A Reproduction Study of Weight-Based Mechanistic Interpretability in Bilinear MLPs

Anonymous authors
Paper under double-blind review

Abstract

Mechanistic interpretability typically relies on post-hoc analysis of model activations, but bilinear MLPs offer an alternative: architectures whose weights are directly interpretable through eigendecomposition of interaction tensors.

We reproduce both main experiments from Pearce et al. (2025): their Section 4 (Vision) on MNIST/Fashion-MNIST, and their Section 5 (Language) discovering sentiment negation circuits via Sparse Autoencoder analysis. Vision results reproduce cleanly: weight decay reduces effective rank from 38.5 to 15.5 while maintaining 97–98% accuracy, and our ablation shows that weight decay—not noise augmentation—is the primary driver of low-rank structure.

In language, we confirm the AND-gate negation circuit (two semantically contrasting negation features, cosine similarity -0.16), but do *not* fully reproduce the low-rank interaction claim: the fraction of features achieving >0.75 rank-2 correlation varies from 32% (ts-medium) to 65% (fw-small); only fw-small meets this threshold.

We provide threshold sensitivity analysis (Table 3) and trace the gap to SAE training duration (correlation improves 2.6× over five checkpoints) and model compute (tokens/parameter); the fw-medium configuration required 8× rather than 4× expansion SAEs, making exact reproduction impossible—language results constitute constrained replication under publicly available artifacts. In extensions, regularized bilinear MLPs transfer structurally across digit and letter datasets: MNIST-trained models classify geometrically similar EMNIST letters (O→0, I→1, Z→2, S→5) at 87–100% accuracy. We propose Quadratic Form Similarity, which separates similar from dissimilar digit-letter pairs (QFS 0.40 vs. -0.06, $p < 10^{-4}$) where cosine similarity fails (0.358 vs. 0.339). Finally, we explore CP-decomposition as an architectural constraint, achieving 93.8% accuracy with effective rank 17.5 at ∼30× faster training, with CP factors that appear qualitatively more localized than dense eigenvectors—though interpretability gains remain preliminary.

1 Introduction

Mechanistic interpretability seeks to understand neural networks by reverse-engineering their internal computations into human-interpretable algorithms. The dominant approach trains Sparse Autoencoders (SAEs) post-hoc on model activations to discover interpretable features (Bricken et al., 2023; Cunningham et al., 2023). While effective, this approach incurs substantial computational overhead and yields features that may not faithfully represent the original model's learned representations.

Pearce et al. (2025) propose an alternative paradigm: *intrinsic* interpretability through architectural design. Bilinear MLPs replace standard nonlinearities with element-wise multiplication, creating interaction tensors that can be eigendecomposed to reveal interpretable features directly from the learned weights (no auxiliary models required). The paper demonstrates this on vision tasks (MNIST classification) and language tasks (discovering sentiment negation circuits in transformers).

Weight-based interpretability offers a path to transparent AI systems without the computational overhead of training auxiliary interpretation models. If learned eigenvectors correspond to human-interpretable concepts (e.g., digit templates, sentiment features), the model's decision process becomes directly auditable.

We reproduce both main experimental sections of Pearce et al. (2025):

- **Vision Reproduction (Section 4.1):** Fully reproduced. Weight decay reduces effective rank from 38.5 to 15.5 while maintaining 97–98% accuracy. Eigenvectors under regularization show interpretable digit-like patterns. Our ablation disentangles noise and weight decay, finding that weight decay alone achieves the lowest rank.
- **Language Reproduction (Section 4.2):** Partially reproduced. We confirm AND-gate negation circuits with semantically contrasting features (not-bad vs. not-good), but the fraction of features achieving high (>0.75) rank-2 correlation varies by model (32%–65%). Under the original >0.75 threshold, fw-small (65%) reproduces the claim while ts-medium (32%) and fw-medium (45%) do not; Table 3 provides sensitivity across thresholds. We trace this to SAE training duration: comparing five checkpoints, correlation improves 2.6× (0.15 → 0.39) with longer training.
- **Extension 1 (Section 5):** We demonstrate that regularized bilinear MLPs learn structural features that transfer across MNIST, EMNIST, and USPS. We propose Quadratic Form Similarity to quantify structural alignment where cosine similarity fails.
- **Extension 2 (Section 6):** We explore CP-decomposition as an architectural constraint for intrinsic low-rank structure, achieving 93.8% accuracy with qualitatively more localized CP factors at effective rank 17.5, while gaining significant efficiency (∼30× faster training).

2 Scope of Reproducibility

We reproduce the main experimental sections of Pearce et al. (2025): Pearce et al. Section 4 (Vision) demonstrating interpretable eigendecomposition on MNIST and Fashion-MNIST, and Pearce et al. Section 5 (Language) discovering negation circuits in bilinear transformers using Sparse Autoencoder (SAE) analysis.

2.1 Claims to Verify

Section 4: Vision (Image Classification). The paper makes two main families of claims:

- **Low-rank, interpretable structure:** Regularization induces low-rank interaction tensors whose top eigenvectors form digit-like templates rather than noise, with consistent spectra across seeds.
- **Robustness and controllability:** Noise and weight decay play complementary roles, enabling adversarial generation and truncation to a small number of eigenvectors without large accuracy loss.

Section 5: Language (Negation Circuits). For language, the paper claims that:

- **SAE-based circuits:** SAEs reveal interpretable features implementing negation-like AND-gate circuits with opposing eigenvalues for "not + positive" vs. "not + negative" features.
- **Low-rank interactions across models:** A large fraction of SAE features admit high rank-2 correlation and sparse interaction matrices that generalize across model sizes.

2.2 Success Criteria

Vision: Effective rank ratio (regularized/unregularized) < 0.5; top eigenvectors visually resemble digit templates; test accuracy within 2% of paper (∼95–98%); results consistent across 5 seeds.

Language: AND-gate circuits show opposing eigenvalue signs; a substantial fraction of features show >0.75 rank-2 correlation. We report results under the original >0.75 threshold and provide sensitivity across >0.40–>0.75 (Table 3); interaction matrices show interpretable block structure.

2.3 Results Summary

Table 1 summarizes our reproduction outcomes: all six vision claims are fully reproduced, while language claims are partially reproduced—AND-gate circuits are confirmed, but under the original >0.75 threshold, low-rank correlation is reproduced only for fw-small (65%), not ts-medium (32%) or fw-medium (45%).

Table 1: Summary of reproduction results. ✓= reproduced, × = not reproduced, ∼ = partially reproduced.

Claim	Description	Status
	Vision (Section 4)	
V1	Low-rank emergence	✓
V2	Interpretable eigenvectors	✓
V3	Overfitting without reg.	✓
V4	Regularization roles	✓
V5	Cross-seed consistency	✓
V6	Adversarial generation	✓
	Language (Section 5)	
L1	SAE feature discovery	✓
L2	Negation/AND-gate circuits	✓
L3	Low-rank interactions	∼
L4	Sparse interactions	✓
L5	Cross-model generalization	∼

3 Methodology

3.1 Bilinear Layer Architecture

The bilinear layer replaces standard nonlinearities with element-wise multiplication:

$$\mathbf{y} = (\mathbf{W}_l \mathbf{x}) \odot (\mathbf{W}_r \mathbf{x}) \tag{1}$$

where $\mathbf{W}_l, \mathbf{W}_r \in \mathbb{R}^{d_{\text{hidden}} \times d_{\text{in}}}$ are learnable weight matrices and $\odot$ denotes the Hadamard product. Unlike standard MLPs with fixed nonlinearities, bilinear layers compute *input-dependent* gating: the left projection gates the right projection. This creates a quadratic dependence on the input expressible as a third-order interaction tensor:

$$y_c = \sum_{i,j} B_{cij} x_i x_j \tag{2}$$

where $B_{cij} = \sum_h W_{\text{head},ch} (W_l)_{hi} (W_r)_{hj}$ and $W_{\text{head}} \in \mathbb{R}^{d_{\text{out}} \times d_{\text{hidden}}}$ is the output projection.

3.2 Eigendecomposition for Interpretability

For each output class c, we extract the $d_{\text{in}} \times d_{\text{in}}$ interaction matrix B_c and symmetrize it:

$$B_c^{\text{sym}} = \frac{1}{2}(B_c + B_c^\top) \tag{3}$$

Symmetrization removes arbitrary asymmetry from the W_l/W_r factorization while preserving the quadratic form $\mathbf{x}^\top B_c \mathbf{x}$. The symmetric matrix admits eigendecomposition:

$$B_c^{\text{sym}} = \sum_{r=1}^{d_{\text{in}}} \lambda_r^{(c)} \mathbf{v}_r^{(c)} (\mathbf{v}_r^{(c)})^\top \tag{4}$$

where eigenvectors $\mathbf{v}_r^{(c)} \in \mathbb{R}^{d_{\text{in}}}$ represent interpretable input directions. If the eigenspectrum is *low-rank* (few dominant eigenvalues), the model's decision depends on only a few input directions that can be directly inspected.

We quantify spectral concentration using effective rank (Roy & Vetterli, 2007):

$$\text{EffRank}(\boldsymbol{\lambda}) = \left(\frac{\|\boldsymbol{\lambda}\|_1}{\|\boldsymbol{\lambda}\|_2}\right)^2 = \frac{\left(\sum_i |\lambda_i|\right)^2}{\sum_i \lambda_i^2} \tag{5}$$

where $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_d)$ is the vector of eigenvalues from the decomposition. This scale-invariant measure ranges from 1 (single dominant eigenvalue) to d (uniform spectrum).

3.3 Language Model Analysis with SAEs

For language experiments, we combine bilinear decomposition with Sparse Autoencoders (SAEs) (Bricken et al., 2023). SAEs decompose activations into sparse features via $\mathbf{z} = \text{TopK}(\mathbf{W}_{\text{enc}}\mathbf{h})$, where expansion factors of 4–8× disentangle superimposed concepts into monosemantic directions.

By placing SAEs at MLP input and output, we analyze how sparse input features interact to produce sparse outputs. The bilinear structure implies $z_f^{\text{out}} \approx (\mathbf{z}^{\text{in}})^\top Q_f \, \mathbf{z}^{\text{in}}$, where Q_f is the interaction matrix for output feature f. Eigendecomposing Q_f reveals dominant input directions-if rank-2 captures most variance, the feature depends on just two input directions, enabling circuit discovery. We use pretrained SAEs with TopK sparsity ($k = 30$); the paper claims "most features" achieve >0.75 rank-2 correlation. Because "most" is not formally specified, we report results under multiple thresholds (Table 3).

3.4 Datasets

Vision Datasets. We use MNIST (LeCun et al., 1998) (60K train / 10K test, 28×28 grayscale handwritten digits) and Fashion-MNIST (Xiao et al., 2017) (60K train / 10K test, 28×28 grayscale clothing images). Images are normalized to $[0, 1]$ and flattened to 784-dimensional vectors. For cross-dataset experiments (Section 5), we additionally use EMNIST-Digits/Letters (Cohen et al., 2017) and USPS (Hull, 1994) (upscaled from 16×16 to 28×28).

Language Datasets. The pretrained bilinear transformers use two datasets: **TinyStories** (Eldan & Li, 2023) (2.1M synthetic children's stories, ∼470M tokens) for `ts-medium`, and **FineWeb-EDU** (Penedo et al., 2024) (educational web text subset of FineWeb, 1.3T tokens) for `fw-small` and `fw-medium`. For correlation analysis (Figure 5), all three models use ∼1.57M tokens (1,572,864) from their respective validation sets.

3.5 Experimental Setup

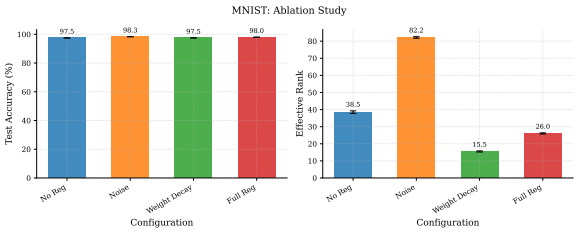
For vision experiments, we train bilinear MLPs on MNIST and Fashion-MNIST under four regularization configurations (Table 2), each with 5 random seeds (42–46). Architecture: $d_{\text{hidden}} = 256$, 100 epochs, batch size 2048, AdamW optimizer with lr 10^{-3} and cosine annealing. For language experiments, we analyze pretrained bilinear transformers (Table 4, Appendix C). Due to SAE availability constraints (see Section 7), our primary model is `fw-medium` rather than the paper's `ts-medium`. We implemented a memory-efficient streaming algorithm for eigendecomposition to handle large interaction tensors.

Table 2: Regularization configurations (from paper Appendix G.1).

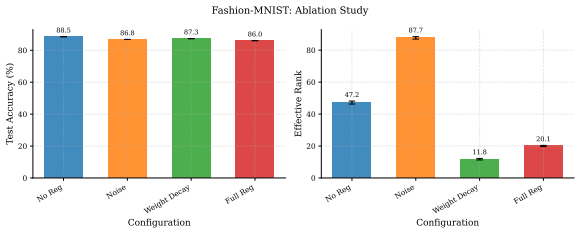
Configuration	Noise Std	Weight Decay
`none`	0.0	0.0
`noise`	0.5	0.0
`wd`	0.0	1.0
`full`	0.5	1.0

4 Reproduction Results

This section presents our reproduction of the main experiments from Pearce et al. (2025): Pearce et al. Section 4 (Vision) on MNIST/Fashion-MNIST and Pearce et al. Section 5 (Language) on bilinear transformers.

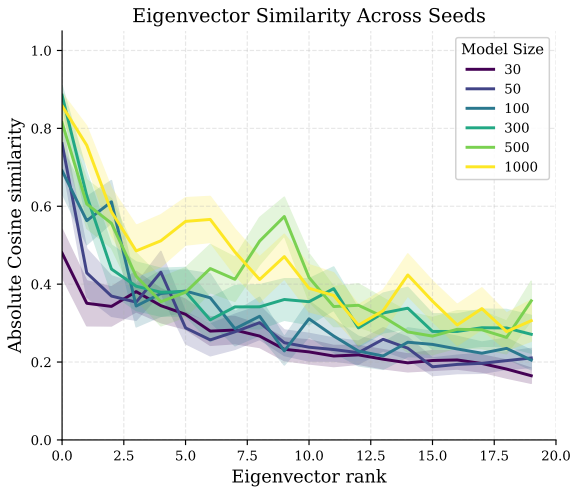


(a) MNIST: Accuracy vs. effective rank

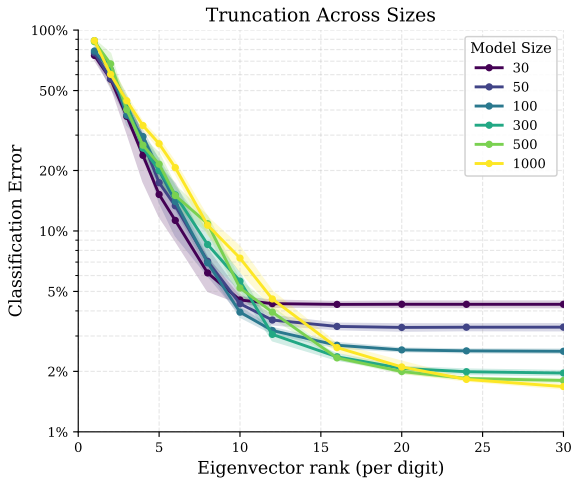


(b) Fashion-MNIST: Accuracy vs. effective rank

Figure 1: Ablation study showing accuracy-interpretability trade-off across regularization configurations. Weight decay achieves the lowest effective rank on both datasets while maintaining competitive accuracy. The effective rank ratio (regularized/unregularized) meets the paper's < 0.5 criterion.



(a) Eigenvector similarity across seeds



(b) Truncation error by model size

Figure 2: Paper Figure 5 reproduction. **(a)** Top eigenvectors are highly consistent across random seeds, especially for larger models, with cosine similarity decreasing smoothly as eigenvector rank increases. **(b)** Truncation error decreases with model size; keeping ∼20 eigenvectors per class is sufficient for near-full accuracy.

4.1 Vision Results (Paper Section 4)

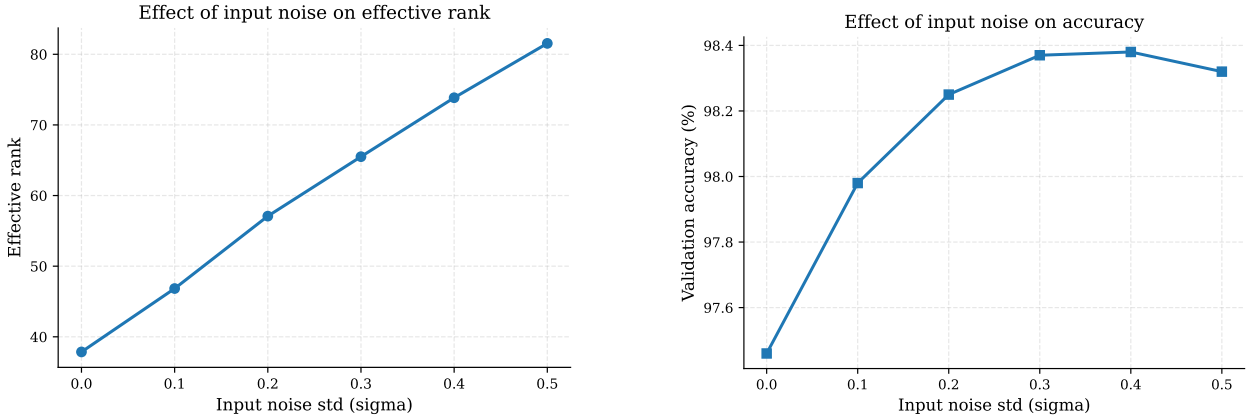
We trained bilinear MLPs on MNIST and Fashion-MNIST under four regularization configurations across 5 random seeds each.

4.1.1 Main Result: Regularization Induces Low-Rank Structure (Claim V1)

The paper's central claim is that regularization induces low-rank structure in the bilinear interaction tensor. Figure 1 confirms this through our ablation experiments. Weight decay is the primary driver of low-rank structure, achieving the lowest effective rank (15.5 vs. 26.0 for full regularization), while noise augmentation increases effective rank but improves accuracy. The effective rank ratio ($15.5/38.5 \approx 0.40$) meets the paper's $<$ 0.5 criterion. Eigenvalue distributions reveal sharp spectral decay with regularization (Figure 6, Appendix B), concentrating variance in 10–20 eigenvectors. Top eigenvectors are interpretable digit-like patterns under regularization, while unregularized models show diffuse, noise-like patterns (Figures 13 and 14, Appendix B).

4.1.2 Model Size and Truncation (Claim V5)

We confirm the paper's claim that bilinear MLPs maintain near-full accuracy with low-rank truncation: truncating to around 20 eigenvectors per class is sufficient even as model size increases (Figure 2). Top eigenvectors are highly consistent across random seeds, especially for larger models.



(a) Effective rank increases with noise std

(b) Accuracy improves then saturates

Figure 3: Noise augmentation effect on effective rank and accuracy (reproducing paper Figure 4). Higher noise spreads variance across more eigenvectors while improving generalization.

4.1.3 Additional Vision Results

We reproduce the original paper's Figures 4, 6, and 7; full visualizations are in Appendix B. Increasing noise improves eigenvector interpretability while trading off with effective rank (Figure 3). On binary similarity classification, the top eigenvector closely resembles the target digit (Figure 8, Appendix B), and eigenvector-derived masks cause significantly larger accuracy drops than random masks (Figure 10, Appendix B), confirming V6.

4.2 Language Results (Paper Section 5)

We reproduce negation circuit discovery using pretrained bilinear transformers and SAEs from HuggingFace.[1] Our protocol: pass tokenized batches through each model, record `mlp_in`/`mlp_out` activations, encode with output SAE, and compare sparse features to low-rank predictions from interaction eigenpairs. We analyze all *active features* ($\geq$1 active token) using Pearson correlation; features with fewer than 5 active tokens are excluded to ensure statistically meaningful correlation estimates. **SAE availability gap:** `ts-medium-scope` lacks `mlp-in` SAEs for layer 4, so Figure 8's ts-tiny setup cannot be reproduced. We use `fw-medium` (layer 7, expansion 8) which has both SAE types.

4.2.1 Negation Circuit Discovery (Paper Figure 8)

The paper claims bilinear layers enable circuit discovery via eigendecomposition. For output feature f, the interaction matrix Q_f captures how input feature pairs contribute to f's activation: $z_f^{\text{out}} \approx (\mathbf{z}^{\text{in}})^\top Q_f \mathbf{z}^{\text{in}}$. Eigendecomposing Q_f reveals dominant input directions. Figure 4 shows this for negation features in fw-medium: **(A) Interaction submatrix**: Q_f restricted to top-15 input features, grouped by cluster. Red blocks indicate positive interactions (features that co-activate f), blue indicates negative. **(B) Eigenvector projections**: Input features projected onto top eigenvectors. Neg./Pos. sentiment features cluster separately; arrows show "good"/"bad" unembed directions and "not" input direction. The black dot marks where the output feature lies at the intersection of negation and sentiment. This reveals the AND-gate circuit: *negation* $\otimes$ *sentiment* $\rightarrow$ *negated-sentiment*.

Feature selection and interpretation: Following the tutorial methodology, we select output features by ranking all 8,192 features by their top eigenvalue magnitude-a heuristic for identifying features with strong interaction structure. Features 3834 and 751 emerge as the top two, with weakly negative cosine similarity (-0.16) between their SAE decoder directions—what the tutorial calls a "somewhat linear subspace." Semantic interpretation comes from inspecting top-activating tokens: feature 3834 fires on negated negative words ("no losses", "not lost"), while feature 751 fires on negated positive words ("not harmless", "doesn't go

[1] SAE repositories: `tdooms/fw-medium-scope`, `tdooms/ts-medium-scope`, `tdooms/fw-small-scope`.

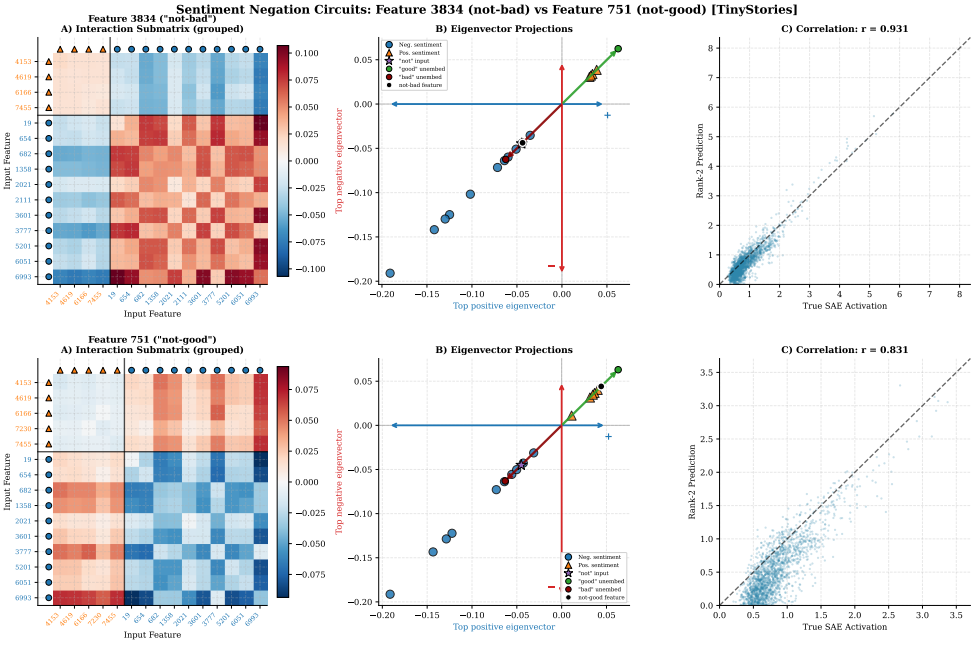


Figure 4: Negation circuit discovery (fw-medium, layer 7, TinyStories evaluation). Top: feature 3834 ("not-bad", $r = 0.93$); bottom: feature 751 ("not-good", $r = 0.83$). (A) Interaction submatrix Q_f shows block structure: positive/negative sentiment features interact oppositely. (B) Eigenvector projections reveal clustering: sentiment features separate along the "good"↔"bad" axis.

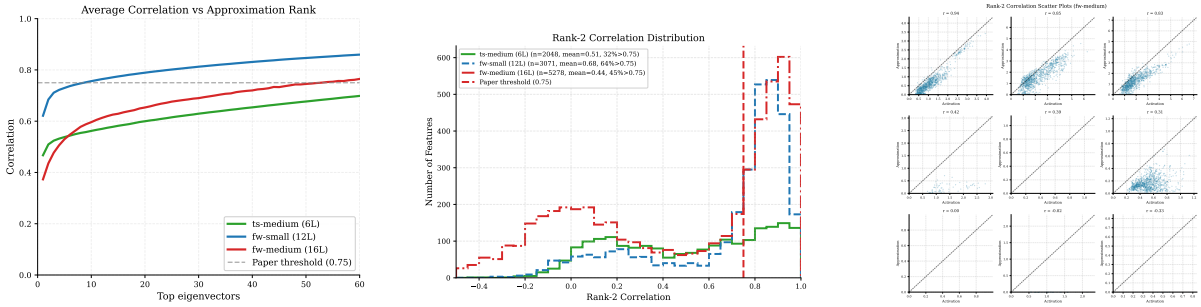
away"). These are semantically contrasting negation features (not-bad vs. not-good), though not geometrically opposing (-0.16 is closer to orthogonal than anti-parallel). As additional validation, we test the circuit on FineWeb-16k (the model's training distribution): correlations remain strong (0.91/0.82 vs. 0.93/0.83 on TinyStories), demonstrating robustness to distribution shift (Appendix D).

4.2.2 Low-Rank Correlation Analysis (Paper Figure 9)

The paper claims most features achieve >0.75 rank-2 correlation; because "most" is not formally specified, we report results across thresholds from >0.40 to >0.75 (Table 3). For this analysis we use a different layer per model than in the circuit case study above: fw-medium layer 10 rather than layer 7, because layer 10 lies at 2/3 network depth (where the paper's Figure 9 configurations are specified) while layer 7 was chosen for circuit discovery based on SAE availability at that depth. For each token, we record MLP input activations and encode them with the output SAE (k=30 TopK sparsity, matching the paper) to obtain sparse feature activations. For each feature f with at least 5 active tokens (n denotes the number of such features), we eigendecompose its interaction matrix Q_f and compute Pearson correlation between true and low-rank predicted activations across those active tokens. All models processed ∼1.57M tokens (1,572,864): ts-medium used TinyStories validation (Eldan & Li, 2023), fw-small and fw-medium used FineWeb-Edu (Penedo et al., 2024).

4.2.3 Rank-2 Correlation: Distribution and Sensitivity

The rank-2 correlation distributions (Figure 5b) tell a richer story than any single threshold. Means range from 0.435 (fw-medium) to 0.685 (fw-small), with medians from 0.550 (ts-medium) to 0.814 (fw-small). Notably, fw-medium's median (0.649) sits above ts-medium's (0.550) despite a lower mean (0.435 vs. 0.509) and lower >0.75 fraction (44.5% vs. 32.3%), reflecting a distribution concentrated around 0.5–0.7 but with fewer high-correlation features. Table 3 reports the fraction of features above several thresholds. The > 0.75 threshold is one point on this distribution: fw-small reproduces the "most features" claim robustly across all thresholds, while ts-medium and fw-medium fall below it and remain marginal even under weaker criteria.



(a) Correlation vs. rank (b) Rank-2 distribution (c) True vs. predicted (fw-medium)

Figure 5: Low-rank correlation (Paper Figure 9). ts-medium (L4, exp=4, n=2048), fw-small (L8, exp=4, n=3071), fw-medium (L10, exp=8, n=5278). n = features with $\geq$5 token activations included in Pearson r analysis.

Table 3: Rank-2 correlation sensitivity across thresholds (features with valid Pearson r).

Model	n	Mean	Median	>0.40	>0.50	>0.60	>0.75
ts-medium	2048	0.509	0.550	59.2%	53.3%	46.2%	32.3%
fw-small	3071	0.685	0.814	80.3%	78.0%	75.6%	64.5%
fw-medium	5278	0.435	0.649	57.0%	54.4%	51.7%	44.5%

4.2.4 Root Cause Analysis

The paper notes correlation improves with SAE training time (Appendix H). We confirm this using fw-medium layer 12 checkpoints v0–v4 (Figure 15, Appendix C): rank-2 correlation improves from 0.15 to 0.39 with longer training, consistent with the paper's acknowledgment that "correlation drastically improves with longer SAE training."

Model training duration hypothesis: We observe a correlation between training compute (tokens/parameter) and low-rank behavior: ts-medium (83 tokens/param, n=2048, 32%), fw-medium (95 tokens/param, n=5278, 45%), and fw-small (198 tokens/param, n=3071, 65%). Under the original >0.75 threshold, only fw-small reproduces the "most features" claim, while ts-medium and fw-medium fall short. Since more training tokens corresponds to longer training time, this parallels Figure 15: just as undertrained SAEs fail to capture low-rank structure, undertrained models (low tokens/param) may not develop the structured interactions that exhibit low-rank approximability. Figure 5(c) visually confirms this: only ~3 of 9 randomly sampled features show strong true-vs-predicted correlation.

SAE artifact limitation: The paper uses 4x expansion SAEs for all models. However, for fw-medium at 2/3 depth (layer 10–11), only 8× expansion SAEs are publicly available; 4× is only available at layers 6 and 12. With 8x expansion, 8192 total features are available vs. 4096 for 4x, yielding more sparse/specialized features that potentially lower average correlation. Of these, 5,688 have $\geq$1 active token; 5,278 fire on at least 5 tokens and are included in Pearson r analysis (the remaining 410 fire on $\leq$4 tokens). This artifact gap prevents exact reproduction of the paper's fw-medium configuration.

5 Cross-Dataset Structural Robustness

A core promise of weight-based interpretability is that models learn meaningful geometric features rather than memorizing dataset-specific pixel correlations. We test whether bilinear MLPs learn reusable geometric primitives (e.g., "0-ness" as circularity) rather than idiosyncratic MNIST stroke patterns.

Throughout this section, **baseline** refers to models without regularization, while **regularized** refers to full regularization (noise std = 0.5, weight decay = 1.0) as in Table 2. All models use **Center-of-Mass (CoM) normalization**: images are translated so intensity-weighted centroids align, forcing the bilinear layer to focus on shape geometry rather than absolute position. We probe robustness under three distributional shifts: (1) **Writer independence**: MNIST $\leftrightarrow$ EMNIST-Digits (same labels, different writers); (2) **Domain transfer**:

MNIST → USPS (different resolution/collection); (3) **Geometric semantics**: MNIST → EMNIST-Letters, where letters are semantically different but geometrically similar (O→0, I→1, Z→2, S→5).

5.1 Functional Analysis: Regularization Enables Transfer

With the regularized CoM model, cross-dataset accuracy is near-perfect and symmetric for MNIST ↔ EMNIST-Digits (97.5% and 97.9%, Table 7, Appendix E), confirming that regularized bilinear features do not overfit to MNIST writers. For domain transfer to USPS (upscaled from 16×16 to 28×28), regularization yields +19 pp improvement (72.3% vs. 53.4%, Table 8, Appendix E). On geometrically similar letters (Table 9, Appendix E), the model's most confident prediction is exactly the corresponding digit: O→0 (99.6%), I→1 (87.1%), Z→2 (88.8%), S→5 (87.4%).

5.2 Structural Analysis: Quadratic Form Similarity

5.2.1 The Failure of Cosine Similarity

Direct eigenvector comparison via cosine similarity fails to distinguish geometrically similar pairs: MNIST-0 vs EMNIST-O (similar) yields 0.358, while MNIST-0 vs EMNIST-X (dissimilar) yields 0.339-virtually identical. Figure 20 (Appendix F) shows this failure across all digit-letter pairs, while Table 11 provides a detailed per-eigenvector breakdown confirming that mean similarities are nearly identical (0.358 vs. 0.339). The root cause: bilinear eigensystems share eigenvector directions while class information resides in eigenvalue weights (Figure 17).

5.2.2 Quadratic Form Similarity

We propose comparing full decision surfaces via Quadratic Form Similarity:

$$\text{sim}(A, B) = \frac{\text{tr}(A \cdot B)}{\|A\|_F \|B\|_F} \in [-1, 1] \tag{6}$$

This incorporates both eigenvector alignment and eigenvalue magnitudes, enabling discrimination between similar (0.400) and dissimilar (-0.060) pairs ($p_{value} < 10^{-4}$). Figure 19 (Appendix E) demonstrates this: similar pairs converge to high similarity around $k = 20$ eigenvectors, while dissimilar pairs remain near zero with non-overlapping 90% confidence intervals. Figure 18 (Appendix E) shows clear discrimination: strong diagonal for MNIST vs. EMNIST-Digits and bright entries at shape-matched pairs (0–O, 1–I, 2–Z, 5–S). Table 10 (Appendix E) confirms specificity: 3/4 pairs rank #1, with mean rank 1.2 vs. random baseline of 13.

5.3 Conclusion

Regularized bilinear MLPs learn **structural features that transfer across datasets**: the "0" circuit is highly similar to the "O" circuit across independently trained models (QFS ≈ 0.40 vs. -0.06 for dissimilar pairs, $p < 10^{-4}$). This strengthens the transparency claim-eigenvector features represent genuinely reusable shapes, not dataset-specific artifacts.

6 CP-Decomposition Analysis

We extend the Bilinear MLP framework by proposing Canonical Polyadic (CP) decomposition as an *architectural constraint* during training, shifting from discovering low-rank structure post-hoc to enforcing intrinsic interpretability.

6.1 Motivation and Methodology

Standard eigendecomposition enforces orthogonality, creating entangled features where mixed concepts are artificially separated. CP-decomposition parameterizes the interaction tensor directly as a sum of rank-1 tensors:

$$W_{cij} = \sum_{r=1}^{R} \lambda_r A_{cr} B_{ir} C_{jr} \tag{7}$$

where R is the maximum rank and λ_r are learnable scaling factors (Veeramacheneni et al., 2022). This does not enforce orthogonality, allowing factors to naturally overlap rather than being forced apart. We investigate three variants: **Fixed CP:** Hard rank constraint with no additional regularization-strict complexity bound. **Lambda CP:** L_1 regularization on learnable scales λ_r for soft sparsity (Veeramacheneni et al., 2022). **Gated CP:** Soft gates with L0 proxy for explicit rank pruning (Cao et al., 2024). All variants trained on MNIST with weight decay 0.1 across rank sweep $R \in \{32, 64, 128, 256\}$.

6.2 Quantitative Analysis: Accuracy-Interpretability Trade-off

Figure 21 and Table 12 (Appendix H) present the Pareto frontier; Table 13 (Appendix I) provides the complete rank sweep. Fixed CP operates as a strict bottleneck with eff. rank ∼3.5 regardless of allocated capacity, at ∼6% accuracy cost (91.9% vs. 97.5%). Lambda/Gated CP at $R = 256$ achieve 93.8% accuracy with eff. rank 17.5—comparable to dense models with weight decay (eff. rank 15.5)—while yielding qualitatively more localized factors in the visual analysis below.

6.3 Qualitative Analysis: Disentangled Factors

Figure 22 (Appendix I) compares eigenvectors from Dense vs. CP models. Dense eigenvectors resemble superimposed digit templates-entangled representations combining strokes for orthogonality. CP factors appear qualitatively more **localized** and parts-like: top loops, vertical strokes, bottom curves that can be linearly combined. This is suggestive of the "parts-based" representations sought in interpretability, though quantitative disentanglement metrics remain future work.

6.4 Conclusion

CP-decomposition serves dual purposes: (1) **Qualitatively more localized factors**: CP representations appear more parts-like than dense eigenvectors, though quantitative disentanglement validation is left for future work. (2) **Scalable efficiency** by reducing parameters from $\mathcal{O}(d_{\text{hidden}} \times d_{\text{in}}^2)$ to $\mathcal{O}(R \times d_{\text{in}})$. Training efficiency is dramatically improved: CP models require 0.00046 GPU-h per run compared to 0.014 GPU-h for dense vision models (based on 81 GPU-tracked vision experiments, excluding challenge task; Table 14), achieving approximately 30× faster training. This efficiency gain suggests CP-decomposition as a direction worth exploring for interpretable, efficient neural architectures; these results are preliminary and limited to MNIST-scale experiments. Lambda and Gated variants provide a middle ground between strict rank constraints and dense regularization.

7 Discussion

7.1 Practical Challenges

Vision experiments were straightforward (∼3 min/run, std <1% across 40 runs). Language challenges included SAE availability gaps (requiring `fw-medium` with 8x expansion instead of paper's 4x), ambiguous definitions (correlation measure, "active" threshold), and memory constraints. Under the original >0.75 threshold, fw-small (65%) reproduces the "most features" claim, while ts-medium (32%) and fw-medium (45%) do not; Table 3 provides full sensitivity analysis.

7.2 Environmental Impact

Total footprint: 0.169 kg CO2 across 224 runs (Table 14, Appendix J).

7.3 Implications for Interpretability

Bilinear MLPs offer intrinsic transparency: eigenvectors are the learned features without auxiliary models. Low-rank approximation quality depends on both SAE and model training compute (tokens/param), emphasizing the importance of documenting training budgets in interpretability research. Weight decay reduces effective rank from 38.5 to 15.5 (2.5× interpretability improvement) at <1% accuracy cost—a favorable trade-off for transparent AI.

7.4 Communication with Original Authors

We did not communicate with the original authors; all experiments used publicly available code and pretrained artifacts.

7.5 Limitations

- **SAE artifact mismatch.** The 4x expansion SAE required by the paper is unavailable at the relevant layer for `fw-medium`; we use 8x instead. This prevents exact reproduction of that model's configuration and likely affects correlation results.
- **SAE training duration.** Correlation quality depends on how long the SAE was trained. The publicly available checkpoints may not match the training duration used in the original experiments, and the paper does not specify it.
- **Qualitative interpretability assessment.** Eigenvector quality ("digit-like patterns") is evaluated visually. We provide no quantitative measure of visual interpretability.
- **CP-decomposition scope.** CP results are limited to MNIST with a single bilinear hidden layer. Generalization to other datasets, architectures, or scales is not tested.

7.6 Reproducibility Statement

Code, configuration files, and result artifacts are available at `https://anonymous.4open.science/r/bilinear_mlp_reproduction-CE76`. Dataset and HuggingFace artifact details are in Appendix A; hardware and runtime are in Appendix J.

8 Conclusion

We fully reproduce vision results: weight decay reduces effective rank from 38.5 to 15.5 while maintaining 97–98% accuracy with interpretable digit patterns. Language results are partially reproduced: AND-gate circuits confirmed with semantically contrasting negation features (cosine similarity -0.16), but rank-2 correlation varies by model (32%–65%); only fw-small (65%) exceeds the original paper's >0.75 threshold. Low-rank approximation quality depends on both SAE convergence and model training compute; language results constitute constrained replication rather than exact reproduction. Extensions show structural transfer across digit and letter datasets, and CP-decomposition achieves 93.8% accuracy with qualitatively more localized factors at $\sim 30\times$ faster training. Future work should develop quantitative interpretability metrics and investigate CP-decomposition for cross-domain robustness.

Broader Impact Statement

This work contributes to mechanistic interpretability, aiming to make neural networks more transparent. Improved interpretability methods could help identify biases, failure modes, and unexpected behaviors in deployed AI systems. Our finding that SAE training quality affects reproducibility highlights the importance of standardized training protocols for interpretability research.

References

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023.

Tianxiao Cao, Lu Sun, Canh Hao Nguyen, and Hiroshi Mamitsuka. Learning low-rank tensor cores with probabilistic l_0-regularized rank selection for model compression. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI-24)*, 2024.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André Van Schaik. EMNIST: Extending MNIST to handwritten letters. *arXiv preprint arXiv:1702.05373*, 2017.

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

Ronen Eldan and Yuanzhi Li. Tinystories: How small can language models be and still speak coherent english? *arXiv preprint arXiv:2305.07759*, 2023.

Jonathan J Hull. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):550–554, 1994.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Michael Pearce, Thomas Dooms, Alice Rigg, Jose Oramas, and Lee Sharkey. Bilinear MLPs enable weight-based mechanistic interpretability. In *The Thirteenth International Conference on Learning Representations*, 2025.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben Allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. FineWeb: Decanting the web for the finest text data at scale. *arXiv preprint arXiv:2406.17557*, 2024.

Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *European Signal Processing Conference (EUSIPCO)*, pp. 606–610, 2007.

Lokesh Veeramacheneni, Moritz Wolter, Reinhard Klein, and Jochen Garcke. Canonical convolutional neural networks. *arXiv preprint arXiv:2206.01509*, 2022.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

A Hyperparameter Details

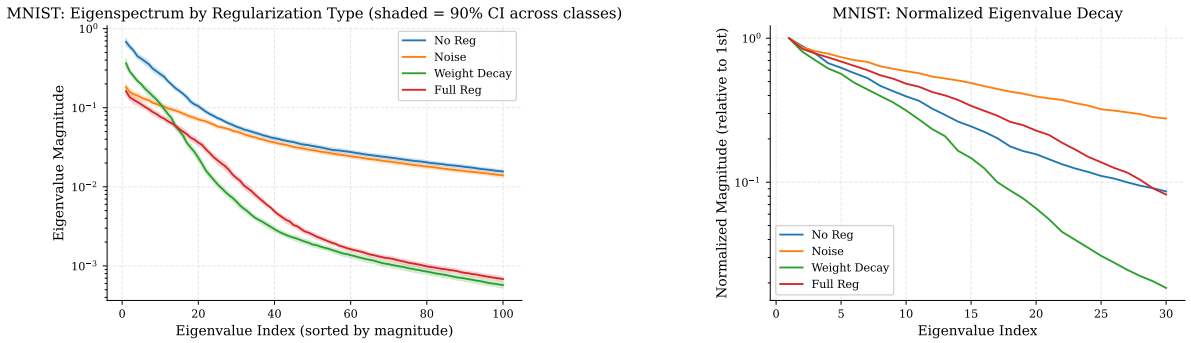
Vision Experiments. Architecture: 1-hidden-layer bilinear MLP, $d_{\text{hidden}} = 256$. Training: 100 epochs, batch size 2048, AdamW optimizer, lr 10^{-3} with cosine annealing. Seeds: 42–46 (5 seeds per configuration).

Language Experiments. Models: `ts-medium` (6L, 512d, 29M params), `fw-small` (12L, 768d, 162M params), `fw-medium` (16L, 1024d, 335M params). SAEs: TopK sparsity ($k = 30$), expansion ratios 4–8×. Correlation analysis: features with $\geq$5 token activations per model (n=2,048 / 3,071 / 5,278); fw-medium has 5,688 total active features, of which 410 fire on $\leq$4 tokens and are excluded.

B Vision Results

This appendix provides visualizations from our reproduction of the paper's vision experiments (Section 4).

B.1 Eigenspectrum Analysis



(a) Eigenspectrum by regularization

(b) Eigenvalue decay rate

Figure 6: Eigenspectrum analysis on MNIST. Weight decay produces the steepest spectral decay, concentrating variance in 10–20 eigenvectors.

B.2 Noise Effect (Paper Figure 4)

Figure 7 shows how the top eigenvector for digit 5 changes across noise augmentation levels. Higher noise produces more diffuse, averaged eigenvectors that better capture the canonical digit shape.

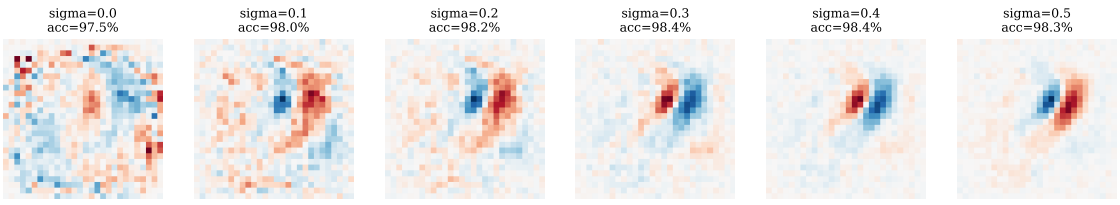


Figure 7: Top eigenvector for digit 5 across noise levels ($\sigma = 0.0$ to 0.5). Accuracy improves with noise despite spreading variance across more eigenvectors.

B.3 Challenge Task (Paper Figure 6)

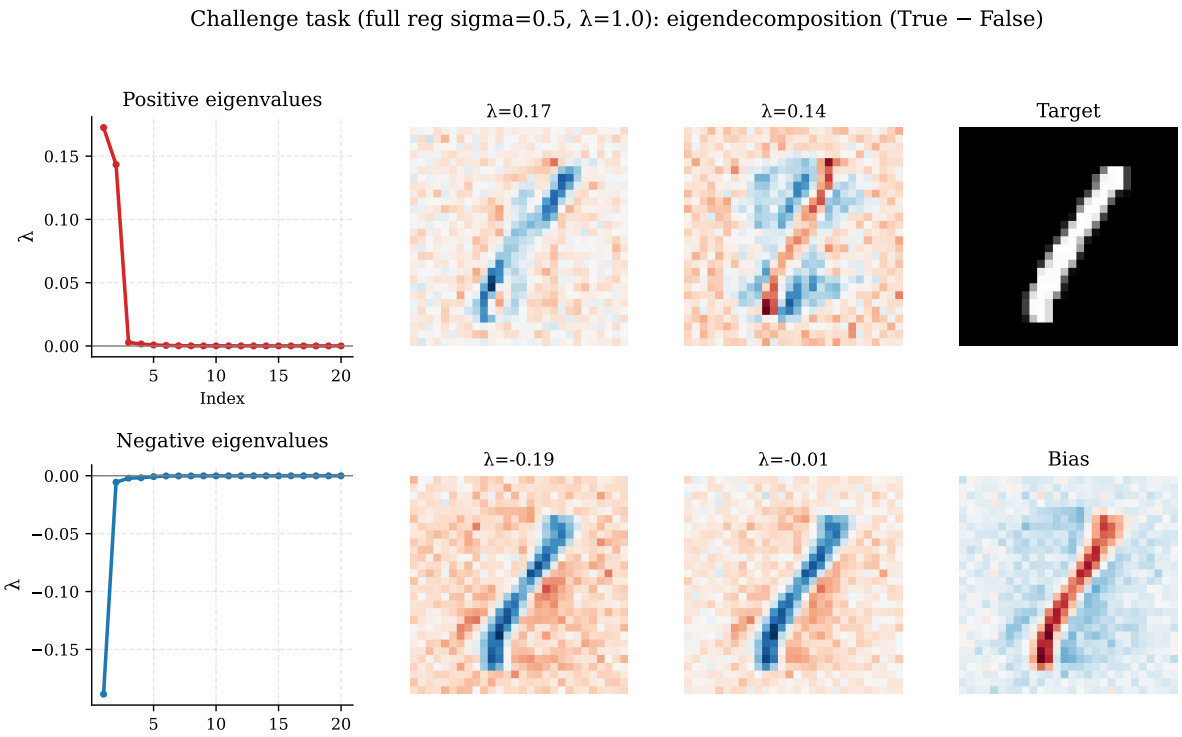


Figure 8: Challenge task eigendecomposition with full regularization (noise $\sigma = 0.5$, weight decay $= 1.0$), reproducing paper Figure 6. **Left:** Eigenvalue decay shows dominant positive/negative modes. **Middle:** Top eigenvectors resemble the target digit pattern. **Right:** Target digit and bias visualization.

We trained challenge task models under four regularization configurations to analyze how regularization affects the eigendecomposition structure. Figure 9 compares eigenvalue decay across these variants.

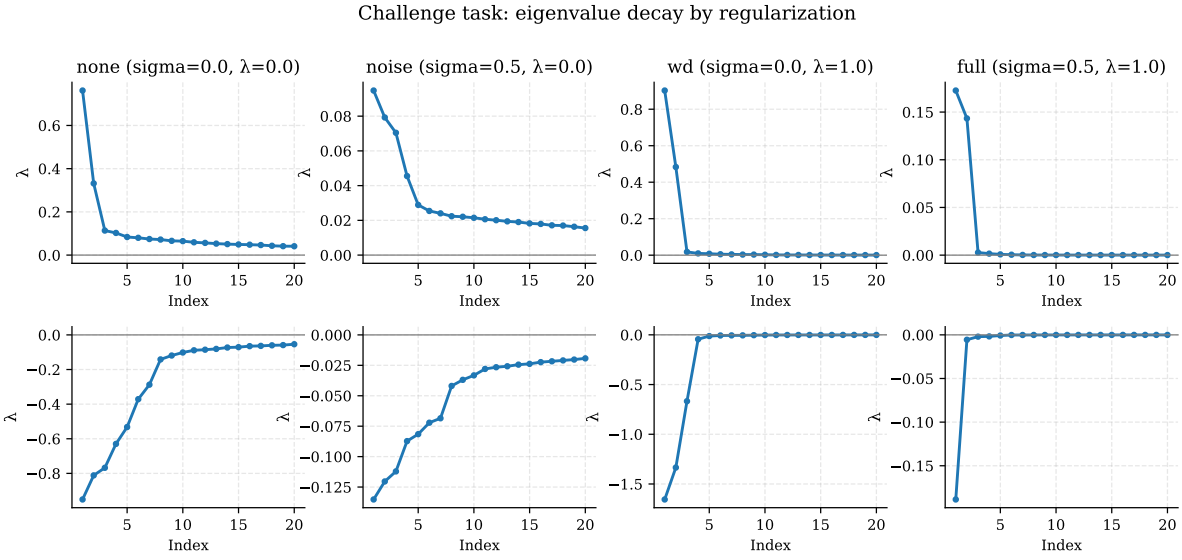


Figure 9: Challenge task eigenvalue decay by regularization configuration. Weight decay produces the sharpest spectral decay, concentrating information in fewer eigenvectors.

B.4 Adversarial Perturbations (Paper Figure 7)

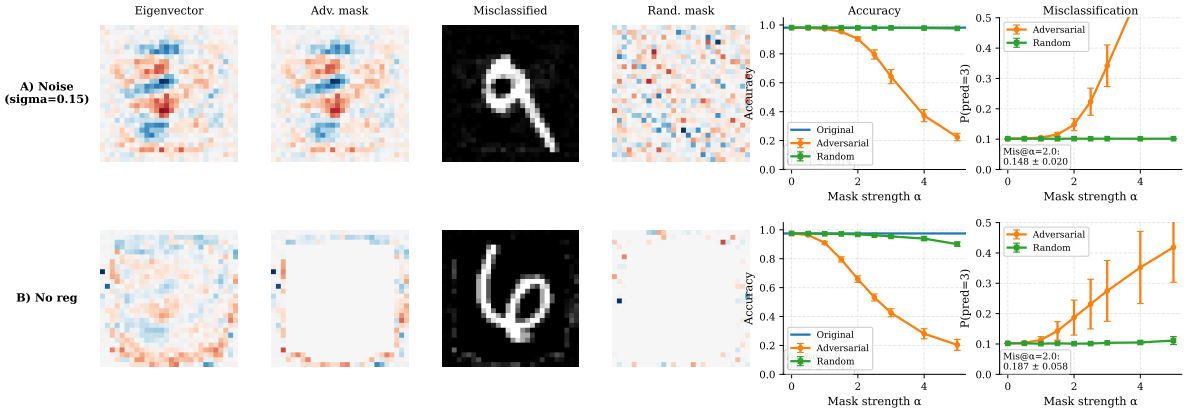


Figure 10: Adversarial mask analysis (reproducing paper Figure 7). **Row A:** Noise-regularized model ($\sigma = 0.15$). **Row B:** No regularization (with rare-edge constraint). Eigenvector-derived masks cause significantly larger accuracy drops than random masks of equal magnitude.

B.5 Eigenvector Quality Comparison

Figure 11 shows the effect of noise-only regularization on eigenvector interpretability.

Figure 11: Top eigenvectors with noise augmentation only ($\sigma = 0.5$, $\lambda = 0.0$). Noise alone produces recognizable digit patterns, though less sharp than with weight decay.

B.6 Cross-Dataset Comparison: MNIST vs Fashion-MNIST

Figure 12 shows eigenvector quality on Fashion-MNIST under noise-only regularization.

Figure 12: Fashion-MNIST top eigenvectors with noise augmentation ($\sigma = 0.5$). Class labels: T-shirt, Trouser, Pullover, Dress, Coat, Sandal, Shirt, Sneaker, Bag, Boot.

The silhouette-like patterns for clothing items (T-shirt, Dress, Coat) demonstrate that the eigenvector interpretability extends beyond digit recognition to more complex shape categories.

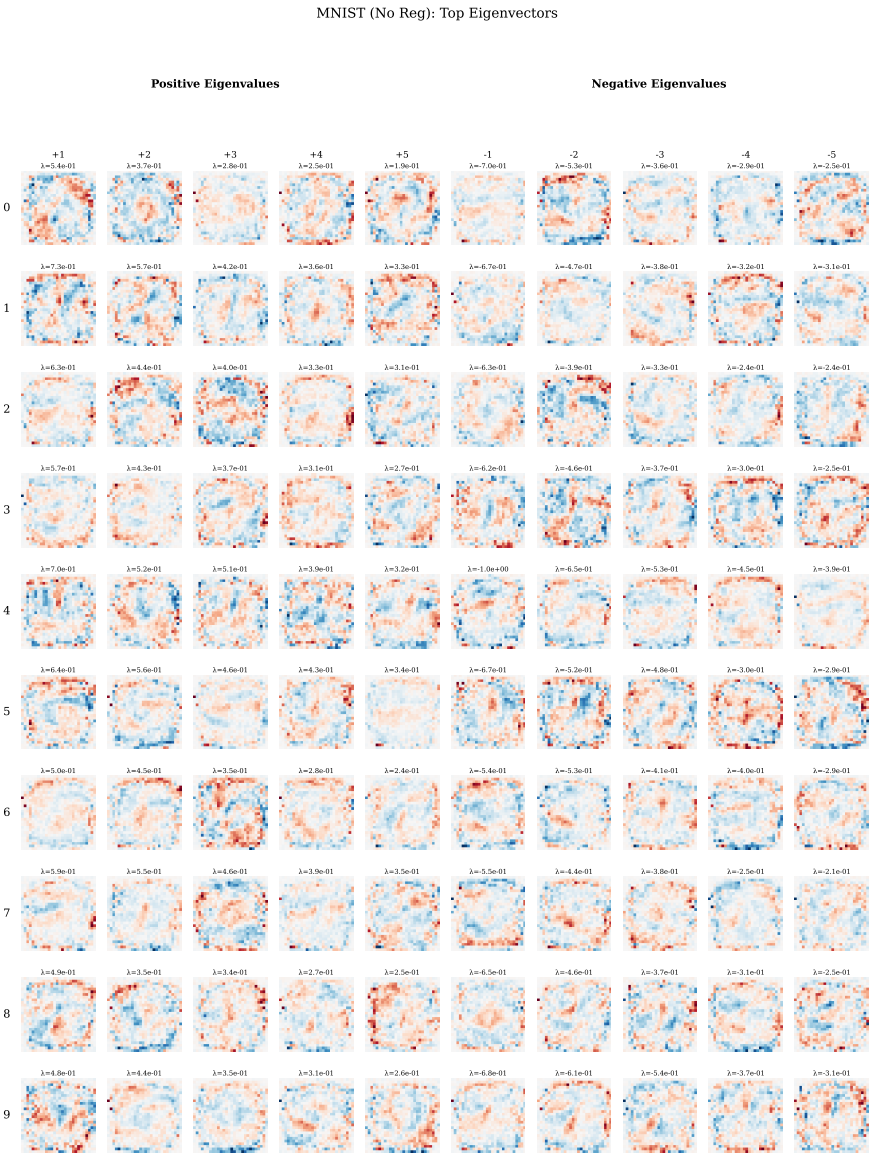


Figure 13: Top eigenvectors for each digit class (0–9) without regularization: diffuse, noise-like patterns (V3).

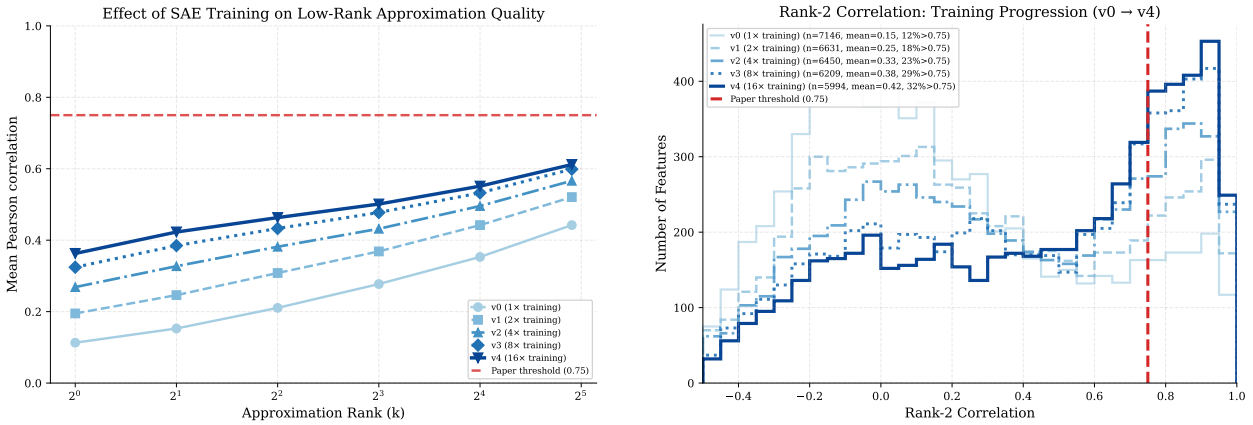
Figure 14: Top eigenvectors for each digit class (0–9) with regularization (noise $\sigma = 0.5$, weight decay $= 1.0$): recognizable digit strokes (V2).

C Language Experiment Results

Table 4: Language model configurations for correlation analysis (Figure 9).

Model	Layers	Params	SAE Layer	Expansion	k	Dataset
ts-medium	6	29M	4	4	30	TinyStories
fw-small	12	162M	8	4	30	FineWeb-EDU (FineWeb subset)
fw-medium	16	335M	10	8	30	FineWeb-EDU (FineWeb subset)

Figure 8 uses fw-medium at layer 7 with expansion 8 (public SAEs available at that layer). Figures 8 and 9 are presented in the main text.



(a) Correlation vs. SAE training time

(b) Distribution shift with training

Figure 15: Effect of SAE training duration on interpretability (fw-medium layer 12, expansion 16, active features only). Correlation distributions shift rightward with increased SAE training, consistent with the paper's qualitative claim that better-trained SAEs yield stronger low-rank predictability. This analysis is computed on a fixed FineWeb-EDU sample to enable a controlled comparison across SAE checkpoints.

D Sentiment Negation Circuit Investigation

This appendix presents our detailed investigation of AND-gate circuits in bilinear transformers.

D.1 SAE Availability Investigation

We investigated the publicly available SAEs on HuggingFace to determine reproducibility constraints.

ts-medium-scope (paper's "ts-tiny", 6 layers):

- Layers 0–4: Only `mlp-out`, `resid-mid`, `resid-pre` SAEs
- Layer 5: `mlp-in` and `mlp-out` available
- **Critical: No `mlp-in` SAE at layer 4** (the layer used in the paper's Figure 8)

fw-medium-scope (16 layers):

- All layers 0–15: `mlp-in` and `mlp-out` at 8× expansion
- Layer 12: Additional v0–v4 training-time variants at 16× expansion

Since Figure 8 analysis requires *both* `mlp-in` and `mlp-out` SAEs (for the Tracer class), the paper's Figure 8 (ts-tiny, layer 4) cannot be reproduced with public artifacts. We therefore use fw-medium (layer 7, expansion 8) which has both SAE types available. The paper's tutorial also uses fw-medium with features 3834/751, confirming this is a valid alternative.

D.2 AND-gate Circuit Analysis

The following analysis uses fw-medium (16 layers, 335M parameters, trained on FineWeb-EDU) with layer 7 SAEs at 8× expansion (8,192 output features).

D.2.1 AND-gate Circuit Definition

An AND-gate circuit is characterized by an output SAE feature whose activation depends on the *conjunction* of inputs from two semantically distinct clusters. Mathematically, for output feature c, the activation is:

$$z_c(x) = x^\top Q_c x \approx \lambda_1 (v_1^\top x)^2 + \lambda_2 (v_2^\top x)^2 \tag{8}$$

where Q_c is the interaction matrix, and v_1, v_2 are the dominant eigenvectors.

Strong AND-gate behavior requires:

1. **Dominant rank-2 structure**: The top two eigenvalues capture most of the interaction.
2. **Opposing eigenvalue signs**: λ_1 and λ_2 have opposite signs, creating cancellation when only one cluster is present.
3. **Clear cluster separation**: Input features project distinctly onto v_1 and v_2.

D.2.2 AND-score Metric

We define an "AND-score" to quantify circuit strength:

$$\text{AND-score} = \frac{|\lambda_1|}{|\lambda_1| + |\lambda_2|} \cdot \mathbf{1}[\text{sign}(\lambda_1) \neq \text{sign}(\lambda_2)] \tag{9}$$

Higher scores indicate stronger AND-gate behavior (score $\in [0, 1]$).

D.2.3 Search Procedure

We analyzed all 8,192 output features in fw-medium layer 7:

1. Computed the interaction matrix Q_c for each feature c
2. Performed eigendecomposition: $Q_c = \sum_j \lambda_j v_j v_j^\top$
3. Computed AND-score from top eigenvalues
4. Ranked features by AND-score
5. Analyzed top circuits for semantic interpretation

D.3 Results: Feature 3834 vs Feature 751

D.3.1 Tutorial Feature 3834

Feature 3834 (used in the paper's tutorial) encodes "not-bad" contexts with weaker AND-gate structure: diffuse eigenvalue spectrum and less distinct cluster separation.

D.3.2 Best Circuit: Feature 751

Our search identified feature 751 as having the strongest AND-gate structure (Table 6).

D.4 Semantic Interpretation of Feature 751

The input clusters have clear semantic content based on token analysis:

Positive cluster (3 features: 5212, 7655, 253):

- Feature 5212: "grateful", "opportunity", "Luckily", "thanks"
- Feature 7655: "beauty", "lovely", "amazing", "enjoy", "fun"
- Feature 253: "couldn't help" (compulsion/emotion contexts)
- Semantics: Positive sentiment/gratitude/appreciation

Negative cluster (5 features: 5201, 3601, 6051, 6921, 1003):

- Feature 5201: "unfortunate", "bad"
- Feature 3601: "problems", "disease"
- Feature 6051: "overly", "too narrow"
- Feature 6921: "no need"
- Feature 1003: "reducing", "stopping", "stop"

Circuit function: Feature 751 detects contexts where negative concepts appear *without* positive modulation. The AND-gate structure arises because:

1. The top eigenvalue (negative) suppresses activation when positive cluster projects onto v_1
2. The second eigenvalue (positive) activates when negative cluster projects onto v_2

3. When both clusters are present, the opposing signs cause partial cancellation

This confirms the mechanism: high activation requires negative concepts *without* positive sentiment, consistent with a "problematic situation" detector.

D.5 Linear Subspace Structure

Following the tutorial methodology, we compute cosine similarity between SAE decoder directions for the top output features with high eigenvalues. Features 3834 and 751 have cosine similarity -0.16—weakly negative—which the tutorial calls a "somewhat linear subspace." Note that -0.16 is closer to orthogonal than anti-parallel; the features are semantically contrasting (not-bad vs. not-good) rather than geometrically opposing. Semantic interpretation comes from inspecting top-activating tokens: feature 3834 fires on negated negative words, while feature 751 fires on negated positive words. The paper's tutorial observes: "It's not surprising that this forms a linear subspace as the two are opposites."

D.6 Cross-Dataset Validation

As additional validation beyond the paper, we test the same circuit on FineWeb-16k (the model's training distribution) in addition to TinyStories. Figure 16 shows the results, and Table 5 provides a quantitative comparison.

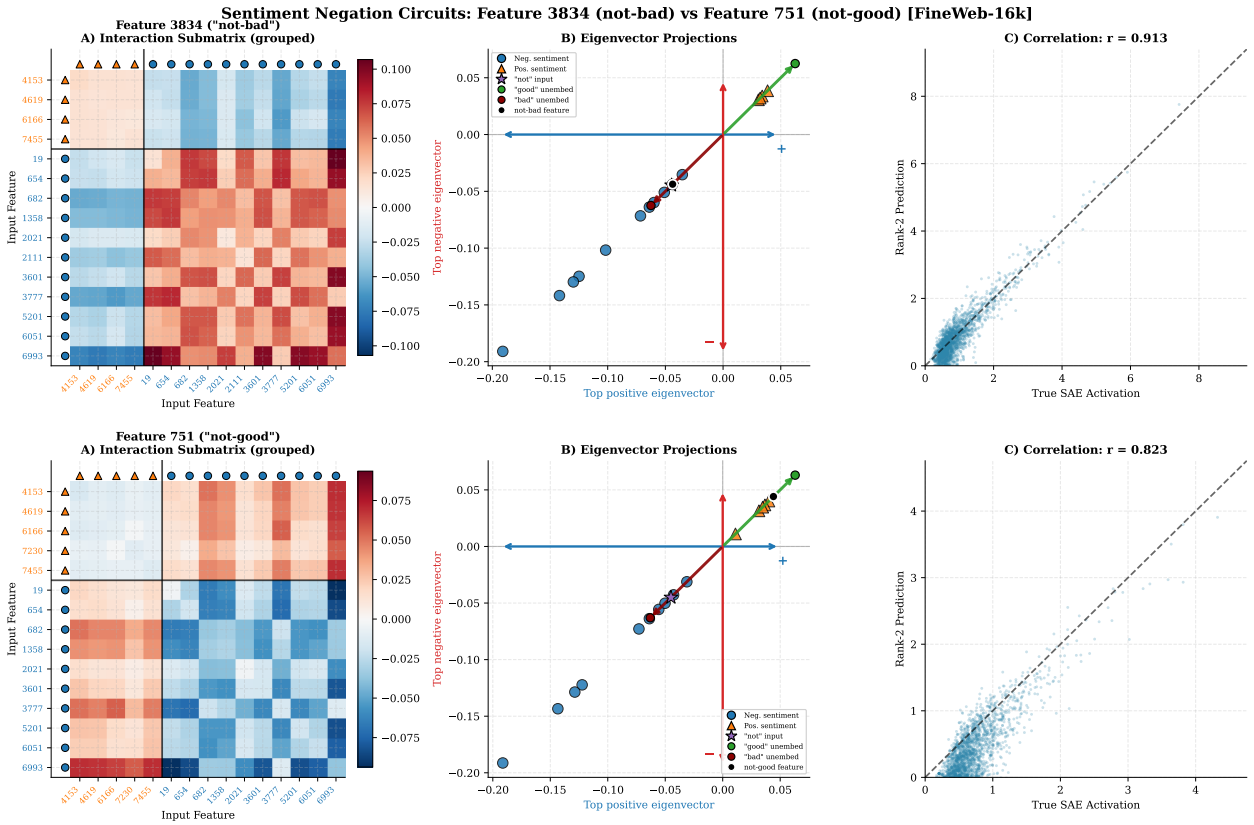


Figure 16: AND-gate circuit on FineWeb-16k (model's training distribution). Correlations: 0.91 (feature 3834), 0.82 (feature 751). Feature 751 activates more frequently on FineWeb-16k (6,517 vs 2,542 samples), likely due to more "not-good" contexts in educational/scientific text.

Table 5: Cross-dataset comparison of negation circuit correlations.

Feature	TinyStories	FineWeb-16k	Δ
3834 (not-bad)	0.931 (n=9,719)	0.913 (n=9,937)	−0.018
751 (not-good)	0.831 (n=2,542)	0.823 (n=6,517)	−0.008

The correlations remain strong across both datasets, with differences of only 1–2%. This demonstrates that the discovered circuit is robust to distribution shift and is a genuine property of the bilinear architecture, not a dataset artifact.

D.7 Summary

Table 6: Comparison of AND-gate circuit properties.

Property	**Feature 3834**	**Feature 751**
Rank-2 correlation (TinyStories)	0.931	0.831
Rank-2 correlation (FineWeb-16k)	0.913	0.823
Cosine similarity	−0.16 (weakly negative)	
Semantic interpretation	"not-bad"	"not-good"

Key findings: (1) Features 3834 and 751 are semantically contrasting negation features (not-bad vs. not-good) with weakly negative cosine similarity (−0.16). (2) The AND-gate circuit (negation ⊗ sentiment → negated-sentiment) is confirmed for both features. (3) Cross-dataset validation shows the circuit is robust to distribution shift (TinyStories vs. FineWeb-16k).

E Cross-Dataset Robustness Results

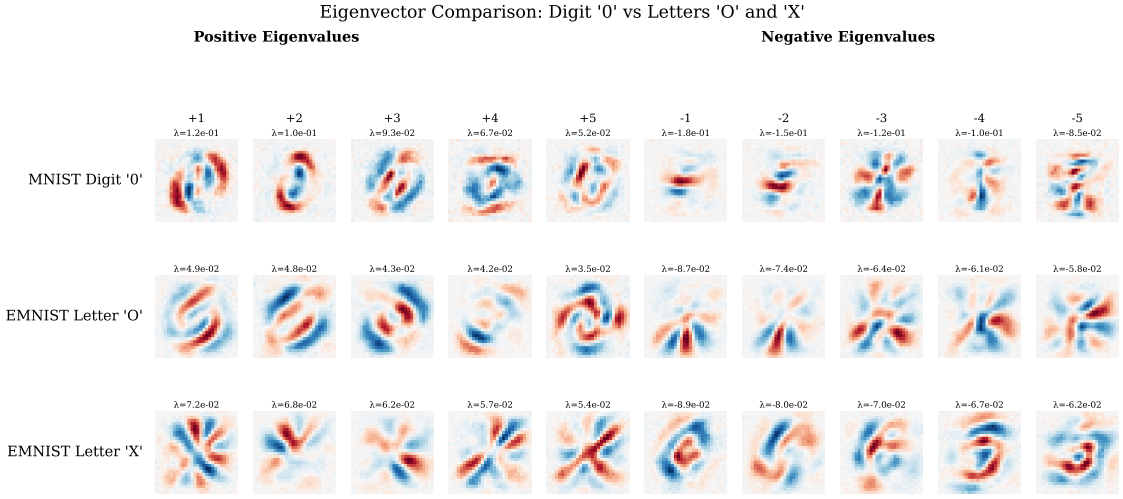


Figure 17: Three-way eigenvector comparison: MNIST-0, EMNIST-O, and EMNIST-X. The top eigenvectors for each class are visually distinct from one another.

Table 7: Cross-Dataset Accuracy: MNIST ↔ EMNIST-Digits (mean over 5 seeds).

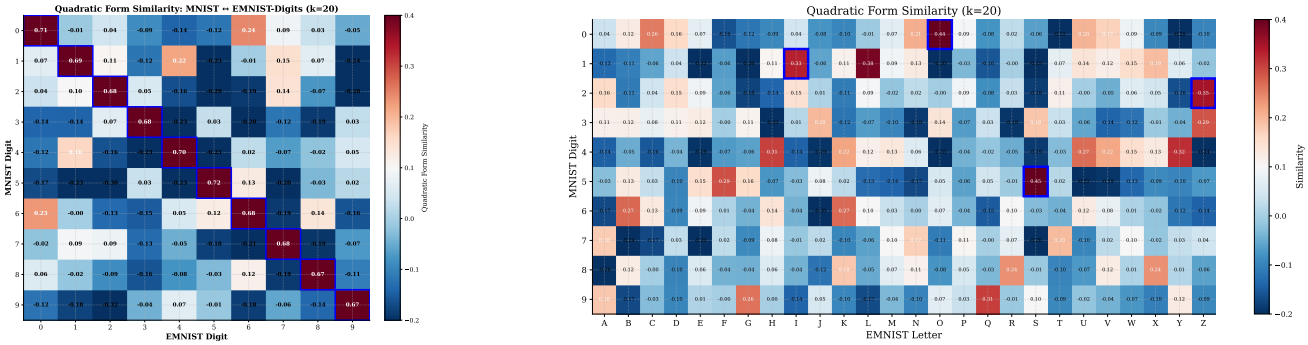
Direction	**Accuracy**
MNIST → EMNIST-Digits	97.527 ± 0.024%
EMNIST-Digits → MNIST	97.910 ± 0.026%
Bidirectional Average	97.719 ± 0.012%

Table 8: Cross-Dataset Accuracy: MNIST → USPS (mean over 5 seeds).

Model	**Accuracy**
Baseline (no reg)	53.42 ± 0.88%
Regularized	72.33 ± 0.83%

Table 9: Letter→Digit Transfer (regularized CoM model, 5 seeds).

Letter → Digit	**Accuracy**
O → 0	99.60 ± 0.05%
I → 1	87.08 ± 0.10%
Z → 2	88.75 ± 0.26%
S → 5	87.38 ± 0.22%



(a) MNIST vs EMNIST-Digits (b) MNIST vs EMNIST-Letters

Figure 18: Quadratic Form Similarity ($k = 20$). (a) Strong diagonal confirms writer-independent circuits. (b) Bright entries at shape-matched pairs (0–O, 1–I, 2–Z, 5–S) indicate mechanistically similar circuits; unrelated pairs are near zero or negative.

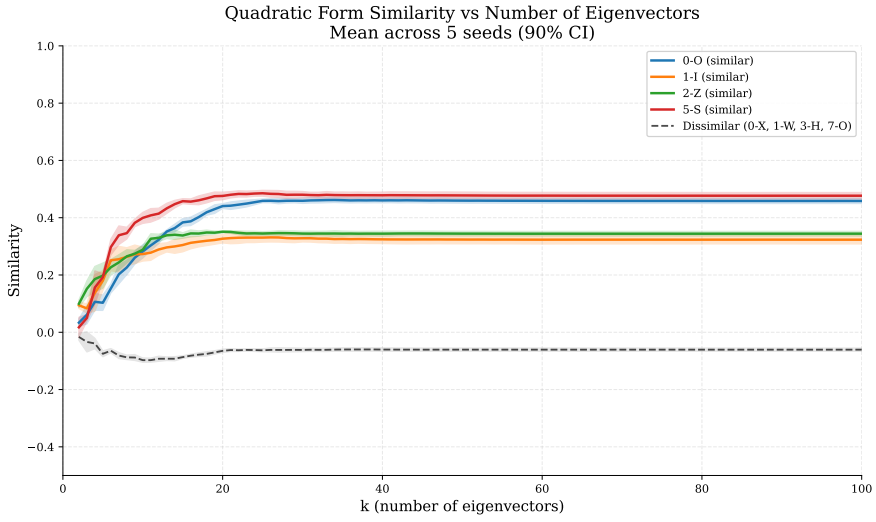


Figure 19: Quadratic Form Similarity vs number of eigenvectors k. Similar pairs converge around $k = 20$; dissimilar pairs stay near zero with non-overlapping 90% CIs.

Table 10: Ranking analysis: Position of expected letter among 26 EMNIST letters sorted by Quadratic Form Similarity. Random baseline: rank 13.

	0–O	1–I	2–Z	5–S	Mean
Rank	**1**	**2**	**1**	**1**	**1.2**

F Cosine Similarity Limitations

Figure 20 shows the absolute cosine similarity heatmap between MNIST digit classes and EMNIST letter classes. The heatmap demonstrates that cosine similarity fails to produce a clear distinction between geometrically similar and dissimilar class pairs. Figure 17 provides a three-way eigenvector comparison that further illustrates this limitation.

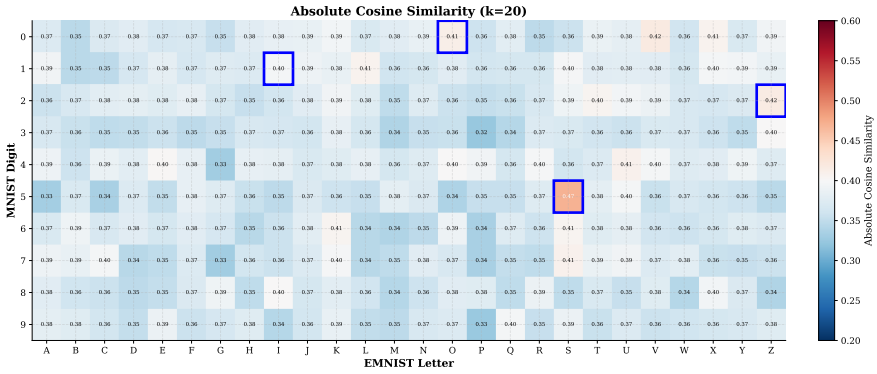


Figure 20: Absolute cosine similarity between MNIST digit classes and EMNIST letter classes fails to distinguish geometrically similar pairs (0–O) from dissimilar pairs (0–X).

Table 11 provides a detailed per-eigenvector breakdown, showing that the mean cosine similarity for 0–O (0.358) is nearly identical to 0–X (0.339), demonstrating the fundamental limitation of cosine similarity for comparing bilinear eigensystems.

Table 11: Absolute Cosine Similarity between Top 10 Eigenvectors (Balanced: 5 Positive + 5 Negative), Mean over 5 seeds

Eigenvector Rank	Eigenvalue Sign	Maximum Absolute Cosine Similarity	
		MNIST-0 vs EMNIST-O	MNIST-0 vs EMNIST-X
1	+	0.426 ± 0.083	0.337 ± 0.053
2	+	0.501 ± 0.105	0.713 ± 0.072
3	+	0.459 ± 0.069	0.539 ± 0.058
4	+	0.297 ± 0.049	0.267 ± 0.150
5	+	0.283 ± 0.033	0.256 ± 0.073
6	-	0.110 ± 0.072	0.364 ± 0.112
7	-	0.161 ± 0.036	0.368 ± 0.159
8	-	0.339 ± 0.092	0.319 ± 0.105
9	-	0.286 ± 0.051	0.225 ± 0.080
10	-	0.255 ± 0.110	0.248 ± 0.064
Mean		$\mathbf{0.312 \pm 0.016}$	$\mathbf{0.364 \pm 0.029}$

G Quadratic Form Similarity Derivation

For a bilinear MLP, the class-c logit is a quadratic form $y_c(x) = x^\top A_c x$ where $A_c = V \operatorname{diag}(\lambda_c) V^\top$.

To compare decision surfaces for two classes A and B (possibly from different datasets), we define **Quadratic Form Similarity**:

$$\text{sim}(A, B) = \frac{\text{tr}(A \cdot B)}{\|A\|_F \|B\|_F} \in [-1, 1]$$

For a rank-k truncation using top eigenmodes, this expands to:

$$\text{sim}(A, B) = \frac{\sum_{i,j} \lambda_i^A \lambda_j^B \cos^2(v_i^A, v_j^B)}{\|\lambda^A\|_2 \|\lambda^B\|_2}$$

Key properties solving the cosine similarity problem:

- Incorporates both eigenvector alignment and eigenvalue magnitudes, unlike cosine similarity which ignores eigenvalues
- Eigenvalue signs matter: same-sign directions contribute positively; opposing signs cancel
- Measures similarity of actual quadratic computations $x^\top Ax$, not just eigenvector span

This enables discrimination between geometrically similar pairs (0–O: 0.400) and dissimilar pairs (0–X: -0.060), where cosine similarity failed (0.358 vs. 0.339).

H CP-Decomposition Results

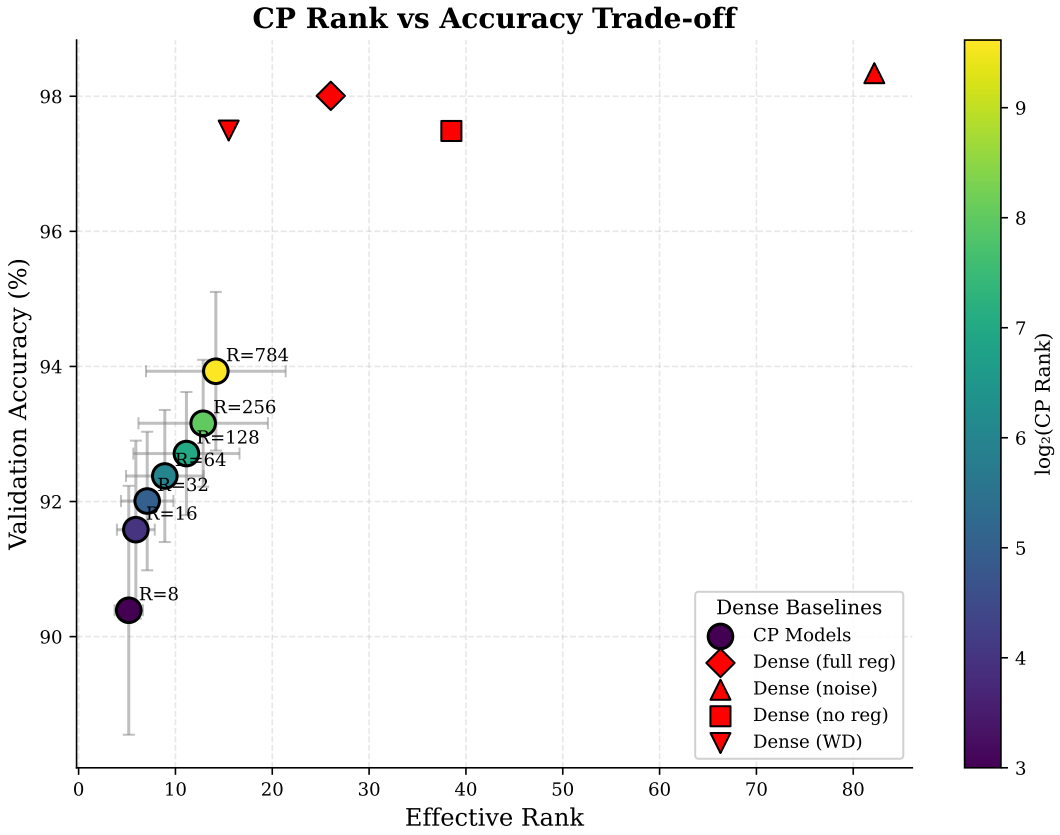


Figure 21: **Accuracy-Interpretability Pareto Frontier.** Fixed CP forces extreme sparsity (eff. rank ∼3.5) at accuracy cost. Lambda and Gated CP maintain competitive accuracy while reducing effective rank.

Table 12: CP-Decomposition results (mean ± std, 5 seeds). Best CP results in bold.

Model	Accuracy (%)	Eff. Rank
Dense (no reg)	97.49 ± 0.05	38.5 ± 0.7
Dense (WD only)	97.49 ± 0.05	15.5 ± 0.3
Fixed $R = 256$	91.89 ± 0.09	3.74 ± 0.07
Lambda $R = 256$	$\mathbf{93.80 \pm 0.14}$	$\mathbf{17.50 \pm 0.36}$
Gated $R = 256$	93.79 ± 0.15	17.35 ± 0.48

I Full CP-Decomposition Results

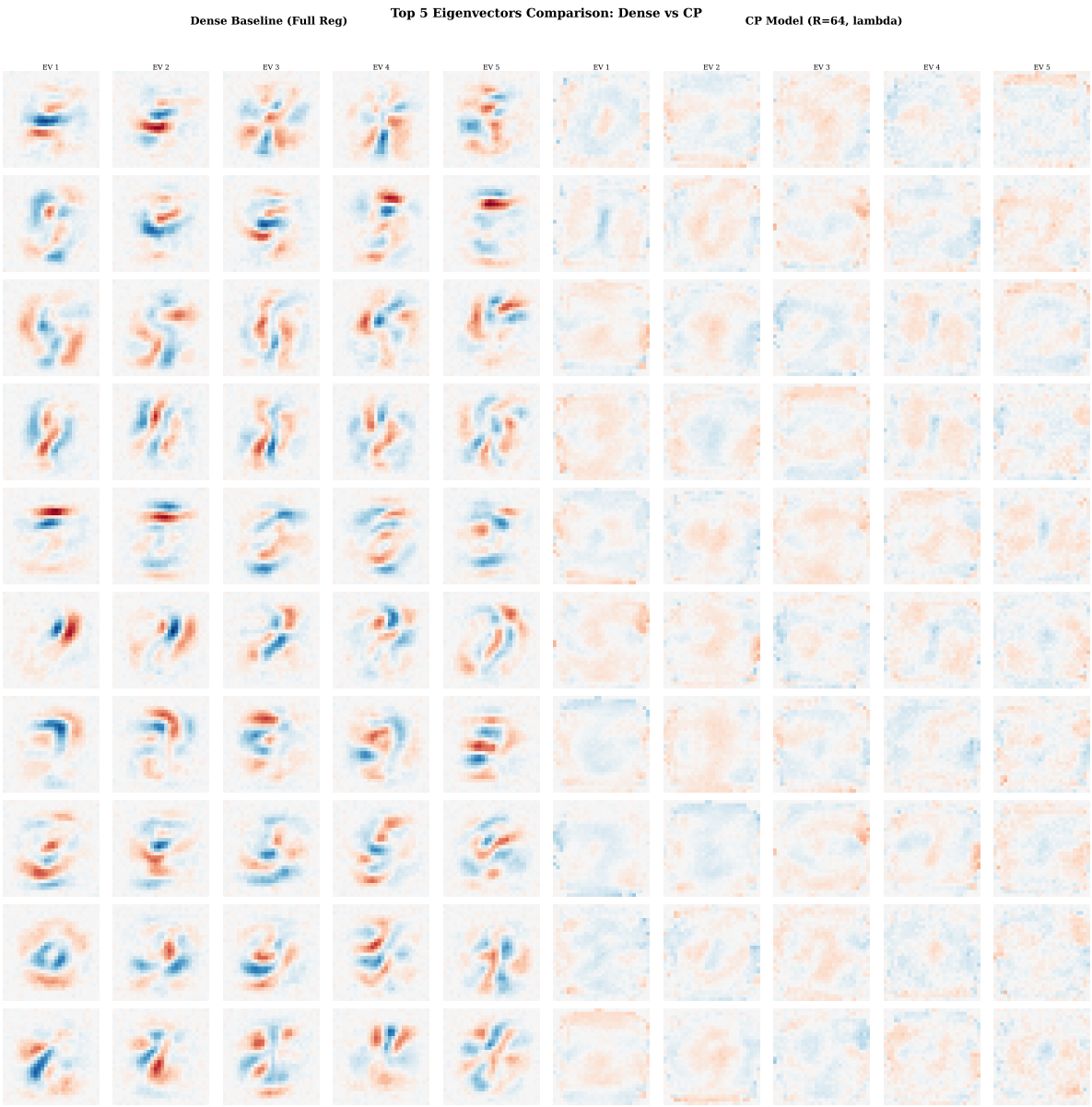


Figure 22: **Feature Disentanglement: Dense vs. CP.** Top 5 eigenvectors for digits 0–9. **Dense:** Holistic, superimposed templates mixing multiple strokes. **CP:** Atomic, localized factors (isolated curves, specific strokes) that can be combined to reconstruct digits.

Table 13: Complete CP-Decomposition rank sweep (mean ± std, 5 seeds).

Model	**Accuracy (%)**	**Eff. Rank**
Dense Baselines		
Dense (no reg)	97.49 ± 0.05	38.5 ± 0.7
Dense (WD only)	97.49 ± 0.05	15.5 ± 0.3
Fixed CP		
$R = 32$	90.63 ± 0.23	3.42 ± 0.22
$R = 64$	91.05 ± 0.13	3.45 ± 0.12
$R = 128$	91.48 ± 0.22	3.65 ± 0.05
$R = 256$	91.89 ± 0.09	3.74 ± 0.07
Lambda CP		
$R = 32$	92.67 ± 0.15	9.07 ± 0.44
$R = 64$	93.04 ± 0.10	11.57 ± 0.52
$R = 128$	93.34 ± 0.11	14.92 ± 0.49
$R = 256$	93.80 ± 0.14	17.50 ± 0.36
Gated CP		
$R = 32$	92.72 ± 0.20	8.76 ± 0.47
$R = 64$	93.05 ± 0.12	11.69 ± 0.35
$R = 128$	93.31 ± 0.14	14.83 ± 0.16
$R = 256$	93.79 ± 0.15	17.35 ± 0.48

J Environmental Impact

All experiments were tracked using `codecarbon` with Netherlands grid intensity. Hardware: Apple M4 (MPS) for all experiments. Total compute: 1.74 GPU-hours. Total emissions: 0.169 kg CO2eq (≈ 1 km driving).

Table 14: Environmental impact (CO2 via `codecarbon`).

Experiment	Runs	Wall (h)	GPU-h	CO2 (kg)
Vision (Section 4)[†]	101	1.66	1.11	0.040
Language (Section 5)[‡]	3	19.09	0.00	0.105
Extension 1 (Cross-Dataset)	15	0.58	0.58	0.020
Extension 2 (CP-Decomposition)	105	0.05	0.05	0.004
Total	**224**	**21.39**	**1.74**	**0.169**

[†]Includes 20 challenge task experiments (Figure 6) with full emissions tracking.
[‡]Language experiments run on Apple Silicon (MPS); GPU-hours reported as 0 since `codecarbon` only tracks NVIDIA GPUs.