PREDICTION IS NOT DESCRIPTION: AN EXTERNAL-LABEL AUDIT OF SPARSE AUTOENCODERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Sparse autoencoders (SAEs) are evaluated almost entirely from inside the model. On those metrics they are hard to tell apart from random directions, and simple supervised baselines beat them. Yet that verdict conflates two properties: whether a direction separates a concept, and whether it can be described. Telling them apart needs a criterion the model never saw. MIMIC-IV admissions carry ICD-9 codes that human coders assign from the chart for billing, with no automated process involved. We train vanilla and JumpReLU SAEs on Gemma-2-2B activations from 50,000 discharge summaries, then audit them against a fixed 46-code panel on held-out notes, beside six alternatives: a general-purpose SAE, covariance-matched random directions, PCA, lexical n-grams, and two directions fitted on the codes themselves. Grounding against the codes fixes a common scale of correlation strength, on which the SAEs reach $|r_{pb}| = 0.86$ against 0.43 for the random directions. Three tests then measure separate properties along that scale. Description and ablation both put the two domain-trained SAEs first. At matched correlation a blind judge recovers the code from an SAE feature's unsupervised explanation 62–75% of the time, and from a direction fitted on that code only 17–38%, at or below the random floor. Ablation raises loss more on notes carrying the code for every domain-specific SAE target, and for no random direction. The general-purpose SAE, applied zero-shot, falls below both on every test it enters. Prediction reverses the order: the lexical n-grams classify the codes better than either SAE. Prediction and description are separate properties of a direction, and it is description that the sparse decomposition supplies.

1 INTRODUCTION

Sparse autoencoders (SAEs) decompose a language model's activations into a large dictionary of sparsely firing latents, and have become the dominant tool for turning distributed representations into named features. Foundational work established that they recover monosemantic directions and support causal intervention (Bricken et al., 2023; Cunningham et al., 2023). Later work scaled the recipe to large models (Gao et al., 2024; Rajamanoharan et al., 2024; Lieberum et al., 2024) and extended it toward circuit-level attribution (Shu et al., 2025; Ameisen et al., 2025). A critical literature now asks whether auto-interpretability scores, reconstruction fidelity, sparse probing and causal editing can separate a learned feature from an artifact of the decomposition (Korznikov et al., 2026; Heap et al., 2025; Ma et al., 2026; Chanin, 2026). These critiques share a reliance on *internal* evaluation: metrics computed inside the model's own computational graph, or scored by a language model reading the same activations. One response is to validate features against ground truth independent of both the model and the decomposition.

Clinical text supplies such a source. Hospital discharge summaries describe diagnoses and treatments in dense specialized language, and every admission is annotated with ICD-9 diagnosis codes by human medical coders, assigned from the chart for billing without reference to any automated text process (Johnson et al., 2023b;a). If the model's representation of a summary encodes clinical structure, and an SAE recovers it as individual latents, those latents should be verifiable against the labels: they should correlate with particular codes, separate the notes that carry them, be describable as those codes, and change the model's predictions when ablated. The labels are noisy, hierarchical, heavily comorbid, and correlated with note length and acuity. Comorbidity also makes the off-target

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

test informative: a feature that fires on a diagnosis and on everything that travels with it is a different object from one that fires on the diagnosis alone.

An external label discriminates only if the alternatives face the same audit: the same notes, panel, selection rule and explanation budget. Otherwise the SAE's advantage comes from the protocol rather than from the features. Existing comparisons do not meet that bar. Each places an SAE against one alternative on one axis, and none includes a concept direction fitted on the labels. That is the competitor an external-label evaluation most needs, since it can beat the SAE at what the labels measure.

Contributions. (1) A prediction/description dissociation. At matched correlation strength, a blind judge recovers the code from an SAE feature's unsupervised explanation 62–75% of the time, and from a direction fitted on that same code only 17–38%, at or below the floor set by covariance-matched random directions. These sources separate a code equally well but are not equally describable, and no measure of predictive strength distinguishes them; over whole representations a supervised bag of n-grams outpredicts every SAE we audit.

(2) Domain training, not training alone, is what the external labels reward. A general-purpose SAE of the same architecture reconstructs clinical activations worse than their own mean. On the external labels it falls below both domain SAEs on every test it enters while staying above the random and PCA arms, so the ordering separates *learned* from *learned on the right data* rather than trained from untrained. This extends O'Neill et al. (2025), whose evidence is reconstruction fidelity and auto-interpretability, to labels independent of both the model and the decomposition.

(3) Grounding against external labels, audited three independent ways. Two domain SAEs and six other sources face the same 46 codes, split, thresholds and selection rule (Appendix J). Each other source tests one alternative explanation of an SAE win: random directions (candidate search), PCA (any unsupervised basis), a general-purpose SAE (wrong training data), difference-in-means and probe directions (label fitting), and TF-IDF (vocabulary). Every source is grounded against the codes and then enters each of three tests (prediction, description and ablation) that applies to it; the tests fail in unrelated ways, and the two domain SAEs face all three. Label-fitted directions have been compared with SAEs on detection and steering previously (Wu et al., 2025), but we evaluate whether they can describe the concept of an external label and help a blind judge choose the same from the description.

2 Related work

Negative results on SAE evaluation. The case we are testing against is that standard SAE metrics do not distinguish a learned feature from an unlearned direction. Korznikov et al. (2026) find in a synthetic setting with known ground truth that SAEs recover 9% of true features at 71% explained variance, while baselines constraining SAE directions or activations to random values match trained SAEs on interpretability scoring, sparse probing and causal editing. Heap et al. (2025) report the same for SAEs trained on randomly initialized transformers and recommend routine randomized baselines, a norm we adopt throughout. Tuned baselines also match or exceed SAE probes under data scarcity, class imbalance, label noise and covariate shift (Kantamneni et al., 2025); steering works only under careful feature selection (Arad et al., 2025); and learned features are neither complete nor atomic (Leask et al., 2025). On Gemma-2-2B, our base model, SAEs are not competitive with simple baselines on either steering or concept detection, with difference-in-means strongest on detection (Wu et al., 2025). Peng et al. (2026) reconcile this literature with the positive one by separating discovery of unnamed concepts from action on named ones. §6 reports the prediction half of our own comparison running similarly, and §5.3 tests that distinction at matched correlation strength, where the question is not which direction detects a concept but which one can be described without being told what to look for.

Benchmarks and the metrics they are built from. SAEBench (Karvonen et al., 2025) scores over 200 open SAEs on eight metrics, treating PCA components as latents and, following Heap et al. (2025), adding SAEs trained on a randomly initialized model. We adopt the PCA construction and the randomized-baseline norm. Its supervised comparison is a probe on the raw residual stream rather than a label-fitted concept direction entered as a competing source, which is where our audit harness differs. Chanin (2026) finds two of its metrics failing multiple reliability lenses and the rest carrying more reseed noise than assumed. McCann (2026) shows detection-style scoring to be invariant to

many features sharing one explanation, so a high score can arise without feature discrimination. The blind retrieval task of §4.3 is a discrimination scorer built to avoid this. Explanation and scoring otherwise follow Bills et al. (2023) as developed by Paulo et al. (2024), including the detection and fuzzing scorers and their random-explanation baseline of 0.51. Prior scores supplied as context metadata shift LLM judgments in seven of eight models tested (Kapetanovic et al., 2026), which is why our judge prompts state no correlation value.

Domain-specific and clinical SAEs. O'Neill et al. (2025) train JumpReLU SAEs on Gemma-2 activations over clinical question answering and report up to 20% more explained variance for domain-confined dictionaries than for broad-domain ones. That domain SAEs reconstruct their own domain better is established; what we add is that the gap survives three external-label tests that reconstruction quality does not speak to. Sainsbury et al. (2026) decompose a clinical sequence model and report 21% feature reproducibility across seeds, the sharpest open question about any single-seed clinical SAE result; we train our JumpReLU SAE at two seeds in response (Appendix B). Mu and Chen (2026) train SAEs on MIMIC-IV discharge notes and label the latents with an LLM-assisted pipeline under ICD-10 retrieval constraints, producing a feature-level audit trail for mortality prediction. There the codes constrain the interpretation; here they are withheld from the explainer and used as the criterion it is scored against. Modi et al. (2026) train SAEs on 1 billion tokens from 50,000 MIMIC-IV notes but on medical-specialist models and to study polysemy in clinical reasoning rather than external-label grounding. Automatic ICD coding from discharge summaries is a mature supervised task (Mullenbach et al., 2018; Vu et al., 2020; Yuan et al., 2022). We use the codes as external labels rather than as a prediction target, and include the TF-IDF source to bound how much of the signal is lexical.

3 SETUP

3.1 CORPUS AND EXTERNAL LABELS

We use discharge summaries from MIMIC-IV (Johnson et al., 2023b) and its note module (Johnson et al., 2023a), restricted to ICD-9-coded admissions for a single label space. Each summary is paired with the ICD-9 codes assigned to its admission by human medical coders, and we use these codes as our external labels. We use 50,000 summaries from the train and validation splits, stratified by diagnosis rarity so that uncommon codes are over-represented (Appendix A). The label panel consists of the 46 codes with prevalence greater than 2%, fixed in advance, and used consistently in all sources and test sets.

The codes were assigned without reference to the SAE, the base model or our measurements. They are also highly comorbid, allowing us to distinguish features that track a diagnosis from those that instead track conditions commonly associated with it. Both properties make the labels a stricter criterion than a curated interpretability set, at the cost of administrative label noise (§7).

Splits. Selection uses shards 0–30 ($n = 5{,}001$ notes), audit shards 281–311 ($n = 4{,}911$). All reported numbers are computed on the audit split, which is also the held-out set used for SAE checkpoint selection. Corpus construction, the extraction checks and the released artifacts are in Appendices A and I.

3.2 SPARSE AUTOENCODERS

Residual-stream activations are taken at layer 16 of Gemma-2-2B (Gemma Team et al., 2024), about 62% of the way through the 26-layer model, and mean-centered over the corpus. We train two SAEs on these clinical activations, both with 18,432 latents (an eightfold expansion). The JumpReLU SAE (Rajamanoharan et al., 2024) has a per-latent threshold, an L_0 penalty and straight-through gradient estimation; the vanilla SAE (Bricken et al., 2023; Cunningham et al., 2023) has a ReLU, an L_1 penalty and periodic dead-latent resampling. On held-out activations the JumpReLU SAE reaches mean L_0 40.92 at explained variance 0.906, and the vanilla SAE's L_0 47.57 at 0.889 (Appendix B).

As a general-purpose counterpart we apply the pretrained layer-16 GemmaScope SAE (Lieberum et al., 2024) of width 16,384 zero-shot. It shares the JumpReLU architecture but was trained on web text. On clinical activations it reconstructs worse than the mean (explained variance -4.21), and splicing its reconstruction into the model costs 0.648 nats against 0.029 for the vanilla SAE (Appendix G).

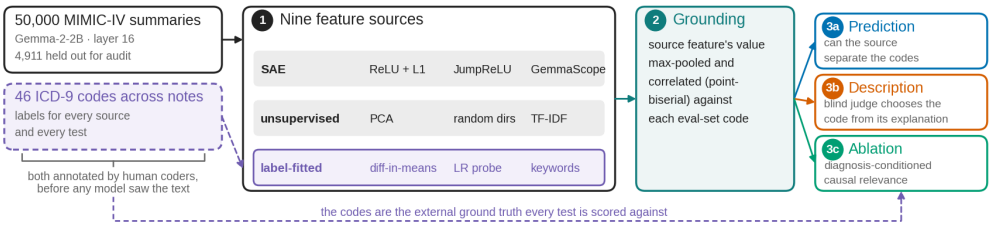


Figure 1: Evaluation overview. Every source (Table 1) is grounded against the same 46 codes and compared with the SAEs on the tests that apply to it.

3.3 Comparison sources

A *source* is a set of candidate features, each of which assigns a value to every token of a note; a note's value for a feature is the maximum over its tokens. Sources differ along two axes. Most are defined in the model's representation space, lexical sources are defined on the note text instead. Independently, a source is either built without the ICD-9 codes or built from them. The SAEs are built in the representation space without the codes; the others are listed below by how they are built.

Built without the codes: **Covariance-matched random directions** are 18,432 samples from $\mathcal{N}(0, \Sigma)$, with Σ estimated on the training shards; they match the SAE's dictionary size and the covariance of the activations, without learning. Their thresholds are set to reproduce the JumpReLU SAE's note-level firing rate, and alternative calibrations are described in Appendix F. **PCA** uses the 2,304 principal components of the same activations, following Karvonen et al. (2025). These two we call *constructed*: built from the activations alone, with neither learning nor labels. **TF-IDF** represents each note by 10,000 unigram and bigram features computed on the note text.

Built from the codes: **Difference-in-means** takes, for each code, the mean activation of training notes carrying the code minus the mean of notes without it, whitened by the activation covariance; this is the mass-mean probe of Marks and Tegmark (2023). The **logistic probe** is the L_2-regularized coefficient vector of a per-code classifier with class-balanced weights. Both are fitted on the training shards. The **lexical baseline** uses a curated list of keywords written for each code. On the note text it is a binary indicator of whether any keyword occurs. Variants of these sources are described in Appendix F.

Selection. Every source is reduced to one feature per code. For the SAEs, random directions, PCA and TF-IDF, this is the candidate with the largest $|r_{pb}|$ against the code on the selection split, so the codes are used only to choose among features that were built without them. The difference-in-means, probe and lexical sources have one feature per code by construction. All sources share the code panel, the selection and evaluation splits and the thresholds (Appendix J). TF-IDF and the text form of the lexical baseline have no direction in the model's activation space, so they enter only the tests that do not require one.

4 Grounding and three independent tests

Each source's features are first grounded against the codes, which fixes a common scale of correlation strength. Three tests then measure separate properties along that scale, as described in 3a, 3b and 3c of Figure 1, by procedures specified in full in Appendices C, E and D. No two of them can fail for the same reason.

4.1 Grounding

A feature's values are max-pooled over each note's tokens and correlated with each binary code label (point-biserial, BH at $q = 0.05$). This gives two quantities. *Strength* is $|r_{pb}|$ against the feature's own code, and a feature is *grounded* at threshold τ when that correlation is BH-significant and exceeds τ; strength is the scale the three tests are read along. *Leakage* is the mean $|r_{pb}|$ against the other 45 codes, computed on notes that do not carry its own code, so that comorbidity is not counted as non-specificity. Judge comparisons between sources are made within bands of strength,

$|r_{pb}| \in [0.1, 0.3)$, $[0.3, 0.5)$ and ≥ 0.5, never pooled across them. Grounding runs entirely outside the model, and what defeats it is the pooling statistic: a maximum over a note's tokens grows with note length, inflating association for long or high-acuity notes. We report raw correlations and repeat the analysis after residualizing each note value on its token count (Appendix C).

4.2 Prediction

Prediction asks how well a source's whole representation separates notes that carry a code from notes that do not. We fit one L_2-regularized logistic regression per code on all of a source's features (the raw activations, the SAE latents or the TF-IDF vector), train it on the training shards and report AUC-ROC and AUC-PR on the evaluation split (Appendix F). The test is blind to whether the model uses a feature and to whether it can be named. It rewards any signal correlated with the code, lexical shortcuts and note length included, and its capacity grows with the number of features.

4.3 Description

Describability measures whether a feature's explanation identifies the code it correlates with. An explainer model describes a feature from its top-activating token contexts alone (Appendix D characterizes those contexts). A judge, blind to the correlation, the code and the source, picks the best fit from nine options: the true code, seven distractors from its own ICD-9 chapter, and an explicit `none of these`. Chance is $1/9 = 11.1\%$ by construction and 14–15% averaged over the realized slates, which are smaller wherever the chapter cannot supply seven distractors (Appendix D). Neither model sees a label or a grounding statistic. The risk here is linguistic: a lenient verdict set, a prompt stating the correlation, or cross-chapter distractors each let a judge score well without identifying anything. We fix all three in the strict direction; §6.3 and Appendix K report what the lax alternatives do.

4.4 Ablation

Ablation asks whether the model uses a feature for its code, which is given by *diagnosis-conditioned causal relevance*. SAE latents are mean-ablated: the latent is held at its dataset mean inside the SAE's reconstruction of the layer-16 residual stream, and layers 17 onward are re-run. Unlike zero ablation, which pushes the activation outside its usual range, mean ablation removes only the feature's note-specific variation, so a rise in loss reflects the information the feature carries rather than an unusual input. We compare the rise in cross-entropy on notes that carry the code against notes that do not (one-sided Mann–Whitney U, Cliff's δ, Benjamini–Hochberg at $q = 0.05$), for each source's ten most strongly grounded features. This test is threatened by the distortion the decomposition already introduces, so effects are baselined on the reconstructed pass and not the clean one. Because the two interventions leave different distortion behind, sources are compared only through Cliff's δ (Appendix E). Lexical sources have no direction to remove and hence were not tested.

The four confounds are unrelated: note length, feature count, verdict leniency and reconstruction distortion. An ordering that survives all four is not produced by any one of them, and a disagreement between tests is then a difference between properties rather than a shared artifact.

5 Results

5.1 Correlational Grounding

Domain trained SAEs ground the strongest. The vanilla and JumpReLU SAEs reach a median $|r_{pb}|$ of 0.574, and they are the only sources with features above 0.6 (61 and 73; Table 1). A second training seed of the JumpReLU† SAE reproduces this (median 0.586, 63 features above 0.6). Random directions ground almost as many features as the SAEs at $|r_{pb}| > 0.1$ (6,945), but only 16 above 0.3 and none above 0.5, so counts at low thresholds do not separate learned features from arbitrary ones. The general-purpose GemmaScope SAE has 54 features above 0.3 and none above 0.7.

Label-supervised directions land below them. Difference-in-means reaches a median of 0.339 and the logistic probe 0.327, and both leak more onto other codes (0.057 and 0.092, against 0.032–0.037 for the domain SAEs). TF-IDF comes closest to the SAEs, with a median of 0.531, a peak of 0.831 and the lowest leakage of any source; its lower counts reflect its smaller number of candidate features.

Table 1: Grounding on the evaluation split, one feature per code. Median and peak $|r_{pb}|$ are over the 46 selected features; leakage is their mean $|r_{pb}|$ against the other codes. The last four columns count features in each source's full set above each threshold. †A second training seed of the JumpReLU SAE. All variants are in Table 9.

Source	Features	Median	Peak	Leakage	>0.1	>0.3	>0.5	>0.6
JumpReLU SAE	18,432	0.574	0.864	0.037	**9,721**	610	**147**	61
JumpReLU SAE, seed 2†	18,432	**0.586**	**0.874**	0.035	9,632	638	137	63
Vanilla SAE	18,432	0.574	0.859	0.032	8,985	**675**	143	**73**
TF-IDF	10,000	0.531	0.831	**0.027**	2,879	265	60	29
Difference-in-means	46	0.339	0.699	0.057	46	28	7	2
Logistic probe	46	0.327	0.630	0.092	46	32	5	2
GemmaScope SAE	16,384	0.309	0.545	0.047	5,790	54	4	0
Random directions	18,432	0.191	0.432	0.056	6,945	16	0	0
PCA	2,304	0.141	0.441	0.048	256	1	0	0

Table 2: Prediction from the whole representation. Per-code L_2 logistic regression on all of a source's features, averaged over the 46 codes. Raw activations are the max-pooled residual stream that the SAEs decompose.

Source	Features	AUC-ROC	AUC-PR
JumpReLU SAE	18,432	0.888	0.543
JumpReLU SAE, seed 2†	18,432	0.870	0.487
Vanilla SAE	18,432	0.881	0.532
Raw activations	2,304	0.808	0.353
TF-IDF	10,000	**0.917**	**0.590**

Each source was also run in alternative forms (plain and diagonally whitened difference-in-means, an unweighted probe, the value form of TF-IDF, and random directions calibrated to token-level sparsity) reported in Table 9 (Appendix C).

5.2 Prediction

A bag of n-grams separates the codes best. Table 2 fits one logistic regression per code on each source's full set of features. TF-IDF reaches a mean AUC-ROC of 0.917, against 0.888 for the JumpReLU SAE and 0.881 for the vanilla SAE. The gap is consistent across codes: against the vanilla SAE, TF-IDF is ahead on 33 of the 46 codes, within 0.02 on the remaining 13, and behind on none (paired Wilcoxon signed-rank, median $\Delta = -0.033$, $p < 10^{-8}$). AUC-PR, which is more sensitive to the rarer codes, shows the same ordering (0.590 against 0.532, median $\Delta = -0.059$). As a whole representation, the SAE is not the best available predictor of these labels.

Within the model, the SAE adds to the representation it decomposes. The raw residual stream at the same layer, max-pooled over each note, reaches 0.808 AUC-ROC and 0.353 AUC-PR. Re-expressing those activations in the SAE's basis raises both, to 0.881 and 0.532 for the vanilla SAE, without any access to the labels.

The ordering changes at the level of a single feature. Taking each source's best feature per code instead of its whole representation, the SAEs lead: median $|r_{pb}|$ of 0.574 for both domain SAEs against 0.531 for the best TF-IDF term, 0.339 for the difference-in-means direction and 0.327 for the logistic probe (Table 1). Over all codes, the vanilla SAE's best latent correlates more strongly than TF-IDF's best term on 23 codes and less strongly on 1, with 22 within tolerance. The directions fitted directly on the labels therefore predict each code less well than an unsupervised latent selected for it, while TF-IDF's advantage appears only when all 10,000 terms are combined.

5.3 LLM Description Code Retrieval

The judge task is the blind nine-way forced choice of §4.3, at a chance floor of 11.1% by construction (14–15% realized), with a second pass in which the judge is shown the code and must endorse it unaided. Results are reported *within* $|r_{pb}|$ bands: retrieval accuracy rises steeply with $|r_{pb}|$ and the sources do not occupy the bands equally, so any pooled figure is a composition artifact.

Table 3: Blind forced-choice retrieval, hit@1, same-chapter distractors, 14–15% realized. Cells give the range across four judges from four laboratories (Claude Sonnet 4.6, DeepSeek-V3, GPT-5-mini, Gemini 2.5 Flash); a single value means all four agree. n is shared across judges; a dash means fewer than five features exist. Per-judge cells and Fisher tests against the random floor are in Appendix K. †JumpReLU SAE seed 2: a second training seed.

	$\lvert r_{pb}\rvert \in [0.1, 0.3)$		$[\mathbf{0.3}, \mathbf{0.5})$		$> \mathbf{0.5}$	
source	n	hit@1	n	hit@1	n	hit@1
keyword *(ceiling)*	6	50–83	26	92–100	6	100
JumpReLU SAE	139	7–12	175	**70–75**	147	**91–94**
JumpReLU SAE, seed 2†	139	10–16	184	**66–76**	137	**87–95**
vanilla SAE	126	13–17	177	**62–73**	143	**94–96**
GemmaScope SAE	100	0	50	46–52	—	—
diff-in-means (LDA)	18	22–28	21	24–38	7	29–43
probe LR (balanced)	17	12–18	24	17	5	0–40
PCA	34	12–15	—	—	—	—
random (note-matched)	284	16–23	16	38–44	—	—

SAE features are more describable at matched correlation. Recovery rates are reported within $|r_{pb}|$ bands (Table 3, Figure 2), because sources occupy the bands unequally. Random directions set the empirical floor at 38–44%, well above chance either way, since their explanations still read as clinical text. In the $[0.3, 0.5)$ band, the JumpReLU SAE is recovered 70–75% of the time, the vanilla SAE 62–73% and a second JumpReLU seed 66–76%. The difference-in-means direction reaches 24–38% and the logistic probe 17%, at or below the floor, although both were fitted on the codes the judge is asked to name. For these directions the judge most often answers `none of these` (52–83%, against 13–28% for SAE features).

Recovery rises with correlation strength. Below $|r_{pb}| = 0.3$, every source is recovered at low rates, the domain SAEs included (7–17%). Above 0.5, the seed-42 SAEs reach 91–96% and a second JumpReLU seed 87–95%, while the few supervised directions in that band reach 29–43%.

Confirming a single code removes the random floor. In a second pass the judge is shown the code and must endorse the explanation outright (Table 12). Random directions fall to 0–3%. Above $|r_{pb}| = 0.5$, seed-42 SAE features are endorsed 42–62% of the time against 14–29% for supervised directions; in the $[0.3, 0.5)$ band the two are similar (12–23% against 19–29%). All four judges order the sources the same way (Appendix K).

5.4 Causal ablation

Each source is tested for diagnosis-conditioned causal relevance (§4.4) on its ten most strongly grounded features, which for the domain SAEs lie above $|r_{pb}| = 0.79$. On this test both domain SAEs outperform every other source in effect size, in the number of significant targets and in off-target specificity (Table 8 and Figure E, Appendix E). Off-target specificity counts how often ablating a feature also produces a significant effect on one of the 45 other codes; with ten targets per source there are 450 such tests, and a feature that is specific to its code should produce none. The vanilla SAE is significant on all ten targets (Cliff's $\delta = 0.312$; mean on-target effect 0.002 nats) and shows no significant effect on any other code. The JumpReLU SAE is also significant on all ten targets ($\delta = 0.276$), with 4 of 450 off-target effects.

6 Discussion

The three tests do not agree, and the disagreement is the finding. Description and ablation put the two domain-trained SAEs first at high correlational strength; prediction, over whole representations, puts a supervised bag of n-grams first. Not every source enters every test: the lexical sources have no direction to ablate, and the general-purpose SAE no whole-representation fit. The judge, run in two modes, reverses below $|r_{pb}| = 0.3$ in both.

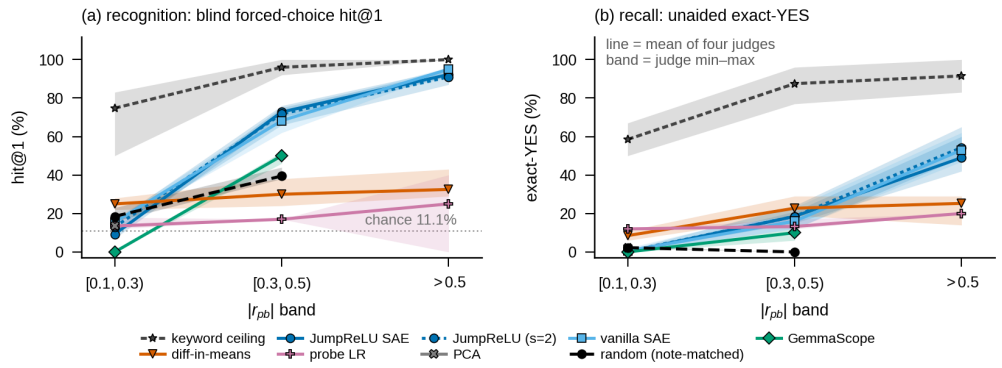


Figure 2: Describability is gated by grounding strength and supplied by domain training. **(a)** Blind forced-choice retrieval, hit@1, from the true code, seven same-chapter distractors and `none`; the dotted line is the $1/9$ design floor, against a realized mean of 14–15% **(b)** Unaided exact-YES endorsement. A source is absent from a band when it holds fewer than five features in it (Tables 3 and 12).

6.1 Prediction and description are separate properties

An SAE feature is not the best available predictor of an ICD code and we do not claim it is. A supervised bag of n-grams classifies the 46 codes better than the SAE representation (mean AUC-ROC 0.917 against 0.881, TF-IDF ahead on 33 of 46 codes and the SAE ahead on none, paired Wilcoxon signed-rank (Wilcoxon, 1945) $p < 10^{-8}$), and on the correlational audit it is the more specific source of the two (18.02× against 15.88×, Table 10).

The dissociation appears when predictive strength is held fixed. In $|r_{pb}| \in [0.3, 0.5)$, the only band in which the domain SAEs, the fitted directions and the calibrated null all hold testable populations, a domain SAE feature's unsupervised explanation lets a blind judge recover its code 62–75% of the time while a direction fitted on that same code manages 17–38%, at or below the random floor (§5.3).

Recognition and recall bracket the measurement and do not agree everywhere. Under unaided endorsement the separation in that band disappears (§5.3), which on 21 features against 175 is not a result either way and we do not report it as one; above $|r_{pb}| = 0.5$ the two instruments agree. Neither band supplies everything. The overlap band has a live null but a separation on only one instrument; the band above it has both instruments agreeing but no null left.

Describability appears only where grounding is strong. Below $|r_{pb}| = 0.3$ the ordering reverses on both instruments: fitted directions are recovered more often than either domain SAE, and arbitrary directions outscore both (Table 3). Neither result is about explanation quality. A direction fitted on code c is about c at any strength, so asking a judge to name c from its description tests almost nothing. Random directions do well for another reason: their explanations are still fluent clinical text, and a nine-way choice within one ICD-9 chapter is often winnable on topic alone (§5.3). SAE latents at $|r_{pb}| = 0.2$ are not tracking these codes, so the judge has nothing to recover. Below 0.3 retrieval does not measure describability, and we pool nothing across bands. TF-IDF, the strongest non-SAE source in the overlap band, was not run on the judge task, so whether a lexical feature at matched correlation is as nameable as a learned one remains open.

6.2 Only domain-trained features survive at the top of the range

The sources do not span the same range of correlational strength, and two thresholds matter. The first is where the calibrated nulls die: the note-matched random arm peaks at 0.432, so below 0.4 an arbitrary direction is a live competitor and we report the floor beside every claim there. Above 0.4 the null is exhausted to a single direction in 18,432. The second is the competitors' ceiling. Every source defined in the model's activation space other than the domain SAEs has a global maximum below 0.7, and above that line they hold no features at all, not few (Appendix J). Above 0.7 there is nothing to match against, so the comparison of §6.1 is unavailable rather than won, and what a domain-trained dictionary provides is features at strengths no constructed alternative in the model's own representation space reaches.

Where the TF-IDF comparison stops. TF-IDF trails the domain SAEs slightly on grounding strength, leaks less, and beats them on prediction, yet it cannot be asked the question that most directly concerns interpretability: whether the model uses this feature.

6.3 What the external labels made possible

The labels do not make the evaluation objective: an explainer still writes every description and judges still score them. What they change is that the model is no longer both the subject of the measurement and its scorer. Three things follow. The labels supply a competitor fitted directly on the target, the class that can beat an SAE at what the labels measure and does so on prediction; an axis along which sources can be matched, without which the dissociation of §6.1 cannot be constructed; and an interpretable score for a random direction, whose correlation with a diagnosis assigned by a human coder before the corpus existed means exactly what it says.

Familiar statistics fail without a calibrated arm. Four quantities reported here would mislead if read on their own. A grounded-feature count needs its threshold: at $|r_{pb}| > 0.1$ the L_0-matched null grounds more directions than the vanilla SAE, and by 0.3 it grounds one against 675. Monospecificity needs it too, since thresholding selects features sitting just above the bar (Appendix C). A forced-choice slate needs its floor, because it is guessable where an unaided endorsement is not. A negative Cliff's δ needs its control: it marks a length-confounded null, not an inverse effect. The labels buy a comparison in which a worse score would have been visible.

7 Limitations

Our evaluation inherits the limitations of its components: SAEs need not recover canonical or atomic features (Leask et al., 2025), auto-interpretability scores alone cannot separate trained from random directions (Heap et al., 2025; McCann, 2026), and ICD-9 codes are assigned for billing and are known to under-code by up to 35% in MIMIC-III (Searle et al., 2020). The labels can validate only the concepts they reliably stand for. All activations come from one model and one layer. We train the JumpReLU SAE at two seeds (Appendix B), but the vanilla SAE and every constructed source are single draws, so representational structure remains partly confounded with seed variation, a concern sharpened by the 21% cross-seed feature reproducibility Sainsbury et al. (2026) report for a clinical SAE. The calibrated nulls are likewise single draws, so the random peak $|r_{pb}|$ of 0.432 carries no error bar. TF-IDF has no direction in activation space and is therefore absent from the ablation, and it was not run on the judge task (§6.1). One model wrote every explanation, so LLM-specific reading of clinical text cannot be excluded, though four judges from four laboratories score them.

8 Conclusion

We evaluated SAE features against ICD-9 codes assigned by human coders, comparing domain-trained SAEs with unsupervised, label-fitted and lexical alternatives on prediction, ablation and describability. The domain SAEs do not predict the codes best, but at matched correlation their features are the ones a blind judge can name, and the model depends on them specifically. A direction fitted on a code can separate it as well as an SAE feature and still not be describable: prediction and description are separate properties of a direction, and metrics built on predictive strength rate the two the same.

This offers one account of why SAEs lose to simple baselines on probing and steering benchmarks yet remain central to interpretability: those benchmarks measure prediction, and prediction is not what the sparse decomposition supplies. Evaluations that aim to credit SAEs for interpretability should hold predictive strength fixed and score description against labels the model never saw, with a calibrated random baseline.

The case that matters most remains open: concepts that no label names yet, where description is the only route to finding them. The construction is not specific to clinical text. It requires paired documents and structured annotation produced independently of them.

AI USE STATEMENT

In this work, we used generative AI tools to implement parts of the method and to support qualitative analysis of its outputs. Claude Sonnet 4.6 wrote the explanation of every audited feature, assigned each feature to one of five descriptive categories, and ran the fuzzing and detection scorers (§4.3, Appendix D). Four judges (Claude Sonnet 4.6, DeepSeek-V3, GPT-5-mini and Gemini 2.5 Flash) performed the retrieval and concordance tasks (§5.3, Appendix K). AI coding assistants helped write and edit parts of the analysis code. We have not used generative AI tools to generate synthetic data, develop theoretical frameworks, propose hypotheses, design the evaluation, clean or reformat the dataset, or interpret results; formulating mathematical claims, writing proofs and translation are not applicable to this work. Additionally, we used generative AI tools to edit the manuscript for readability. We have reviewed all AI-assisted work: the authors reviewed all AI-written code and AI-edited text, every judge run was audited per feature for blank, failed and unparseable responses (Appendix K), and every judge-side number is regenerated from the released per-feature records by a single script (Appendix I). We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work uses MIMIC-IV and MIMIC-IV-Note, deidentified records from Beth Israel Deaconess Medical Center released with institutional review board approval and a waiver of informed consent. The data were accessed by credentialed users under the PhysioNet Credentialed Health Data Use Agreement; no new data were collected and no patients were contacted. Note text was processed on our own compute instances and, for the explainer and scorers, through Amazon Bedrock, which PhysioNet lists as acceptable for credentialed data and which does not retain inputs or use them for training. The judges received only feature explanations and ICD-9 code descriptions, never note text. No note text or protected health information appears in this paper or in any released artifact; released artifacts are aggregate summaries scanned for identifiers (Appendix I), and we release code rather than data. ICD-9 codes are billing labels from a single U.S. hospital system and are known to under-code diagnoses (§7), so the features we identify reflect its coding practices and patient population. They are interpretability measurements, not validated clinical tools, and should not be used for diagnosis or coding.

REFERENCES

Emmanuel Ameisen, Jack Lindsey, Tom Conerly, Joshua Batson, Chris Olah, et al. Circuit tracing: Revealing computational graphs in language models. *Transformer Circuits Thread*, 2025. URL https://transformer-circuits.pub/2025/attribution-graphs/methods.html.

Dana Arad, Aaron Mueller, and Yonatan Belinkov. SAEs are good for steering – if you select the right features. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 10252–10270. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.519. URL http://dx.doi.org/10.18653/v1/2025.emnlp-main.519.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction, 2024. URL https://arxiv.org/abs/2406.11717.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300, 1995. doi: 10.1111/j.2517-6161.1995.tb02031.x. URL https://doi.org/10.1111/j.2517-6161.1995.tb02031.x.

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. https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html, 2023.

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 Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features/index.html.

David Chanin. Are sparse autoencoder benchmarks reliable?, 2026. URL https://arxiv.org/abs/2605.18229.

Norman Cliff. Dominance statistics: Ordinal analyses to answer ordinal questions. *Psychological Bulletin*, 114(3):494–509, 1993. doi: 10.1037/0033-2909.114.3.494. URL https://doi.org/10.1037/0033-2909.114.3.494.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models, 2023. URL https://arxiv.org/abs/2309.08600.

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

Gemma Team, Morgane Riviere, Shreya Pathak, et al. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

Thomas Heap, Tim Lawson, Lucy Farnik, and Laurence Aitchison. Automated interpretability metrics do not distinguish trained and random transformers, 2025. URL https://arxiv.org/abs/2501.17727.

Alistair Johnson, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. MIMIC-IV-Note: Deidentified free-text clinical notes (version 2.2). PhysioNet, 2023a. URL https://physionet.org/content/mimic-iv-note/2.2/.

Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, Li-wei H. Lehman, Leo A. Celi, and Roger G. Mark. MIMIC-IV, a freely accessible electronic health record dataset. *Scientific Data*, 10(1), Jan 2023b. ISSN 2052-4463. doi: 10.1038/s41597-022-01899-x. URL http://dx.doi.org/10.1038/s41597-022-01899-x.

Subhash Kantamneni, Joshua Engels, Senthooran Rajamanoharan, Max Tegmark, and Neel Nanda. Are sparse autoencoders useful? a case study in sparse probing, 2025. URL https://arxiv.org/abs/2502.16681.

Ante Kapetanovic, Kemal Altwlkany, Andro Mercep, Tomislav Duricic, and Emanuel Lacic. Anchoring bias in LLM-as-a-Judge systems: Prior scores compromise evaluation independence, 2026. URL https://arxiv.org/abs/2608.25869.

Adam Karvonen, Can Rager, Johnny Lin, Curt Tigges, Joseph Bloom, David Chanin, Yeu-Tong Lau, Eoin Farrell, Callum McDougall, Kola Ayonrinde, Demian Till, Matthew Wearden, Arthur Conmy, Samuel Marks, and Neel Nanda. SAEBench: A comprehensive benchmark for sparse autoencoders in language model interpretability, 2025. URL https://arxiv.org/abs/2503.09532.

Anton Korznikov, Andrey Galichin, Alexey Dontsov, Oleg Y. Rogov, Ivan Oseledets, and Elena Tutubalina. Sanity checks for sparse autoencoders: Do SAEs beat random baselines?, 2026. URL https://arxiv.org/abs/2602.14111.

Patrick Leask, Bart Bussmann, Michael Pearce, Joseph Bloom, Curt Tigges, Noura Al Moubayed, Lee Sharkey, and Neel Nanda. Sparse autoencoders do not find canonical units of analysis, 2025. URL https://arxiv.org/abs/2502.04878.

Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. Gemma scope: Open sparse autoencoders everywhere all at once on Gemma 2, 2024. URL https://arxiv.org/abs/2408.05147.

George Ma, Zhongyuan Liang, Irene Y. Chen, and Somayeh Sojoudi. Do sparse autoencoders identify reasoning features in language models?, 2026. URL https://arxiv.org/abs/2601.05679.

Henry B. Mann and Donald R. Whitney. On a test of whether one of two random variables is stochastically larger than the other. *The Annals of Mathematical Statistics*, 18(1):50–60, 1947. doi: 10.1214/aoms/1177730491. URL https://doi.org/10.1214/aoms/1177730491.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets, 2023. URL https://arxiv.org/abs/2310.06824.

Jordan F. McCann. Descriptive collision in sparse autoencoder auto-interpretability: When one explanation describes many features, 2026. URL https://arxiv.org/abs/2605.12874.

Mirage Modi, Jordan E. Krull, Donte Johnson, Xiaoying Wang, Timothy D. Gauntner, Mingjia Li, Hao Cheng, Anjun Ma, Ping Zhang, Daniel G. Stover, Zihai Li, and Qin Ma. Why large language models' clinical reasoning fails: Insights from explainable deep learning. *medRxiv*, 2026. doi: 10.64898/2026.01.26.26344845. URL https://www.medrxiv.org/content/10.64898/2026.01.26.26344845v1.

Jin Mu and Guanhua Chen. Making clinical language models auditable: Concept-guided fine-tuning for robust prediction, 2026. URL https://arxiv.org/abs/2608.27397.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. Explainable prediction of medical codes from clinical text, 2018. URL https://arxiv.org/abs/1802.05695.

Charles O'Neill, Mudith Jayasekara, and Max Kirkby. Resurrecting the salmon: Rethinking mechanistic interpretability with domain-specific sparse autoencoders, 2025. URL https://arxiv.org/abs/2508.09363.

Gonçalo Paulo, Alex Mallen, Caden Juang, and Nora Belrose. Automatically interpreting millions of features in large language models, 2024. URL https://arxiv.org/abs/2410.13928.

Kenny Peng, Rajiv Movva, Jon Kleinberg, Emma Pierson, and Nikhil Garg. Position: Use sparse autoencoders to discover unknowns, 2026. URL https://arxiv.org/abs/2506.23845. International Conference on Machine Learning (ICML) 2026.

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

Jeanine Romano, Jeffrey D. Kromrey, Jesse Coraggio, and Jeff Skowronek. Appropriate statistics for ordinal level data: Should we really be using t-test and Cohen's d for evaluating group differences on the NSSE and similar surveys? In *Annual Meeting of the Florida Association of Institutional Research*, 2006.

Chris Sainsbury, Feng Dong, and Andreas Karwath. Sparse autoencoder decomposition of clinical sequence model representations: Feature complexity, task specialisation, and mortality prediction, 2026. URL https://arxiv.org/abs/2605.04072.

Thomas Searle, Zina Ibrahim, and Richard Dobson. Experimental evaluation and development of a silver-standard for the MIMIC-III clinical coding dataset. In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, pages 76–85. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.bionlp-1.8. URL http://dx.doi.org/10.18653/v1/2020.bionlp-1.8.

Dong Shu, Xuansheng Wu, Haiyan Zhao, Daking Rai, Ziyu Yao, Ninghao Liu, and Mengnan Du. A survey on sparse autoencoders: Interpreting the internal mechanisms of large language models, 2025. URL https://arxiv.org/abs/2503.05613.

Robert F. Tate. Correlation between a discrete and a continuous variable. point-biserial correlation. *The Annals of Mathematical Statistics*, 25(3):603–607, 1954. doi: 10.1214/aoms/1177728730. URL https://doi.org/10.1214/aoms/1177728730.

Thanh Vu, Dat Quoc Nguyen, and Anthony Nguyen. A label attention model for ICD coding from clinical text. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20)*, pages 3335–3341, 2020. URL https://arxiv.org/abs/2007.06351.

Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945. doi: 10.2307/3001968.

Zhengxuan Wu, Aryaman Arora, Atticus Geiger, Zheng Wang, Jing Huang, Dan Jurafsky, Christopher D. Manning, and Christopher Potts. AxBench: Steering LLMs? even simple baselines outperform sparse autoencoders. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 67035–67080. PMLR, 2025. URL https://arxiv.org/abs/2501.17148.

Zheng Yuan, Chuanqi Tan, and Songfang Huang. Code synonyms do matter: Multiple synonyms matching network for automatic ICD coding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 808–814, Dublin, Ireland, May 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.acl-short.91.

A DATA AND PREPROCESSING

Table 4: Corpus statistics.

Notes	50,000
Tokens	15,172,037
Shards	312
Maximum sequence length	8,192
ICD-9 codes at prevalence $> 2\%$	46
Selection split (shards 0–30)	5,001 notes
Held-out audit split (shards 281–311)	4,911 notes

Sampling. Drawn from the MIMIC-IV train and validation splits only; the test split was never accessed. Notes shorter than 500 characters were dropped. Notes were stratified into four bands by the frequency of each note's rarest active top-50 ICD-9 code (quartile bins), with a per-band budget of 35 / 30 / 25 / 10 percent from very rare to common and deficits carried forward. Seed 42.

Extraction and centering. Post-block residual stream at layer 16 (`hidden_states[17]`), fp32 compute and fp16 storage, tokenizer truncation with special tokens. Mean-centering is two-pass: a float64 accumulator over all tokens, a float32 saved mean, fp16 centered output. Post-extraction checks cover shape, finiteness, non-zero values and inter-note diversity. Centering is required for the domain SAEs: without it, training produces a dominant DC-offset latent that captures the mean rather than variation.

Privacy. No note text and no protected health information appears in this paper or in any released artifact. All verification during development was structural: row counts, dtypes and character-length integers.

Compute. All heavy computation ran on cloud GPUs: about 9 GPU-hours for activation extraction, about 6 hours per SAE on an A100-40GB for training, about 9 hours per SAE on an L4 for grounding, and about 4 A100-hours per source for ablation across the three SAE arms and the three projected-direction arms, roughly 24 GPU-hours for that stage. The four judges of §5.3 were run as follows: Claude Sonnet 4.6, GPT-5-mini and Gemini 2.5 Flash through OpenRouter, and DeepSeek-V3 on locally hosted weights, in accordance with the MIMIC-IV data use agreement's restrictions on where note-derived text may be sent. The third and fourth judge runs alone total 8,640 judgments for the headline arms and 4,964 for the forced-binary and cross-chapter arms (Appendix K).

B SAE CONFIGURATION AND TRAINING

The decoder-norm constraint is implemented as gradient projection removing the component parallel to each decoder direction before the optimizer step, followed by a hard renormalization after it. Both steps are required; omitting either breaks the constraint. Checkpoint selection uses explained variance on the held-out eval shards, which are the same shards as the audit split of §3.1. We name this explicitly because it is an eval-split quantity and not a test-set quantity. Before each production run the activation scale was measured on centered activations and the sparsity penalty adjusted to target $L_0 \in [20, 80]$ on a 2,000-note calibration run. A latent is counted dead if its maximum activation over the held-out eval shards is below 10^{-6}, the threshold the vanilla SAE's resampling step uses.

Second training seed. We repeat the JumpReLU run at seed 2 on the same activations, with the architecture and sparsity settings above unchanged, and take the checkpoint at step 46,000. It reaches mean L_0 40.25 at explained variance 0.906 with a dead-latent fraction of 0.0285, against 40.92, 0.906 and 0.0262 at seed 42, so the reconstruction–sparsity operating point reproduces to within 0.7 in L_0 and to three decimal places in explained variance. The judge tables of Appendix K and the grounding tables of Appendices C and J carry the second seed as a separate row or column.

Table 5: SAE configuration. GemmaScope is applied zero-shot: it was trained on general-purpose web text, not on clinical activations, and is included to separate *learned* from *learned on this data*.

	JumpReLU (domain)	vanilla (domain)	GemmaScope (general)
Status	trained	trained	pretrained, zero-shot
Training seed	42 and 2	42	—
d_{in} / d_{sae} / expansion	2304 / 18,432 / 8×	2304 / 18,432 / 8×	2304 / 16,384 / —
Activation	JumpReLU	ReLU	JumpReLU
Sparsity penalty	L_0, $\lambda = 10$	L_1, $\ell_1 = 10$	(pretrained)
Bandwidth ε	1.0	—	—
`log_threshold_init`	-2.0	—	—
Learning rate	2e-4	2e-4	—
Adam (β_1, β_2)	$(0.0, 0.999)$	$(0.0, 0.999)$	—
LR warmup steps	2,000	2,000	—
Sparsity warmup steps	5,000 (λ_{L_0})	3,000 (ℓ_1)	—
Early stopping	—	patience 3 evals	—
Epochs	3 (best ckpt step 36,000)	3 (early-stopped, step 30,000)	—
Batch (tokens)	4,096	4,096	—
Dead-latent handling	self-correcting θ	resample at 5,000	—
Decoder constraint	unit norm	unit norm	as released
Checkpoint selection	held-out eval-shard EV	held-out eval-shard EV	—
Explained variance (held-out)	0.906	0.889	-4.21
Mean L_0 (held-out)	40.92	47.57	50.2
Dead-latent fraction (held-out)	0.0262	0.0001	0.0276

C Extended grounding results

Definitions. For every feature and code we compute the point-biserial correlation (Tate, 1954) between the note-level max-pooled feature value and the binary label,

$$r_{pb} = \frac{M_1 - M_0}{s}\sqrt{\frac{n_1 n_0}{N^2}},$$

with two-tailed p-values from the associated t-statistic and Benjamini–Hochberg correction at $q = 0.05$ over the full $k \times 46$ family (Benjamini and Hochberg, 1995). A feature is *grounded* at threshold τ if it has a BH-significant association with $|r_{pb}| > \tau$; its *monospecificity* is the number of such codes, and we report the fraction of grounded features associated with exactly one. *Leakage* is the off-target quantity: for a feature selected for code c, the mean $|r_{pb}|$ against the other 45 codes, computed on c-negative notes only so that genuine comorbidity is not scored as non-specificity, with a minimum of 10 positives per off-target code. We report leakage alongside the specificity ratio because the ratio carries the on-target correlation in its denominator and is therefore mechanically coupled to grounding strength, while leakage contains no on-target term. The length control residualizes each pooled value on note token count by ordinary least squares before recomputing the whole pipeline.

Table 6: Threshold sweep and monospecificity for the three SAEs on the held-out audit split ($n = 4{,}911$). "M/O/P" specificity counts grounded latents associated with exactly one code, two or three codes, and four or more. †JumpReLU SAE seed 2: a second training seed.

	JumpReLU			vanilla		GemmaScope		JumpReLU s2†	
$\lvert r_{pb}\rvert >$	grounded	mono	M/O/P	grounded	mono	grounded	mono	grounded	mono
0.1	9,721	0.299	2904/3245/3572	8,985	0.313	5,790	0.219	9,632	0.296
0.2	2,075	0.612	1270/645/160	2,063	0.619	295	0.759	2,119	0.615
0.3	610	0.793	484/117/9	675	0.816	54	0.870	638	0.796
0.4	276	0.884	244/31/1	291	0.904	13	0.846	286	0.895
0.5	147	0.925	136/11/0	143	0.944	4	1.000	137	0.949
0.6	61	1.000	61/0/0	73	1.000	0	—	63	1.000
0.7	28	1.000	28/0/0	29	1.000	0	—	32	1.000

The calibrated nulls behave the same way. At $|r_{pb}| > 0.1$ the L_0-matched random arm is monospecific for 0.354 of its grounded directions and PCA for 0.488, both above all three SAEs, and above $|r_{pb}| = 0.3$ each reaches 1.000 on the single direction it has left. Monospecificity rises with the

Table 7: The same sweep after residualizing each pooled activation on note token count by OLS and re-applying BH correction. Held-out split. †As in the previous table.

	JumpReLU		vanilla		GemmaScope		JumpReLU s2†	
$\lvert r_{pb}\rvert >$	grounded	mono	grounded	mono	grounded	mono	grounded	mono
0.1	5,711	0.468	5,355	0.478	1,039	0.615	5,649	0.458
0.2	1,276	0.695	1,335	0.705	125	0.808	1,294	0.689
0.3	451	0.792	503	0.827	39	0.897	476	0.813
0.4	236	0.881	257	0.922	12	0.917	238	0.903
0.5	124	0.944	131	0.962	3	1.000	124	0.960

threshold for every source, nulls included, because thresholding selects features sitting just above the bar.

Length residualization removes 41.3% of the JumpReLU SAE's grounded latents at $|r_{pb}| > 0.1$ (9,721 to 5,711) and 40.4% of the vanilla SAE's, but only 15.6% at $|r_{pb}| > 0.5$ (147 to 124; vanilla 143 to 131, 8.4%). The confound is concentrated in weakly grounded latents. The top correlation falls from 0.8643 to 0.8550. GemmaScope loses 82.1% of its grounded latents (5,790 to 1,039) and its mean maximum $|r_{pb}|$ falls from 0.0894 to 0.0556, a 37.8% drop, so a larger share of its apparent grounding was length.

The two domain SAEs independently surface the same conditions at the top of the ranking, which suggests the pattern reflects the base model's representation of clinical text rather than an artifact of one dictionary. A shared top-code list is not by itself evidence of that, however. The random arm's ten strongest associations concentrate on two high-prevalence codes, 428.0 (congestive heart failure, seven of ten) and 427.31 (atrial fibrillation, three of ten), so agreement at the top of a ranking can reflect prevalence and note structure rather than anything about the decomposition. What distinguishes the SAEs is the strength at which the agreement holds, not the agreement.

D AUTO-INTERPRETABILITY: PROMPTS, CATEGORIZATION AND SCORERS

Tiers. 964 features were explained and scored: 276 strongly grounded ($|r_{pb}| > 0.4$), 100 weakly grounded ($0.1 < |r_{pb}| \leq 0.3$), 484 non-grounded and 100 dead. The non-grounded tier is smaller than the 1,000 planned because the pipeline was rate-limited.

Categorization. The explainer assigns each feature to one of five categories. Strongly grounded features are 47.1% clinical concept and 42.9% clinical vocabulary; non-grounded features are 99.8% noise; dead features are 75% structural pattern. Weakly grounded features sit between, at 14% clinical concept and 39% clinical vocabulary.

Judges and bands. Four judges from four laboratories score every arm: Claude Sonnet 4.6 (Anthropic), DeepSeek-V3, GPT-5-mini (OpenAI) and Gemini 2.5 Flash (Google). A single model writes every explanation, from top-activating contexts together with non-activating ones, each context a 30-token window either side of the marked trigger token. Because hit@1 rises steeply with $|r_{pb}|$ and the sources do not populate the bands equally, every comparison is made *within* an $|r_{pb}|$ band and no pooled figure is reported.

Scorers. Fuzzing and detection reach 0.938 and 0.962 globally with no tier separation (strong 0.938/0.960, weak 0.941/0.956, dead 0.932/0.975). The shuffled-explanation control re-scores each feature against a deranged explanation, under both a global derangement and a within-tier permutation. Global: fuzzing 0.932 real against 0.496 shuffled (95% CI $[0.489, 0.502]$, $n = 274$), detection 0.961 against 0.522 ($n = 285$); paired Wilcoxon $p < 10^{-16}$ in both. The scorers work; they do not discriminate between tiers.

Prompt variants. All four judge prompts are reproduced verbatim below. The reported protocol uses the de-anchored concordance prompt and the blind forced-choice retrieval prompt; the anchored and forced-binary variants exist to measure what the two conventional choices (stating the correlation, and offering PARTIAL) do to the result. The explainer prompt and the fuzzing and detection scorer prompts follow Paulo et al. (2024) and are omitted for space.

Arm 0, anchored concordance (the conventional prompt, measured but not reported as a result).

```
A sparse autoencoder feature has been auto-interpreted as:
"{explanation}"
This feature also has the strongest statistical correlation
(point-biserial r = {r_pb}) with the ICD-9 diagnosis code:
{icd_code} ({icd_description}).
Does the auto-interp explanation describe the same concept
as the ICD code?  Respond with:
- YES -- the explanation clearly describes the ICD concept
- PARTIAL -- the explanation overlaps but describes a
related, broader, or narrower concept (e.g., a drug used
to treat the condition, a symptom of the condition, or a
broader category containing the condition)
- NO -- the explanation describes something unrelated
Format:  <verdict> | <one-sentence rationale>
```

Arm 1, de-anchored (the reported concordance prompt). Identical, with the correlation sentence deleted and a PARTIAL subtype requested.

```
A sparse autoencoder feature has been auto-interpreted as:
"{explanation}"
Consider the ICD-9 diagnosis code {code} ({description}).
Does the explanation describe the same clinical concept as
the code?
- YES -- the explanation clearly describes the code's
concept
- PARTIAL -- related but not identical; also name the
subtype:  one of treatment, symptom, broader-category,
narrower-concept, related-other
- NO -- unrelated
Format for YES/NO: <verdict> | <one-sentence rationale>
Format for PARTIAL: PARTIAL | <subtype> | <one-sentence
rationale>
```

Forced binary. PARTIAL removed from the option set.

```
[...as arm 1 ...]  You MUST answer YES or NO. There is no
partial option:  if the explanation describes a related but
different concept (a treatment for the condition, a symptom
of it, a broader category containing it, or a neighbouring
diagnosis), that is NO.
Format:  <YES or NO> | <one-sentence rationale>
```

Blind forced-choice retrieval. The judge never sees the code the feature correlates with, only the explanation and a shuffled slate.

```
A sparse autoencoder feature has been auto-interpreted as:
"{explanation}"
Which ONE of these ICD-9 codes does the explanation best
describe?  If none fit, choose the 'none' option.  Answer
with one letter, then a rationale.
{options}
Format:  <letter> | <one-sentence rationale>
```

Slate construction. A slate holds up to nine options: the five codes with the largest $|r_{pb}|$ for that feature (the first of which is the true code by construction), three hard negatives drawn from the codes with the smallest $|r_{pb}|$, and an explicit `none of these`. Options are shuffled per feature with a seeded RNG and lettered. The design floor is therefore 1/9 = 11.1%. Under the same-chapter rule the 46-code panel cannot always supply seven distractors from the true code's own chapter. For the JumpReLU pool, 66% of slates hold all nine options and the rest hold six (93 features), four (52), three (7) or two (4); the mean per-feature floor is 14.5% (vanilla 14.9%, seed-2 JumpReLU 14.2%).

Every separation reported in §5.3 is tested against the matched random arm rather than against this combinatorial floor, so the reported comparisons do not depend on it. The two conditions differ only in the chapter admissibility rule applied to distractors: the cross-chapter slate requires each distractor to lie in a *different* ICD-9 chapter from the true code, and the hard-negative slate requires the *same* chapter.

E CAUSAL ABLATION

Protocol, SAE arms. For each target (j, c), and for each held-out note, we record cross-entropy loss over the note's final 25% of non-padding tokens under three conditions: a clean forward pass, a pass with the SAE reconstruction spliced into the layer-16 residual stream, and a pass with feature j set to its dataset mean before decoding. Writing $\bar{z}_j$ for that mean, mean-ablation replaces the residual stream $\hat{x}$ with

$$\hat{x}' = \hat{x} + (\bar{z}_j - z_j)\, W_{\text{dec},j},$$

so the feature is held at its average level rather than removed from the distribution entirely. Layers 17 onward are re-run from $\hat{x}'$. The reported effect is $\ell_{\text{abl}} - \ell_{\text{recon}}$, baselined on the reconstructed pass so that the decomposition's own distortion is held fixed and only the feature's contribution varies; the reconstruction tax $\ell_{\text{recon}} - \ell_{\text{clean}}$ is recorded alongside. For GemmaScope, which is applied to uncentered activations, $\mu = 0$ and the centering step is skipped.

Protocol, direction arms. The SAE protocol cannot be reused for a bare direction, which has no dictionary to reconstruct from. A pseudo-SAE built from 18,432 random directions produces $\max|\hat{x}| = 2.1 \times 10^6$ against the fp16 ceiling of 65,504, and the same construction on 46 difference-in-means directions carries a 7.47-nat reconstruction tax, 4.6× the base loss. Direction arms are therefore ablated by projection on the raw residual stream (Arditi et al., 2024),

$$x' = x - (x \cdot d)\, d,$$

with $\ell_{\text{abl}} - \ell_{\text{clean}}$ as the effect and no reconstruction tax by construction. Absolute nats are consequently not comparable across the reconstruction/projection line; Cliff's δ is a within-arm rank contrast and is comparable across it.

Targets and statistics. Each source contributes its ten most strongly grounded (feature, code) pairs, filtered by monospecificity and firing density, plus one random and one low-$|r_{pb}|$ control. For each target the ablation effects on code-positive and code-negative notes are compared by a one-sided Mann–Whitney U test (Mann and Whitney, 1947); effect size is Cliff's δ (Cliff, 1993), binned following Romano et al. (2006); and Benjamini–Hochberg at $q = 0.05$ is applied within each source's twelve-target family, so no arm inherits another's correction.

Table 8: Full ablation grid: each source's top-10 grounded targets plus two controls, on the 4,911 held-out notes. δ is the median Cliff's δ; BH-sig counts targets significant at $q = 0.05$ within each 12-target family; off-sig is over $10 \times 45 = 450$ tests; the last column repeats both after residualizing each note's effect on length. SAEs are mean-ablated, directions projected out; see text.

				specificity			length-adj.
source	$\lvert r\rvert$ range	δ	BH-sig	on-tgt Δ	ratio	off-sig	δ (sig)
vanilla SAE	0.822–0.860	**0.312**	**10/10**	0.357	**12.44**×	**0/450**	0.311 (10/10)
JumpReLU SAE	0.799–0.864	**0.276**	**10/10**	0.246	**11.10**×	4/450	0.191 (6/10)
GemmaScope SAE	0.428–0.545	0.195	7/10	0.221	4.58×	22/450	0.166 (6/10)
difference-in-means	0.386–0.699	0.055	4/10	0.045	1.17×	19/450	0.055 (4/10)
PCA	0.220–0.425	−0.051	2/10	−0.009	−0.75×	11/450	+0.028 (3/10)
random-matched	0.309–0.432	−0.096	0/10	−0.090	−1.53×	7/450	−0.005 (0/10)

The causal comparison is best-versus-best Each arm contributes its own ten most strongly grounded targets, and the domain SAEs' targets (0.799–0.864) lie above every other arm's, so their lead in Table 8 cannot be separated from their lead in Table 1 and the $|r|$ range column is part of the result rather than context for it. This is a consequence of the supply asymmetry rather than a choice: a matched-strength ablation would require competitors at 0.8, and none exist.

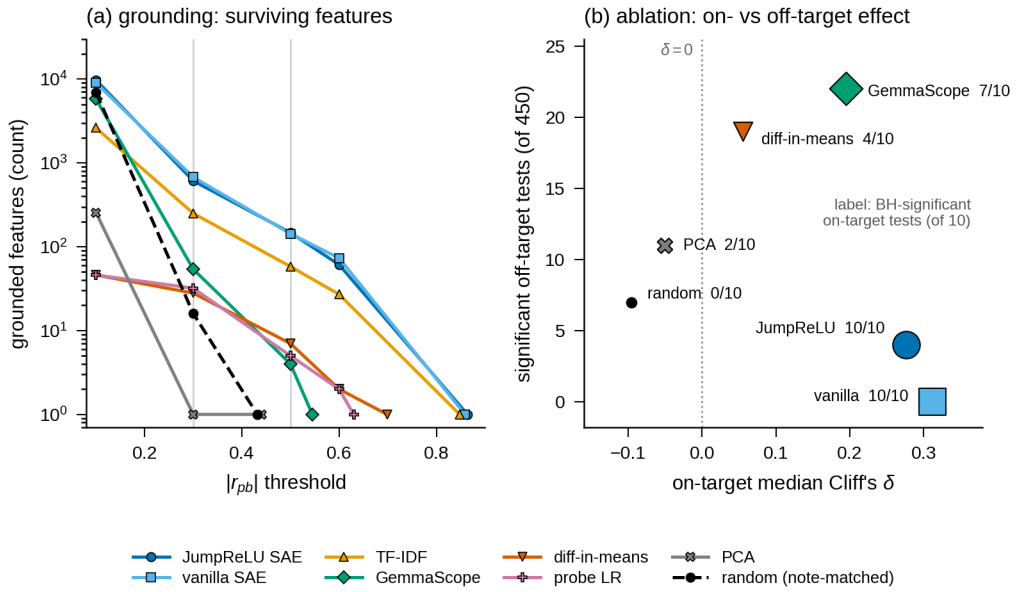


Figure 3: Two instruments, one ordering. **(a)** Grounded-feature count against $|r_{pb}|$ threshold on the held-out split, log scale; each curve ends at its peak. The grey guides mark the band edges used in Figure 2. **(b)** Diagnosis-conditioned causal relevance for the six sources that admit a model-internal intervention: scatterplot of median on-target Cliff's δ over each source's ten most strongly grounded targets against the number of BH-significant off-target tests out of 450.

The random arm calibrates the Cliff's δ floor. A previously used threshold of $\delta^*_{95} = 0.0732$ was derived from off-target (feature, wrong-code) pairs. The random-matched arm supplies the measured equivalent from a source that is null by construction, and 0.0732 sits near the 25th percentile of its $|\delta|$ distribution rather than the 95th. No reported claim moves, since the arm returns 0 of 12 significant targets under BH, but "δ above δ^*_{95}" is not a usable stand-in for exceeding a noise floor. BH significance is the load-bearing criterion throughout.

Two qualifications. First, the effects are small in absolute terms. Vanilla's mean on-target effect is 0.00202 nats, 0.124% of base loss and 0.069× the reconstruction tax the decomposition itself introduces; holding one latent at its dataset mean inside an otherwise unchanged reconstruction is not expected to move loss much on any single note, and what the test asks is only whether it moves more on code-positive ones. Second, length adjustment does not treat the two domain SAEs alike: it leaves vanilla unchanged (0.311, ten of ten) and costs JumpReLU more (0.191, six of ten), whose low-$|r_{pb}|$ control is itself significant where vanilla's is not. The architecture-independent form of the result rests on vanilla here, as it rests on JumpReLU in §5.3.

The difference-in-means controls are not a null. At $k = 46$ every direction in this source is built from some code's labels, so the arm has no uncorrelated direction available to serve as a control. Its two control targets are its lowest-$|r_{pb}|$ directions, at 0.198 and 0.194, and 21 of the 46 directions clear the target screen. Its control median of -0.016 should therefore be read as the effect of a weakly grounded supervised direction, not as a null distribution. The random-matched arm supplies the null for the other direction sources.

F Baseline construction

Covariance-matched random directions. The activation covariance Σ is estimated in one pass over train shards only, with a float64 accumulator, and never over the audit split, so the directions are not shaped by the notes they are later tested on. We draw 18,432 directions from $\mathcal{N}(0, \Sigma)$ by eigendecomposition, clipping the negative eigenvalues that fp16 accumulation produces, and normalize to unit length. Because $\max_t f(x_t) = f(\max_t x_t)$ for the threshold function $f(x) =$

$x \cdot \mathbb{I}[x > \tau]$, thresholding commutes with max-pooling, so we store un-thresholded pooled values once and derive every sparsity arm afterwards at no further cost. This commutation does *not* hold for mean- or top-k-pooling, so the projection refuses any pooling other than max. Three arms were produced: two arms whose per-direction thresholds are quantile-calibrated to reproduce the token-level mean L_0 of the JumpReLU SAE (40.92) and the vanilla SAE (47.57); and the **note-matched** arm, which calibrates each direction's threshold to reproduce the JumpReLU SAE's per-feature *note-level* firing distribution. The body reports the note-matched arm alone. Token-level L_0 matching does not control the quantity the grounding test consumes: after max-pooling over thousands of tokens, note-level density is 0.95 under token-L_0 matching and 1.00 dense, against 0.656 for the note-matched arm and 0.675 for the SAE it is matched to. The note-matched arm is also the stronger null on the axis that matters, with the highest peak $|r_{pb}|$ of the random arms (0.432).

PCA. The 2,304 principal components of the same centered activations, used as latents with the transform matrix as encoder, following the SAEBench baseline (Karvonen et al., 2025).

Difference-in-means directions. Built on train shards 31–280 ($n = 40{,}088$ notes), audited on 281–311, selected on 0–30. For each code, $d_c \propto M^{-1}(\bar{x}_{y_c=1} - \bar{x}_{y_c=0})$, normalized to unit length, with three choices of M: the identity (plain), $\mathrm{diag}(\Sigma)$ (diagonal z-scoring) and Σ under Ledoit–Wolf shrinkage (LDA-whitened). Shrinkage is 0.00097 and reduces the condition number from 6.9×10^{16} to 3.7×10^{5}. The whitened arm is the mass-mean probe direction of Marks and Tegmark (2023).

Logistic-probe directions. Per-code L_2 logistic regression on the same pooled train activations, with the coefficient vector normalized as the direction. Two arms differ only in whether class weights are balanced.

TF-IDF. 10,000 unigram and bigram features with sublinear term frequency, fitted on the matched note texts. Two arms use binary presence and TF-IDF value. Each term is a feature, and the same `top_per_code` selection rule is applied, so TF-IDF receives the same best-of-k search the SAE receives.

Lexical keyword baseline. Binary keyword indicators from a curated 64-code dictionary; word-boundary regex for keywords of three characters or fewer, case-insensitive substring otherwise. The SAE's best latent beats the keyword indicator on 45 of 46 codes for the JumpReLU SAE (mean $\Delta r = 0.287$) and 44 of 46 for the vanilla SAE (0.308). GemmaScope wins on only 25 of 46 and loses on 12 (mean $\Delta r = 0.034$).

Classification baselines. Validated logistic regression over the corpus, scored by AUC-ROC and AUC-PR. On the raw pooled 2,304-dimensional activations the mean AUC-ROC is 0.808. On the 18,432 SAE features it is 0.881. On TF-IDF it is 0.917, so the SAE does not add value over supervised text features: TF-IDF wins 33 of 46 codes, the SAE wins none, and the paired Wilcoxon signed-rank test over per-code differences gives a median Δ of -0.033 with $p < 10^{-8}$ (AUC-PR: 0.590 against 0.532, median Δ -0.059). This is the prediction half of the dissociation and it runs against the SAE.

Reconciling the two TF-IDF comparisons. Two different statistics compare the SAE and TF-IDF at feature level and they should not be read as contradicting each other. The supplementary comparison inside the classification run takes each code's single best feature by $|r_{pb}|$ over the full corpus with no selection split, and gives a mean best $|r_{pb}|$ of 0.579 for the SAE against 0.519 for TF-IDF, with the SAE ahead on 23 of 46 codes, TF-IDF ahead on 1, and 22 comparable within tolerance. Table 1 instead reports the *median* selected $|r_{pb}|$ on the held-out split with selection performed on a disjoint split, giving 0.574 against 0.518–0.531. Mean against median, full corpus against held-out, and no selection split against a disjoint one. Both say the same thing: the SAE's best single feature per code is slightly stronger than TF-IDF's, and the margin is small.

G GemmaScope domain-shift diagnostics

GemmaScope is a JumpReLU SAE of the same architecture as our domain JumpReLU SAE, trained on general-purpose web text rather than on clinical activations. Applied zero-shot to clinical activations it has an explained variance of -4.21, meaning its reconstruction is worse than predicting the

mean. We ran three variants to separate a decoder-bias shift from a feature-direction mismatch, on a 2.5M-token sample; the -4.21 above is the full-corpus figure of Appendix B.

Explained variance, as published	-4.20
Explained variance, clinical mean substituted for decoder bias	-4.20
Explained variance, fully recentered activations	-6.56
Fraction of pre-activations above threshold	0.711
cos(decoder bias, mean clinical activation)	0.90

Recentering does not recover explained variance and makes it worse, which rules out a decoder-bias shift as the explanation. That is consistent with the dictionary's feature directions not spanning the clinical activation subspace, but these diagnostics test a mean shift and do not by themselves establish a direction mismatch, so we state the weaker claim. The 0.711 above-threshold fraction, against a nominal average L_0 near 42, indicates the SAE is firing far more densely than it was calibrated to. In loss units the same mismatch appears as a reconstruction tax of 0.648 nats when its reconstruction is spliced into the layer-16 residual stream, against 0.029 for the vanilla SAE and 0.009 for the domain JumpReLU SAE, a 22-fold difference on the quantity the ablation protocol of §4.4 baselines against.

H Feature inspection

For each strongly grounded latent we scan the activation shards for its top-activating tokens, retokenize the matched notes for context, and record the firing rate on code-positive against code-negative notes together with a lexical diversity score over the top tokens. Only token strings are reported, never note text. The per-latent table gives the feature index, its top code, $|r_{pb}|$, top-k tokens, the positive-to-negative firing-rate ratio and the diversity score.

I Data and code availability

Code will be released. MIMIC-IV is obtained through PhysioNet credentialing and is not redistributable, so we release the pipeline rather than the corpus. Released artifacts are aggregate summaries: the per-source `audit_summary.json` files behind Table 1 and Table 10, the necessity-suite comparison summary, the ablation summaries and post-hoc specificity files, and the per-feature judge verdict records behind Table 3 and Table 13. The verdict records were scanned for de-identification markers, dates, names and record numbers, and the verbatim prompt and raw judge output columns are stripped before release because a prompt embeds the activation contexts it was built from. A single script regenerates every judge-side number in §5.3 from those per-feature records.

J The audit path: configuration and full tables

Audit configuration, held constant across all fourteen audited arms. r threshold 0.1 for both the grounded bar and the off-target significance bar, so no method can be flattered by a laxer bar on one axis than the other; Benjamini–Hochberg at $q = 0.05$ applied to the full $k \times 46$ grounding matrix and separately per selected feature across its off-target codes; `min_off_pos` 10 and `min_pool` 10; off-target restricted to code-negative notes, with the all-notes variant computed alongside as a cross-check; selection on shards 0–30 and audit on shards 281–311, with `in_sample_selection: false` recorded in every source's summary.

Off-target specificity, both variants. The c-negative restriction matters. On all notes the median specificity ratio is 13.6–13.7 for the TF-IDF arms against 18.0–17.0 under the c-negative restriction, because co-occurring codes inflate the apparent off-target correlation when code-positive notes are left in the pool. We report the c-negative variant throughout and treat the all-notes numbers as the confounded cross-check.

Selection-bias check, per source. Median selected $|r_{pb}|$ on the selection split against the audit split, as a percentage change: diff-in-means diagonal -3.3, TF-IDF binary $+2.4$, PCA $+2.1$, diff-in-means

Table 9: Grounded-feature count by threshold, all thirteen audited arms, held-out audit split. Counts are over each source's full dictionary and are bounded above by k. A feature counts as grounded at τ if it has a BH-significant correlation above τ with any of the 46 codes, whereas the judge bands of §5.3 bin each feature by its correlation with its own code. The two nearly coincide for the SAEs, whose judge pool is a sample of the dictionary rather than all of it, and diverge for the 46-direction sources, where a direction built for one code often correlates more strongly with a comorbid one. This is the table behind the claim that the columns which separate sources begin at $|r_{pb}| > 0.3$. The note-matched random arm's summary artifact reports only the four thresholds used in Table 1; "n/a" marks the two it does not carry.

source	k	peak $\lvert r_{pb}\rvert$	> 0.1	> 0.2	> 0.3	> 0.4	> 0.5
JumpReLU SAE	18,432	0.864	9,721	2,075	610	276	147
JumpReLU SAE, seed 2[†]	18,432	0.874	9,632	2,119	638	286	137
vanilla SAE	18,432	0.859	8,985	2,063	675	291	143
TF-IDF (binary)	10,000	0.831	2,879	740	265	114	60
TF-IDF (value)	10,000	0.848	2,614	743	250	115	58
GemmaScope SAE	16,384	0.545	5,790	295	54	13	4
diff-in-means (LDA)	46	0.699	46	43	28	19	7
probe LR (unweighted)	46	0.637	46	46	31	19	5
probe LR (balanced)	46	0.630	46	46	32	21	5
PCA	2,304	0.441	256	18	1	1	0
random (note-matched)	18,432	0.432	6,945	n/a	16	n/a	0
random (L_0-matched)	18,432	0.314	9,132	127	1	0	0
diff-in-means (diagonal)	46	0.310	46	41	3	0	0
diff-in-means (plain)	46	0.291	46	42	0	0	0

Table 10: Per-source medians over the 46 selected features, held-out split. Specificity ratio is on-target $|r_{pb}|$ divided by mean off-target $|r_{pb}|$; $n_{\text{off sig}}$ is the median count of BH-significant off-target codes per selected feature. [†]A second training seed of the JumpReLU SAE (preliminary).

source	spec. ratio	$n_{\text{off sig}}$	leakage
TF-IDF (binary)	18.02	1.0	0.0273
TF-IDF (value)	17.02	0.5	0.0251
vanilla SAE	15.88	2.0	0.0324
JumpReLU SAE, seed 2[†]	15.08	2.5	0.0351
JumpReLU SAE	14.94	3.0	0.0373
GemmaScope SAE	6.29	4.0	0.0468
diff-in-means (LDA)	5.68	8.0	0.0572
probe LR (unweighted)	4.42	17.0	0.0885
probe LR (balanced)	3.67	16.5	0.0921
random (note-matched)	3.07	6.0	0.0558
PCA	2.88	4.0	0.0484
random (L_0-matched)	2.12	12.0	0.0751
diff-in-means (diagonal)	1.37	17.0	0.0938
diff-in-means (plain)	1.25	17.0	0.0940

plain $+2.1$, TF-IDF value -1.8, diff-in-means LDA -1.8, random L_0-matched $+1.6$, JumpReLU SAE -1.5, random note-matched -1.1, probe LR balanced -0.8, GemmaScope $+0.8$, probe LR unweighted $+0.4$, vanilla SAE -0.1. Six of thirteen rise out-of-sample. The five 46-direction arms use `identity` selection and make no selection at all, so their shifts (-3.3 to $+2.1$) measure split noise alone; the four arms searching 18,432 candidates per code shift by -1.5 to $+1.6$, inside that range. Best-of-k search leaves no bias distinguishable from split noise, and it is not what produces the SAEs' lead: JumpReLU's -1.5% is 0.009 in $|r_{pb}|$, against a lead of 0.235 over the strongest non-SAE source in activation space.

Matched-$|r_{pb}|$ control. Pairing codes on which two sources reach on-target correlations within 0.05 leaves 5 codes for the vanilla SAE against LDA and against the probe, 6 for the JumpReLU SAE against the probe, 15 for GemmaScope against LDA, 14 for GemmaScope against the probe, and *none at all* against PCA or the random arms, which never reach comparable strength on any code. The domain-SAE comparisons point the right way but are inconclusive at those sample sizes:

the SAE has lower leakage on 2 of 5, 4 of 5 and 5 of 6 matched codes with two-sided Wilcoxon $p \geq 0.31$. GemmaScope is the only SAE with an estimable control, and it has lower leakage on 13 of 14 matched codes against the unweighted probe (median 0.036 against 0.074, $p = 0.0017$). The decisive result is the one that needs no test: PCA and random directions never reach a comparable $|r_{pb}|$ at all.

Coupling control, per method. Spearman ρ between on-target $|r_{pb}|$ and leakage across the 46 codes of each source: probe LR unweighted $+0.43$ ($p = 0.003$), random note-matched $+0.42$ ($p = 0.004$), diff-in-means LDA $+0.39$ ($p = 0.008$), probe LR balanced $+0.29$ ($p = 0.050$), random L_0-matched $+0.15$, PCA $+0.10$, diff-in-means diagonal $+0.03$, diff-in-means plain -0.11, JumpReLU SAE -0.16, GemmaScope -0.22, vanilla SAE -0.22 (last three all $p > 0.14$), TF-IDF value -0.27 ($p = 0.075$), TF-IDF binary -0.34 ($p = 0.019$). Pooled over the 736 (method, code) points of the activation-space arms (TF-IDF excluded), the OLS slope of leakage on on-target strength is -0.0309 ($r = -0.191$, $p = 1.8 \times 10^{-7}$). Median leakage residuals under that fit: vanilla SAE -0.0284, PCA -0.0252, JumpReLU SAE -0.0240, GemmaScope -0.0239, diff-in-means LDA -0.0137, random L_0-matched $+0.0029$, diff-in-means diagonal and plain $+0.0183$, probe LR unweighted $+0.0195$, probe LR balanced $+0.0250$. After controlling for on-target strength, the unsupervised bases leak less than their strength predicts and the label-fitted directions leak more; low leakage is shared by the SAEs with PCA, and TF-IDF, outside this fit, has the lowest raw leakage of any source (Table 10).

K JUDGE PROTOCOL: FULL GRIDS AND VALIDITY CHECKS

Pools and bands. Every judge-side number is reported within a disjoint $|r_{pb}|$ band, because hit@1 rises steeply with $|r_{pb}|$ and the sources do not occupy the bands equally, so any pooled figure is a composition artifact. The SAE rows merge the published 380-feature explanation pool with a stratified sampling pool built for band coverage, deduplicated by feature index; the two use identical configurations and different feature draws. Every non-SAE source contributes its own features in each band, and n is reported in every cell because grounding scarcity is itself one of the results: GemmaScope has 54 held-out features above $|r_{pb}| = 0.3$ and 4 above 0.5 in total, so its empty cells are complete populations rather than samples, and reporting a wide interval there would misrepresent a hard scarcity as sampling noise.

The judge is not the weak link. Four judges from four laboratories order the sources identically, and on three of the four domain-SAE cells above $|r_{pb}| = 0.3$ they agree within 5 points. In two places the choice still matters. The vanilla SAE in the middle band draws an 11-point spread and is the same cell where its separation from the random floor fails BH correction under three of the four judges, while JumpReLU survives for three (Table 11); the architecture-independent form of this result therefore rests on JumpReLU, with vanilla directionally consistent but not uniformly significant. And exact-YES spans 18 points on the JumpReLU cell above 0.5 where forced choice spans 3, so where the two instruments disagree we prefer forced choice and report exact-YES as the stricter corroboration it is. No non-SAE learned source separates from the random floor under any judge.

Separation from the random floor. Each source is tested against the matched random arm in the one band where every source has a testable population.

Recall rather than recognition. The strict endorsement grid behind §5.3. The judge is shown the code and must endorse it outright, so there is no slate to guess from and the forced-choice floor disappears.

Judge agreement. Under same-chapter forced choice the judge largely stops mattering. Across the SAE cells of Table 3 the four judges span at most 5 percentage points on three of the four: JumpReLU 73/73/75/70 and 92/92/94/91, vanilla 94/94/96/96, keyword 92/96/96/100, GemmaScope 52/46/52/50. The exception is the vanilla SAE in the middle band, 67/71/73/62, where Gemini's 62% opens an 11-point spread, the same cell that misses significance in Table 11 above, so the two observations are one finding rather than two. The seed-2 JumpReLU SAE (71/74/76/66 and 95/92/95/87) repeats the pattern: Gemini is again the low judge, opening spreads of 10 and 8 points, and again the judge under which its middle-band test fails correction. Compare the 18-point spread that exact-YES still

Table 11: Fisher exact, two-sided, in the $|r_{pb}| \in [0.3, 0.5)$ band. Fisher rather than χ^2 because the random arm holds 16 directions there. Each judge is tested against *its own* floor: the random arm scores 38% under S, G and M but 44% under D, so D is the least powerful column rather than the most. The final column applies Benjamini–Hochberg at $q = 0.05$ across the six tests within a judge; the seed-2 row replaces the seed-42 JumpReLU row in that family rather than adding a seventh test. seed-2 †JumpReLU SAE : a second training seed.

source	hit@1 (S / D / G / M)	p (S)	p (D)	p (G)	p (M)	survives BH
keyword *(ceiling)*	92 / 96 / 96 / 100	**0.0002**	**0.0002**	**<0.0001**	**<0.0001**	all four
JumpReLU SAE	73 / 73 / 75 / 70	**0.0074**	0.0204	**0.0030**	**0.0125**	S, G, M
JumpReLU SAE, seed 2†	71 / 74 / 76 / 66	**0.0105**	0.0167	**0.0024**	0.0312	S, G
vanilla SAE	67 / 71 / 73 / 62	0.0266	0.0441	**0.0076**	0.0681	G only
GemmaScope SAE	52 / 46 / 52 / 50	0.394	1.000	0.394	0.407	none
diff-in-means (LDA)	24 / 38 / 29 / 29	0.475	0.749	0.726	0.726	none
probe LR (balanced)	17 / 17 / 17 / 17	0.159	0.080	0.159	0.159	none

Table 12: Exact-YES concordance: the judge is shown the code and must endorse it outright, with PARTIAL counted as a non-endorsement. This is recall where Table 3 is recognition: the judge must produce the code unaided rather than pick it from nine options, so the two tables bracket the measurement rather than repeat it. Cells are percent, n shared across the four judges; a dash means fewer than five features exist. †JumpReLU SAE seed 2: a second training seed.

	$\|r_{pb}\| \in [0.1, 0.3)$					$[\mathbf{0.3}, \mathbf{0.5})$					$> \mathbf{0.5}$				
source	n	S	D	G	M	n	S	D	G	M	n	S	D	G	M
keyword *(ceiling)*	6	67	50	67	50	26	85	77	96	92	6	100	83	100	83
JumpReLU SAE	139	0	0	0	0	175	15	19	21	19	144	**43**	**42**	**60**	**51**
JumpReLU SAE, seed 2†	139	0	0	1	0	184	12	19	24	19	137	**49**	**53**	**65**	**50**
vanilla SAE	126	0	0	2	0	177	12	15	23	14	143	**47**	**53**	**62**	**50**
GemmaScope SAE	100	0	0	0	0	50	6	10	12	12	—	—	—	—	—
diff-in-means (LDA)	18	6	11	11	6	21	19	29	24	19	7	14	29	29	29
probe LR (balanced)	17	12	12	12	12	24	12	12	17	12	5	20	20	20	20
PCA	34	0	3	3	3	—	—	—	—	—	—	—	—	—	—
random (note-matched)	284	2	2	3	2	16	0	0	0	0	—	—	—	—	—

shows on one cell and the 10.8-point spread the pooled metric produced under an earlier cross-chapter protocol. Small-n sources are a further exception and should not be read as judge disagreement: probe LR above $|r_{pb}| = 0.5$ spans 0–40% on $n = 5$.

Anchoring. The conventional concordance prompt states the feature's correlation value. Re-judging only the features previously marked YES with that sentence removed retains 88% of Sonnet 4.6's verdicts, 73% of Gemini 2.5 Flash's, 58% of GPT-5-mini's and 45% of DeepSeek-V3's on the JumpReLU SAE ($n = 85$), and 85% / 52% / 57% / 35% on the vanilla SAE ($n = 88$), and 85% / 57% / 59% / 48% on the seed-2 JumpReLU SAE ($n = 87$). Anchor-dependence is a continuum across judges rather than one outlier. Two consequences: an anchor-free prompt is the safer default for any new judge, which is what we use throughout, and a single-judge anchored concordance rate is not yet a measurement.

Cross-chapter against same-chapter, per band and judge. Change in hit@1 in percentage points when distractors are redrawn from the true code's own chapter, on all four judges. Negative means the source loses ground when the chapter-level shortcut is removed. The published 380-feature JumpReLU pool is reported separately from the stratified pool because it is the only row with a population large enough to read. †JumpReLU SAE seed 2: a second training seed.

Read the $n = 136$ row and treat the rest as underpowered. There the JumpReLU SAE loses 8.8–11.0 points under every judge, a four-way agreement that the same-chapter slate is genuinely the harder test. Every other row rests on $n = 10$–40, where a single feature moves the figure by 2.5–10 points, which is why the stratified rows disagree across judges (-22.5 to 0.0) while the 136-feature row does not. The SAE losing the most is the expected direction: it is the only non-ceiling source scoring

Table 13: The pooled YES+PARTIAL verdict, the metric most concordance results are reported on. Random directions outscore both domain SAEs in the lowest band under all four judges, and the judges span 57–88% on identical features purely on how generously each reads "partial". This table is the negative control, not a result. †JumpReLU SAE seed 2: a second training seed.

	$[0.1, 0.3)$					$[0.3, 0.5)$					> 0.5				
source	n	S	D	G	M	n	S	D	G	M	n	S	D	G	M
keyword *(ceiling)*	6	100	100	100	100	26	100	100	100	100	6	100	100	100	100
JumpReLU SAE	139	65	88	57	59	175	92	98	94	92	147	99	100	99	99
JumpReLU SAE, seed 2†	139	52	89	52	44	184	95	99	96	94	137	98	100	100	98
vanilla SAE	126	63	89	56	56	177	96	99	95	95	143	98	100	100	98
GemmaScope SAE	100	24	72	18	19	50	84	92	80	90	—	—	—	—	—
diff-in-means (LDA)	18	33	56	28	33	21	52	67	43	38	7	57	86	57	57
probe LR (balanced)	17	24	71	29	29	24	38	46	38	38	5	100	100	100	100
PCA	34	32	91	35	32	—	—	—	—	—	—	—	—	—	—
random (note-matched)	284	**79**	**93**	**70**	**65**	16	100	100	94	94	—	—	—	—	—

source	band	n	S	D	G	M
JumpReLU SAE *(pub 380)*	$[0.4, 0.5)$	**136**	$\mathbf{-8.8}$	$\mathbf{-11.0}$	$\mathbf{-11.0}$	$\mathbf{-10.3}$
JumpReLU SAE, seed 2† *(pub 380)*	$[0.4, 0.5)$	143	-11.9	-13.3	-9.8	-15.4
keyword (ceiling)	$[0.3, 0.4)$	13	-7.7	-7.7	-7.7	0.0
keyword (ceiling)	$[0.4, 0.5)$	13	0.0	$+15.4$	0.0	0.0
JumpReLU SAE *(strat 200)*	$[0.3, 0.4)$	40	-10.0	-22.5	0.0	-10.0
JumpReLU SAE *(strat 200)*	$[0.4, 0.5)$	40	-7.5	-5.0	-17.5	-15.0
JumpReLU SAE, seed 2† *(strat 200)*	$[0.3, 0.4)$	40	-20.0	-12.5	-7.5	-7.5
JumpReLU SAE, seed 2† *(strat 200)*	$[0.4, 0.5)$	40	-15.0	-12.5	-20.0	-20.0
diff-in-means (LDA)	$[0.3, 0.4)$	10	-10.0	-10.0	-10.0	-20.0
diff-in-means (LDA)	$[0.4, 0.5)$	11	0.0	$+9.1$	-9.1	0.0
probe LR (balanced)	$[0.3, 0.4)$	14	-7.1	-7.1	0.0	0.0
probe LR (balanced)	$[0.4, 0.5)$	10	0.0	0.0	0.0	0.0

well above chance under the easy slate, so it has the most to lose, and the ceiling is nearly unmoved because a curated keyword direction names its concept outright.

Forced-binary re-judging. Removing PARTIAL from the option set entirely, on all four judges, gives a binary YES rate of 32.4 / 22.4 / 33.7 / 31.1% for the JumpReLU SAE, 37.6 / 23.9 / 35.3 / 30.0% for its seed-2 replicate and 33.4 / 21.3 / 32.4 / 26.6% for the vanilla SAE against 9.3 / 7.0 / 9.3 / 10.7% for random directions (S / D / G / M), the correct ordering restored under every judge, at a 2.5–4.0× separation. Only 6.3–24.2% of previously PARTIAL verdicts survive as YES, which confirms that PARTIAL was absorbing non-answers rather than naming a real intermediate category, and locates the failure of the pooled metric in the verdict set rather than in any judge.

Operational note on reasoning-model judges. GPT-5-mini is a reasoning model. At the judging pipeline's default `max_tokens` of 256 it spends the entire budget on hidden reasoning and returns empty content, which the parser records as UNKNOWN, so a failed call is silently scored as a wrong answer and hit@1 is depressed. It must be run with minimal reasoning effort. Gemini 2.5 Flash emits no reasoning tokens and needs no override. Because both this failure mode and provider rate-limiting produce plausible-looking summaries rather than errors (a failed call is recorded as a wrong answer, not skipped), every judge run reported here was gated on a raw per-feature audit: across the 8,640 third- and fourth-judge judgments of the headline arms, 0 blank, 0 error and 1 unparseable, and across the 4,964 judgments of the forced-binary and cross-chapter arms, 0 blank, 0 error and 0 unparseable.