over. The closed-triangle requirement removes chained triples—in the uncalibrated run, 67% of nominally

Table 7: Cross-architecture matches under the corrected protocol, against the uncalibrated protocol applied to the same checkpoints. Enrichment is observed matches over expected false positives.

Pair	Uncalibrated	**Corrected**	Expected FP	Enrichment
Llama–Qwen	5,923	**16**	7.4	2.2×
Mistral–Qwen	8,099	**23**	7.1	3.2×
Llama–Mistral	12,174	**15**	8.7	1.7×
Three-way universal	**3,753**	**1**	0.15	—

universal triples had no verified Llama–Mistral edge, and the 3,753 triples collapsed onto 644 distinct Qwen and 341 distinct Mistral features. Null calibration removes the remainder.

The Llama–Mistral asymmetry inverts. Under the uncalibrated protocol Llama–Mistral was by far the largest pair (12,174 matches, nominally 74% coverage), an asymmetry it is tempting to read as shared architectural lineage. Under the corrected protocol the same pair yields the *fewest* matches of the three (15) and the *lowest* enrichment (1.7×). The reason is visible in the null rather than in the models: the Llama–Mistral noise floor is far higher than the others (mean 0.744 against 0.59), so its calibrated threshold rises to 0.980 and few pairs clear it. A single threshold applied across pairs with different noise floors will reorder them, and the ordering it produces is an artifact of that floor. This is a second reason to calibrate per pair: not only the absolute counts but their *relative* sizes are otherwise unreliable.

The corrected regions partition the dictionary; the uncalibrated ones did not. With one three-way match the Venn structure over Llama's dictionary reduces to near-disjoint sets: 15 features match Qwen only, 14 match Mistral only, 1 matches both, and 16,354 of 16,384 match neither. We give the counts rather than a diagram because a Venn figure at these proportions would be illegible. The arithmetic is itself a check on the protocol. Reciprocal matching is one-to-one, so the regions must sum to the dictionary size, and they do. Under the uncalibrated protocol they summed to 18,258 regions for a 16,384-feature Qwen dictionary—more members than the set has—which was a visible signal that many-to-one matching had broken the set algebra, available without any null at all.

4.5 The Method's Ceiling: Same Model, Different Seed

A near-zero match count admits two readings: the models share little, or the detector cannot see sharing. Nothing reported so far distinguishes them. The direct test is to run the procedure on a case where sharing is guaranteed by construction.

We trained two further SAEs on the *same* Llama-3.2-3B layer-14 activations under the *same* configuration as the checkpoint used above—dictionary 16,384, $k = 128$, 50,000 steps, batch 2,048, identical learning-rate schedule—varying only the seed, which sets both weight initialisation and batch order. Any feature that is a real property of the model's representation, rather than an accident of one optimisation trajectory, should appear in all three

Table 8: Matching two SAEs that differ only in training seed, against the cross-architecture result from Table 7. Same source model, same activations, same hyperparameters. Pearson metric; the cosine metric gives 1,087–1,211 matches (8.9–10.0%) and is reported in the data release.

Pair	Matched	% of matchable	Enrichment over null
seed 123 vs. seed 456	965	7.93%	108×
seed 42 vs. seed 123	978	8.06%	109×
seed 42 vs. seed 456	1,002	8.26%	112×
Cross-architecture, from Table 7:			
Llama–Qwen	16	0.23%	2.2×
Mistral–Qwen	23	0.33%	3.2×
Llama–Mistral	15	0.18%	1.7×

dictionaries. We ran the identical matching procedure, including the same permutation calibration, on all three pairs.

The detector is not blind. Where sharing is guaranteed the procedure recovers it at 108–112× the permutation null, against 1.7–3.2× across architectures (Table 8). Instrument failure is therefore not the explanation for the cross-model result. That was the rival reading we could not previously exclude, and it is now excluded on a positive control rather than by argument.

But the ceiling is 8%, not 100%. Two SAEs trained on identical activations from the same model, differing only in seed, agree on roughly **8% of their matchable features**. Around 92% of each dictionary does not survive a reseed.

This changes the denominator for everything above. Cross-architecture sharing should be read against what is achievable rather than against dictionary size: 0.18–0.33% observed against an 8% ceiling, a factor of 25 to 45 below the maximum this method could return, rather than against 100%. The gap remains large and the conclusion of §4.4 stands, but stated against the wrong denominator it overstates its own strength. The bound is also general: no cross-model study using activation-pattern matching of this kind can report more than about 8% agreement, whatever models it compares, because that is what the method returns when the models are the same one.

What survives is near-duplicate. The matched pairs are not merely similar. Their similarity distribution has median 0.991, with 59% above 0.99 and 27% above 0.995. This is the planted-signal detection floor of §4.6 confirmed on real data: the procedure recovers features that are essentially identical across runs and nothing weaker. A feature genuinely shared but realised at correlation 0.9 is invisible here exactly as the simulation predicts.

The reproducible fraction is itself only half reproducible. Of the seed-42 dictionary, 978 features matched seed 123 and 1,002 matched seed 456—but those two sets intersect in only 515 features, 52.7% of the smaller set (Jaccard 0.352). There is no stable core of

Table 9: The three seed runs by standard SAE quality metrics, at step 50,000. Dead counts here are the rolling 5,000-step window logged during training, as in Table 2.

Seed	Final loss	FVE	L_0	Dead
42 (reported above)	0.006174	0.9939	128.0	3,753
123	0.006171	0.9951	128.0	3,600
456	0.006182	0.9898	128.0	3,696

robust features that reseeding recovers. Different runs recover overlapping but substantially different subsets, so reproducibility is not a property a feature has; it is closer to a property of the pair of runs being compared.

None of this is visible in the quality metrics. The three runs are indistinguishable by everything the literature reports.

Loss agrees to 0.2%, FVE to half a percentage point, L_0 exactly, dead-feature count to 4% (Table 9). By the standard report these are the same autoencoder trained three times. They share 8% of their features. Reconstruction quality, sparsity and dead-feature count are jointly blind to the identity of the dictionary—not merely insensitive to the collapse dynamics of §4.1, but unable to distinguish two dictionaries that disagree about 92% of their content. We think this is the strongest form of the paper's argument: the metrics the field reports do not identify what an SAE has learned.

Two results replicate along the way. Dead-on-evaluation counts across the three seeds are 3,344, 3,206 and 3,274, within 4% of one another, so the collapse magnitude of §4.1 is a property of the training procedure rather than an accident of one seed. And the fixed-threshold argument of §4.4 can now be demonstrated on known ground truth: the same-model permutation null has mean 0.757, so $\tau = 0.80$ sits barely above the noise floor even here, and returns roughly 5,500 matches against the 965 that survive calibration—a sixfold inflation in a case where the sharing being inflated is real.

Caveats. Fingerprints are computed on the corpus the SAEs trained on, following the protocol used throughout this paper, so the 8% ceiling is an in-sample figure. It is measured on one model at one layer, and whether that value is characteristic of TopK SAEs generally is unmeasured. The seed governs weight initialisation and batch order together, so this is total retraining variation rather than either source in isolation.

4.6 What the Method Could Have Detected

A null result is a statement about a detector as much as about the data, so we measured this detector's sensitivity rather than assuming it. Real Mistral SAE fingerprints supply the null population; into it we plant $K = 50$ synthetic cross-model pairs at a known correlation α, run the identical reciprocal-best-match procedure with the identical permutation calibration, and record the fraction recovered. 300 trials per point, α swept from 0.5 to 1.0, dictionary size swept from 256 to the full 8,208 matchable features. Table 10 reports the resulting detection floor.

Table 10: Minimum detectable effect size at 80% power. τ is the permutation-calibrated threshold; MDES is the smallest planted correlation the procedure recovers 80% of the time. Data: `data/power-analysis/power_table.json`.

Dictionary size	Null best-match mean	τ (null p99)	MDES @ 80% power
256	0.495	0.926	0.95
1,024	0.567	0.947	0.95
4,096	0.609	0.959	1.00
8,208	0.649	0.958	1.00

MDES tracks τ. At every dictionary size the smallest detectable correlation sits at or just above the calibrated threshold. This is not a tuning failure that a better choice of τ would fix—it is forced. The threshold must clear a null whose mean rises with dictionary size (0.495 at $D = 256$, 0.649 at $D = 8{,}208$), because the statistic is a maximum over D candidates and the expectation of a maximum grows with the number of draws. Raising τ to control false positives necessarily raises the floor on what can be found. The two move together by construction.

The practical consequence is severe and, we think, underappreciated. A pair of features genuinely corresponding at correlation 0.8—a correspondence most practitioners would call strong—is invisible to this method at any dictionary size we tested. The band the method can see, roughly $[0.95, 1.0]$, is narrow enough that it detects near-duplicates and little else.

Two caveats on the estimate itself. The simulation ran at 100 chunks against the 1,000 used in the main analysis, so its fingerprints are noisier, its τ higher, and the true MDES at full resolution is likely somewhat lower—the figures here are an upper bound. And planted pairs are synthetic: a real shared feature might differ from its counterpart in ways a fixed-correlation construction does not capture. Neither caveat is large enough to move the conclusion, because the mechanism—threshold chasing a growing null—does not depend on either.

What this does and does not show. It does not show that these three models fail to share representational structure. The obvious rival explanation for a near-zero count is instrument failure—that the procedure cannot see sharing at all—and §4.5 excludes it on a positive control: given two SAEs that differ only in seed, the same procedure recovers agreement at 108–112× the null. A second form of the objection, that these particular SAEs are too degraded, we can rule out directly. Running the identical analysis on partial checkpoints (Llama at 20% of its step target, Mistral at under 2% on a ten-times-smaller corpus) returns zero universal features. Training all three to the same specification raises the usable dictionary fraction from 51% to 74% (Llama) and 33% to 50% (Mistral), and moves the answer from zero to one—which is to say, not at all. A 70%-larger usable dictionary buys no additional matches, so undertraining is not what suppresses them.

Two confounds remain. A third to a half of each dictionary is still dead or low-support, so the analysis cannot see it; and the fingerprint statistic—mean activation over 500-token chunks—discards within-chunk structure, so a feature responding to a rare token is indistinguishable from one responding to a chunk's topic. A negative result from a coarse statistic is weak evidence of absence.

The power analysis (§4.6) sharpens what the null result claims. It is not "these models share no features" but "no feature triple is shared at correlation ≥ 0.95" — everything below that floor is outside the method's reach, and we cannot bound how much lives there. The seed control adds the other half of the frame: the finding is 0.18–0.33% agreement against a same-model ceiling of 8%, not against 100%. Stated with both bounds the result is considerably weaker than a bare zero suggests, and it is the strongest form the evidence supports.

What it does show is methodological and transfers beyond this dataset. Cross-model feature matching by activation-pattern similarity requires a permutation null: without one, the search over a large dictionary manufactures high similarities from nothing, and a threshold that reads as stringent can sit below the noise floor. But the null is not only a correction to apply—it is also a diagnostic. Where it lands tells you the method's detection floor, and if that floor sits at 0.95 the method is answering a much narrower question than the one being asked. We would encourage anyone reporting a cross-model matching result, positive or negative, to report the floor alongside it.

5 Discussion

5.1 Scope

The evidence here is three TopK autoencoders trained to a common specification on Llama-3.2-3B, Mistral-7B-v0.3 and Qwen2.5-3B, all at $k = 128$. Results characterise the TopK objective at one sparsity level on three models; they do not compare training objectives, and no L1, Gated or JumpReLU autoencoder is trained or evaluated anywhere in this work. No linear probes are trained on the resulting features, no human interpretability ratings are collected, and no steering evaluation uses decoder directions. No claim in this paper rests on any of these.

Monitor dictionary utilization, not just reconstruction. The collapse in §4.1 is invisible in FVE and in the loss curve. It is visible only in a dead-feature count, and only if that count is logged during training rather than read off the final checkpoint. Since the auxiliary loss of Gao et al. [6] is designed precisely to prevent it, the practical recommendation is simply to use it—but the measurement matters independently, because a run using the auxiliary loss still needs verification that it worked.

Report reconstruction over a corpus, not a batch. Batch FVE at the final step spans 0.36 across adjacent logged steps in our shortest run and 0.02 in our longest. A single reading is a draw from that distribution. Full-corpus evaluation costs one forward pass and removes the ambiguity entirely.

Expansion ratio is a hyperparameter, not a free choice. Holding dictionary size constant across models with different hidden dimensions—a natural way to make a comparison "controlled"—silently varies expansion ratio, and the model with the highest ratio collapsed hardest. Whether expansion ratio drives collapse or merely correlates with it here, we cannot say with two long runs.

Report the detection floor, not just the threshold. A cross-model matching result—positive or negative—is uninterpretable without knowing the smallest correspondence the procedure could have found. Ours is 0.95–1.00 (§4.6), which means our zero is compatible with a substantial shared vocabulary sitting just below it. The measurement costs one simulation against fingerprints you already have, and it converts an unfalsifiable "we found nothing" into a bounded claim.

Frequency-based feature selection selects the wrong features. This follows directly from §4.2 and is, we suspect, the most transferable practical finding. It affects any pipeline whose first step is "take the most active features."

5.2 Limitations

The matching method has a high detection floor. Minimum detectable effect size is a correlation of 0.95–1.00 (§4.6), so our negative universality result bounds shared structure only above that level. Lowering the floor is not a matter of more tokens: the null rises with dictionary size because the statistic is a maximum over candidates, so it needs a similarity measure whose null does not grow that way, or a matching procedure that does not search the whole dictionary per feature.

Only one objective. Everything reported here is about TopK SAEs without an auxiliary loss. The collapse dynamics in particular are specific to hard-sparsity training; L1 SAEs fail differently.

Confounded cross-model comparison. All three runs are matched on steps, corpus, batch size, k and dictionary size, so training length is not a confound. But expansion ratio, source model, layer depth and activation-norm scale all vary together across only three runs with one seed each. We report orderings and avoid attributing any of them to a single factor.

No held-out data for any converged model. All three trained over 205 epochs of their full dump. The single held-out measurement comes from a partial 1,000-step checkpoint. See §4.3.

Quantization confound. Mistral activations come from 4-bit weights.

Single corpus, single seed. All results use WikiText-103 and seed 42. There is one run per condition, so we report no variance across seeds and cannot say whether the collapse step is stable.

No semantic labelling. We never establish what any feature represents. The matching analysis asks whether activation patterns correspond, not whether the corresponding features mean the same thing—and with zero three-way matches, the question does not arise.

6 Conclusion

We trained TopK SAEs at matched sparsity and dictionary size on three model families and instrumented the training rather than only its endpoint. Without an auxiliary revival loss, the dictionary collapses abruptly between steps 2,000 and 5,800, leaving 14.3–57.4% of features permanently dead at convergence while reconstruction quality continues to improve—a

failure mode that the usual metrics conceal. What survives is dominated by dense directions: 1.2% of the dictionary carries roughly 40% of the sparse code, which invalidates frequency-based feature selection.

Cross-architecture feature matching finds essentially nothing that survives a permutation null. The conventional protocol applied to partial checkpoints reports 3,753 shared features; the corrected count is one—not statistically distinguishable from null ($\Pr(\geq 1) = 0.14$). The gap is largely attributable to three procedural choices—many-to-one matching, chained rather than closed triples, and an uncalibrated threshold—each of which is easy to make and none of which is visible in the output.

We report the adjacent objective comparison as not conducted.

Reproducibility Statement

This paper is archived at 10.5281/zenodo.21906712, a Zenodo concept DOI that always resolves to the current version.

All SAE training code, activation collection scripts, training logs, checkpoints, and analysis scripts are available at https://github.com/american-code/ResearchPapers, under `data/sae-runs/` (training logs and configurations for every run), `data/sae-analysis/` (the matching pipeline and its outputs), `data/seed-stability/` (the control of §4.5) and `data/power-analysis/` (the planted-signal sweep). The three converged runs used seed 42. Training was conducted on Apple Silicon using the MLX framework. The corrected matching analysis is `data/sae-analysis/cross_arch_matching_v2.py` and the superseded version is retained as `cross_arch_matching.py` for comparison; reconstruction and null statistics are regenerated by `data/sae-analysis/recompute_corrections.py`.

The activation collection and SAE training that produced these checkpoints ran on the two-node streaming system described in a companion paper [7], which reports the transport measurements, the bit-exactness of the split boundary, and the validation of the parameter-averaging rule used by the distributed trainer. Nothing in this paper's analysis depends on that system—the checkpoints and fingerprints it produced are archived and can be re-analysed directly—but it is where they came from.

Ethics Statement

This work involves no human subjects and no personal data; all activations are derived from WikiText-103, a public corpus of Wikipedia text.

SAEs and the feature-matching methods studied here have applications in interpretability and safety research and potential for misuse in targeted manipulation of model behaviour. Our results are negative and release no feature directions of either kind.

References

[1] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016. URL https://arxiv.org/abs/1610.01644.

[2] Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

[3] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Chris Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features/index.html.

[4] 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. URL https://arxiv.org/abs/2309.08600.

[5] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Chris Olah. Toy models of superposition. *Transformer Circuits Thread*, 2022. URL https://transformer-circuits.pub/2022/toy_model/index.html.

[6] Leo Gao, Tom Dudé la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093*, 2024. URL https://arxiv.org/abs/2406.04093.

[7] J. Melton. Distributed mechanistic interpretability at scale: Activation streaming, split-layer inference, and distributed sparse autoencoder training, 2026. URL https://doi.org/10.5281/zenodo.21906710. Companion paper. Describes the two-node activation-streaming system and the distributed SAE trainer that produced the checkpoints and fingerprints analysed here. The DOI is a Zenodo concept DOI and always resolves to the current version.

[8] Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Tom Lieberum, Vikrant Varma, János Kramár, Rohin Shah, and Neel Nanda. Improving dictionary learning with gated sparse autoencoders. *arXiv preprint arXiv:2404.16014*, 2024. URL https://arxiv.org/abs/2404.16014.

[9] Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. Jumping ahead: Improving reconstruction fidelity with JumpReLU sparse autoencoders. *arXiv preprint arXiv:2407.14435*, 2024. URL https://arxiv.org/abs/2407.14435.

[10] Alex Turner, Lisa Thiergart, Gavin Udell, David Leike, Ulisse Mini, and Monte MacDiarmid. Activation addition: Steering language models without optimization. *arXiv preprint arXiv:2308.10248*, 2023. URL https://arxiv.org/abs/2308.10248.

[11] Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023. URL https://arxiv.org/abs/2310.01405.