Mechanistic Interpretability

Instructor: Prof. Shaker El Sappagh
course: Neural Networks

Name	ID
Mostafa Mohamed Saleh	223101188
Youssef Ahmed Ibrahim	223101109
Karim Mohamed Mostafa	223102240
Youssef Bassem Atef	223102042

Project Overview

Transcoders

- Activation based Analysis
- Transcoder Architecture
- Activations collection for Transcoder Training
- Visualize Transcoder Features
- Class-Specific Feature Analysis
- CONFUSION PAIRS

Bilinear MLP

- Weight based Analysis
- Bilinear Layer Implementation
- Eigendecomposition Analysis
- Visualize Eigenvalue Spectra
- Visualize Eigenvectors
- Adversarial Masks from Eigenvectors

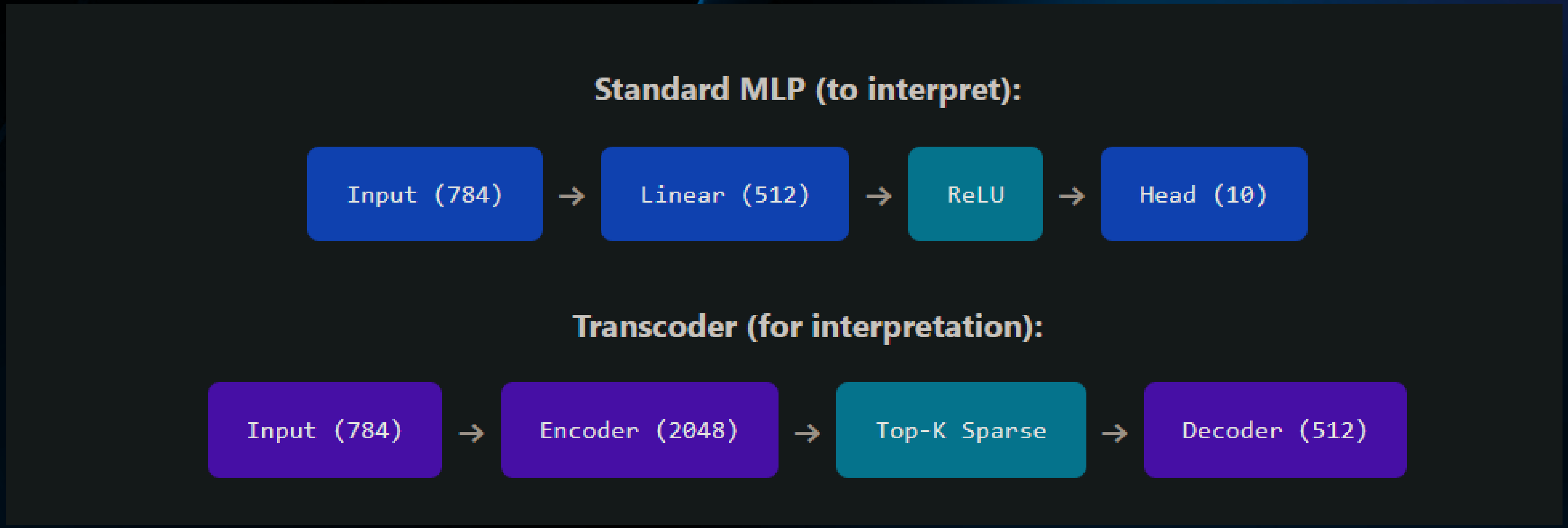
Activation vs Weight based Analysis

- What is analyzed: Activation-based methods analyze neuron activations at inference (Transcoders), while weight-based methods analyze learned parameters (Bilinear MLPs).
- Scope: Activation-based interpretability provides input-dependent, local explanations; weight-based interpretability provides global, model-level insights.
- Feature understanding: Transcoders reveal semantic features actually used for a given input; Bilinear MLPs expose explicit feature–feature interactions learned during training.
- Model behavior: Activation-based methods capture dynamic and context-dependent behavior; weight-based methods capture static structural relationships.

Base Models & Dataset

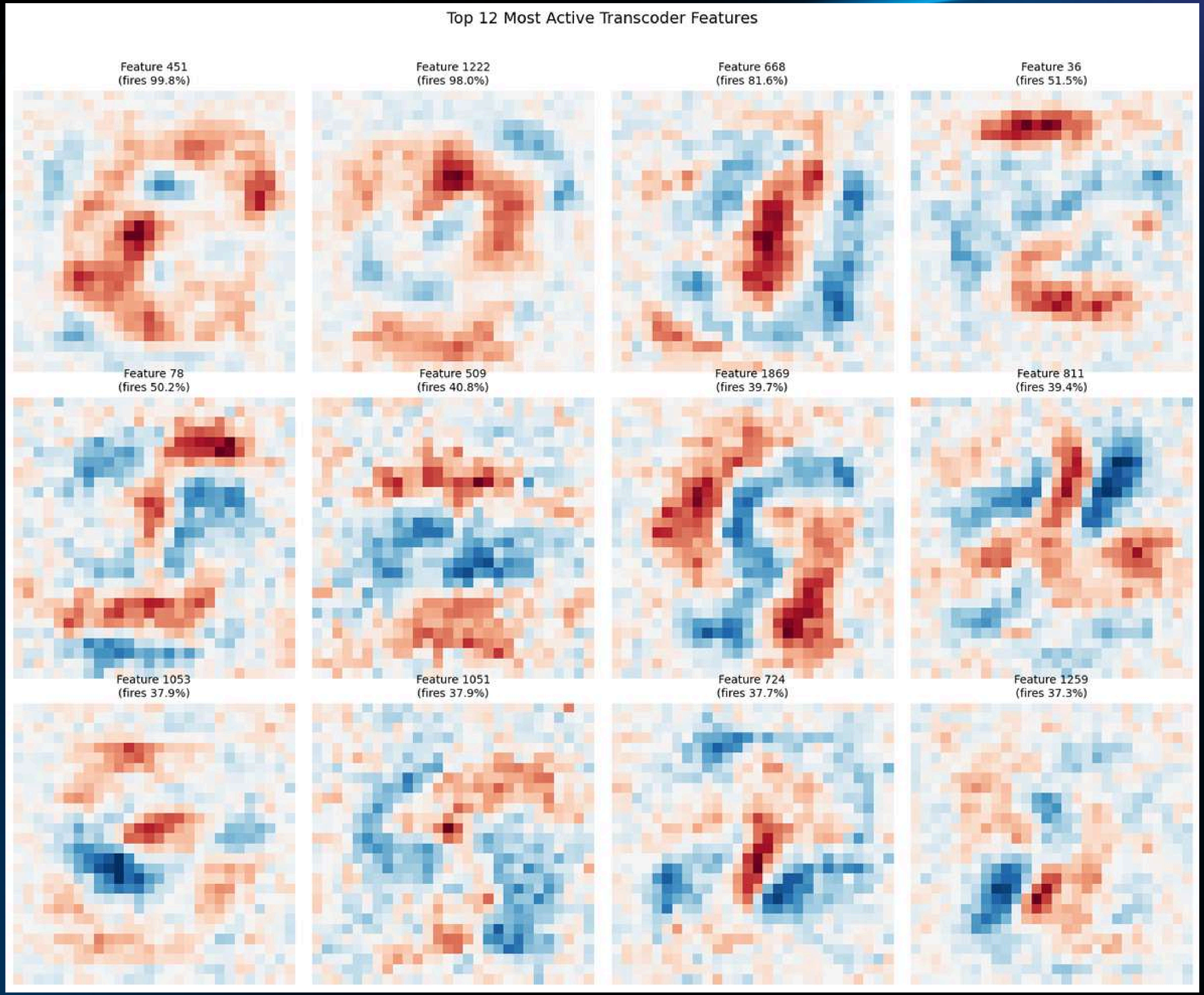
- Dataset: MNIST handwritten digits dataset (28×28 grayscale images, 10 classes: digits 0–9)
- Input Representation: Images flattened to 784-dimensional vectors
- Standard MLP Architecture: 784 → 512 → 10 Linear → ReLU → Linear
- Bilinear MLP Architecture: 784 → 512 → 10 Linear → Bilinear (W ⊙ V) → Linear

Transcoder architecture & Activation collection

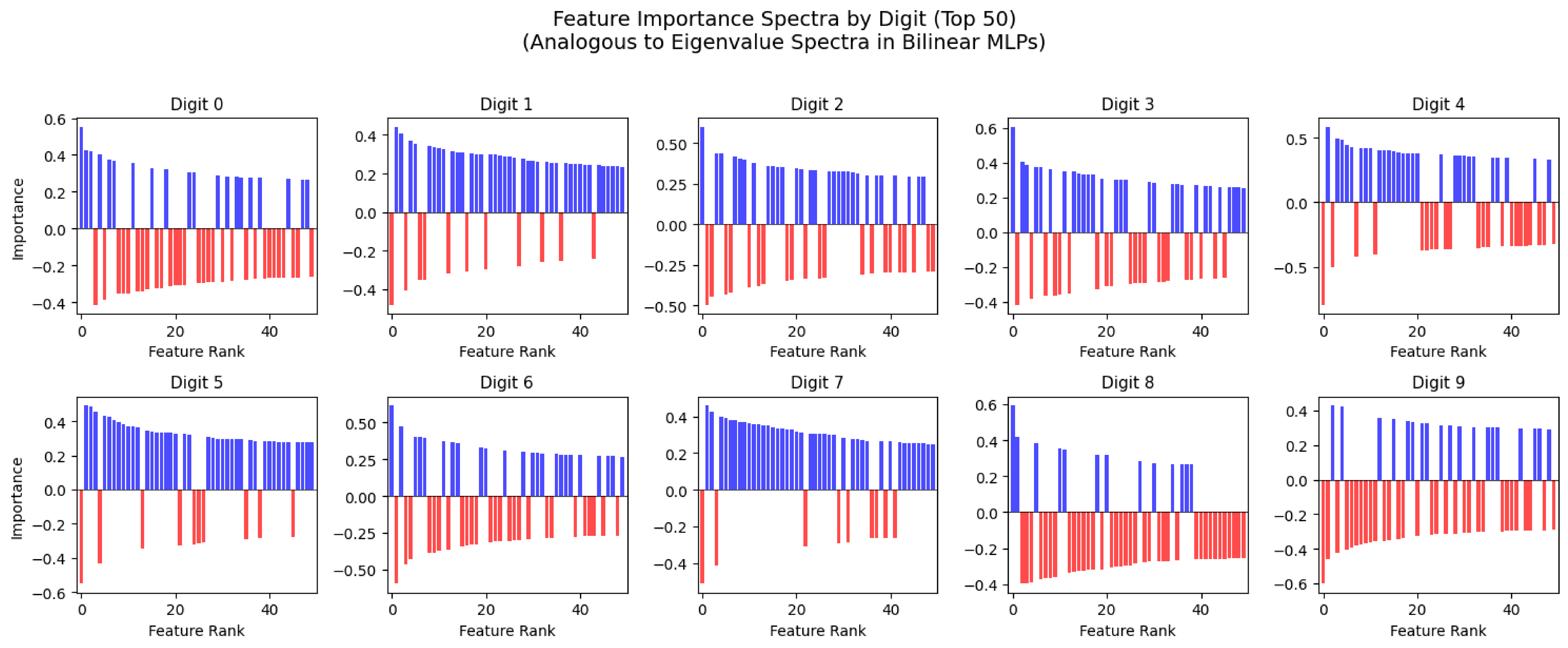


- Activation collection: Freeze the base model and run data through it to collect hidden-layer activations, which serve as the training data for the transcoder.
- Transcoder training: Train the transcoder to encode and reconstruct these activations, learning a mapping from raw activations to a more interpretable feature space without changing the main model.

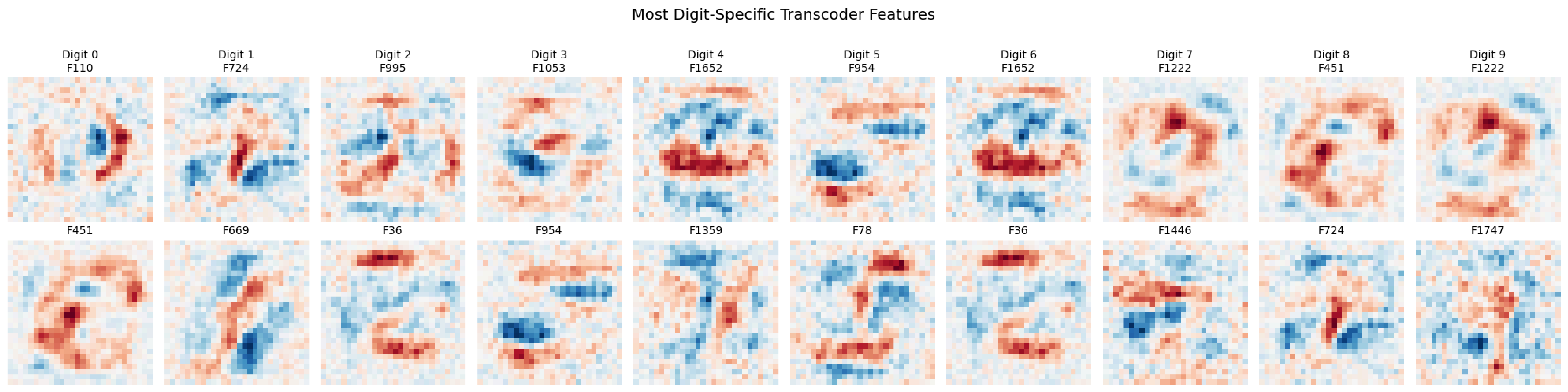
Visualize Transcoder Features



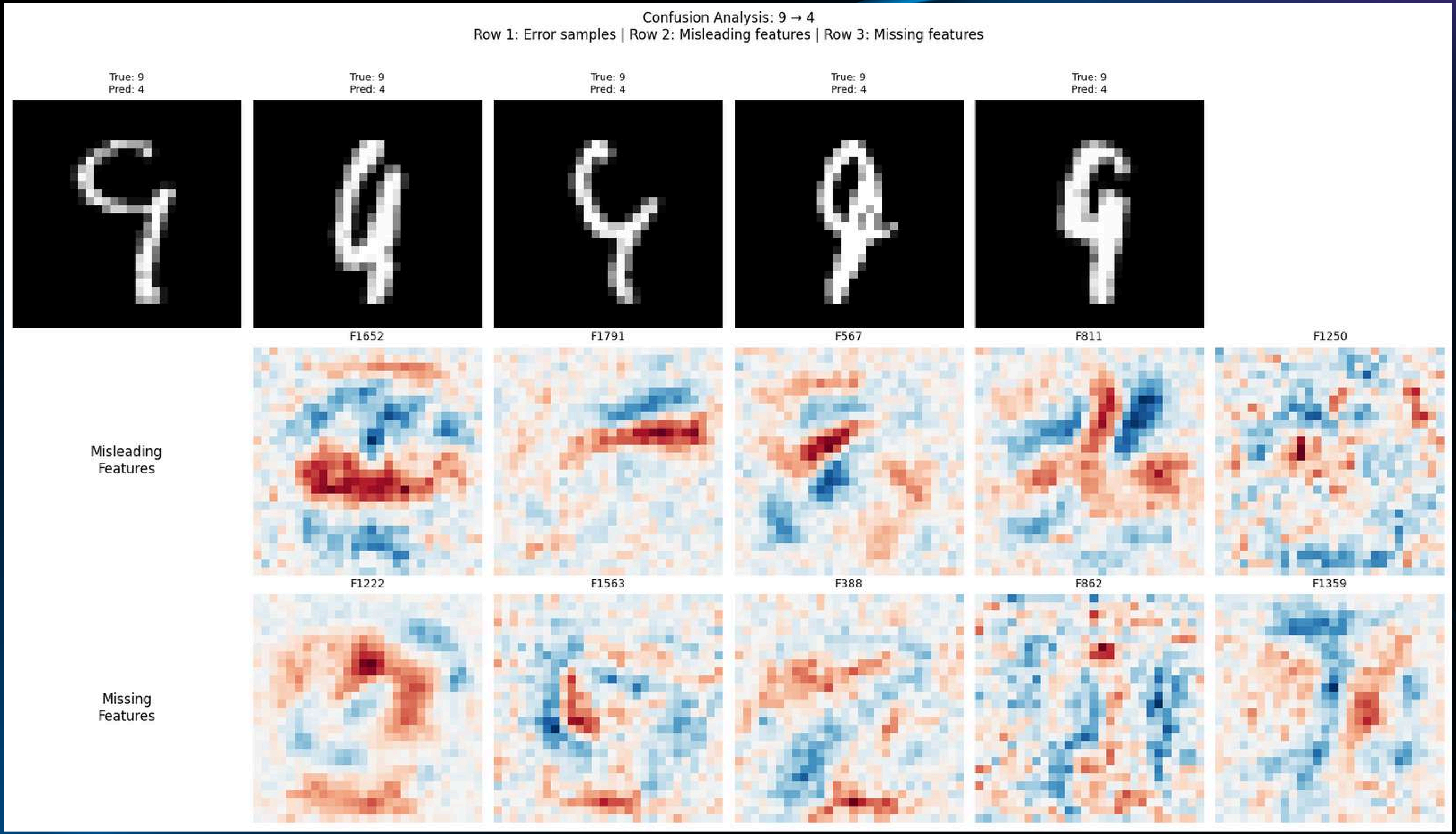
Visualize Transcoder Features



Class-Specific Feature Analysis



CONFUSION PAIRS



Bilinear MLPs: Weight-Based Interpretability

- Instead of applying a nonlinearity, the bilinear layer computes an element-wise product of two linear projections

$$
\begin{aligned}
g(\boldsymbol{x}) &= (\boldsymbol{W}\boldsymbol{x}) \odot (\boldsymbol{V}\boldsymbol{x}) \\
g(\boldsymbol{x})_a &= (\boldsymbol{w}_{a:}^T \boldsymbol{x})\,(\boldsymbol{v}_{a:}^T \boldsymbol{x}) \\
&= \boldsymbol{x}^T (\boldsymbol{w}_{a:} \boldsymbol{v}_{a:}^T) \boldsymbol{x}
\end{aligned}
$$

Mathematical Foundation

Interaction Matrix

For each output dimension *a*, we can express the computation as a quadratic form:

$$g(x)_a = x^T B_a x$$

$$\text{where } B_a = w_a v_a^T \text{ (outer product)}$$

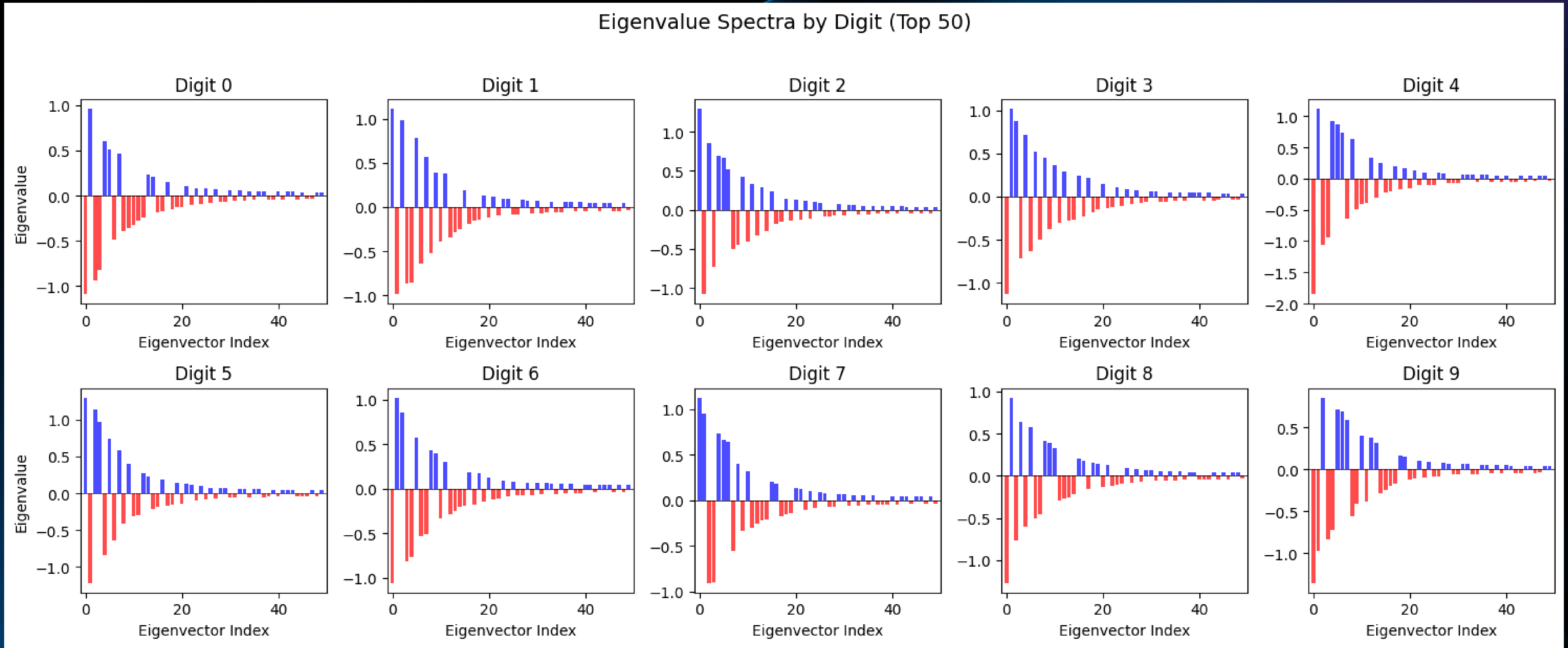
Eigendecomposition

Since B_a is symmetric (after symmetrization), we can decompose it into eigenvectors and eigenvalues:

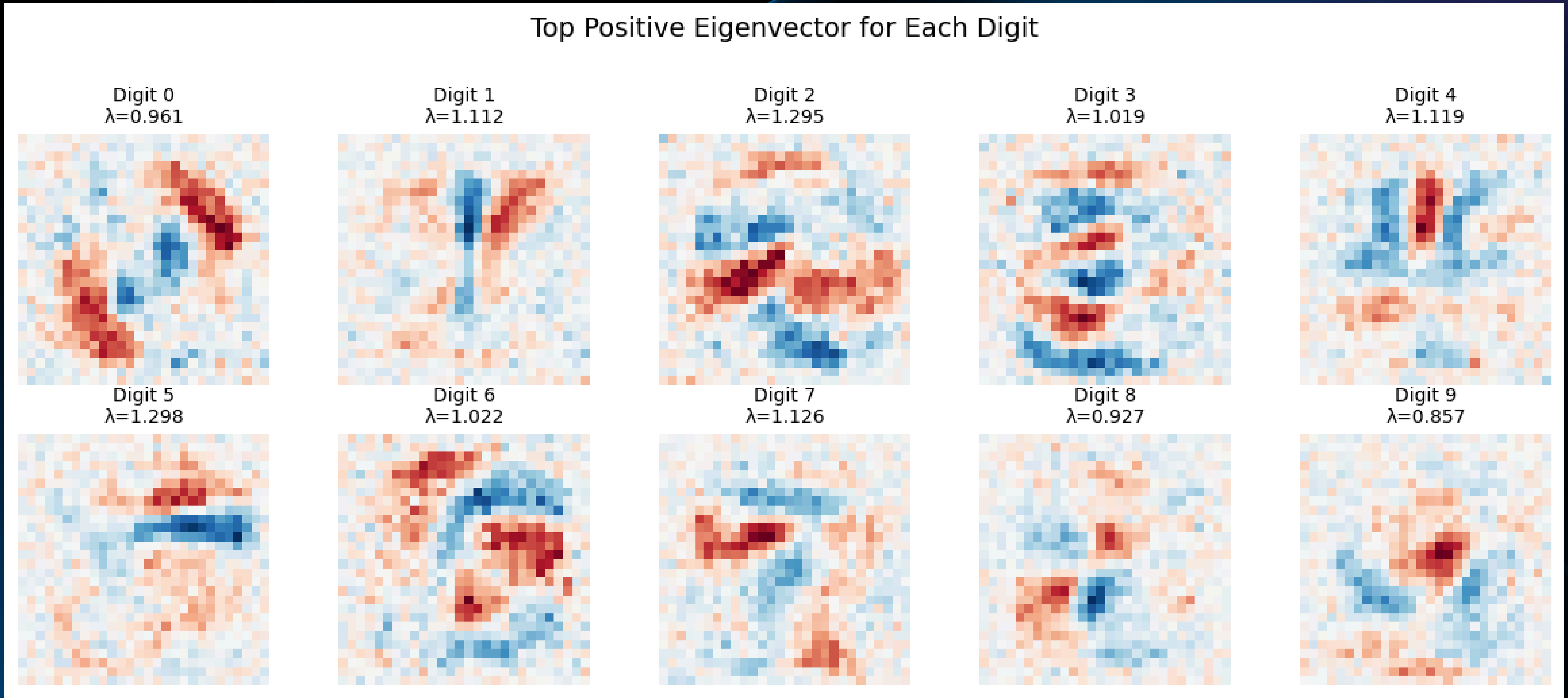
$$B = \sum_i \lambda_i v_i v_i^T$$

$$\text{Output: } x^T B x = \sum_i \lambda_i (v_i^T x)^2$$

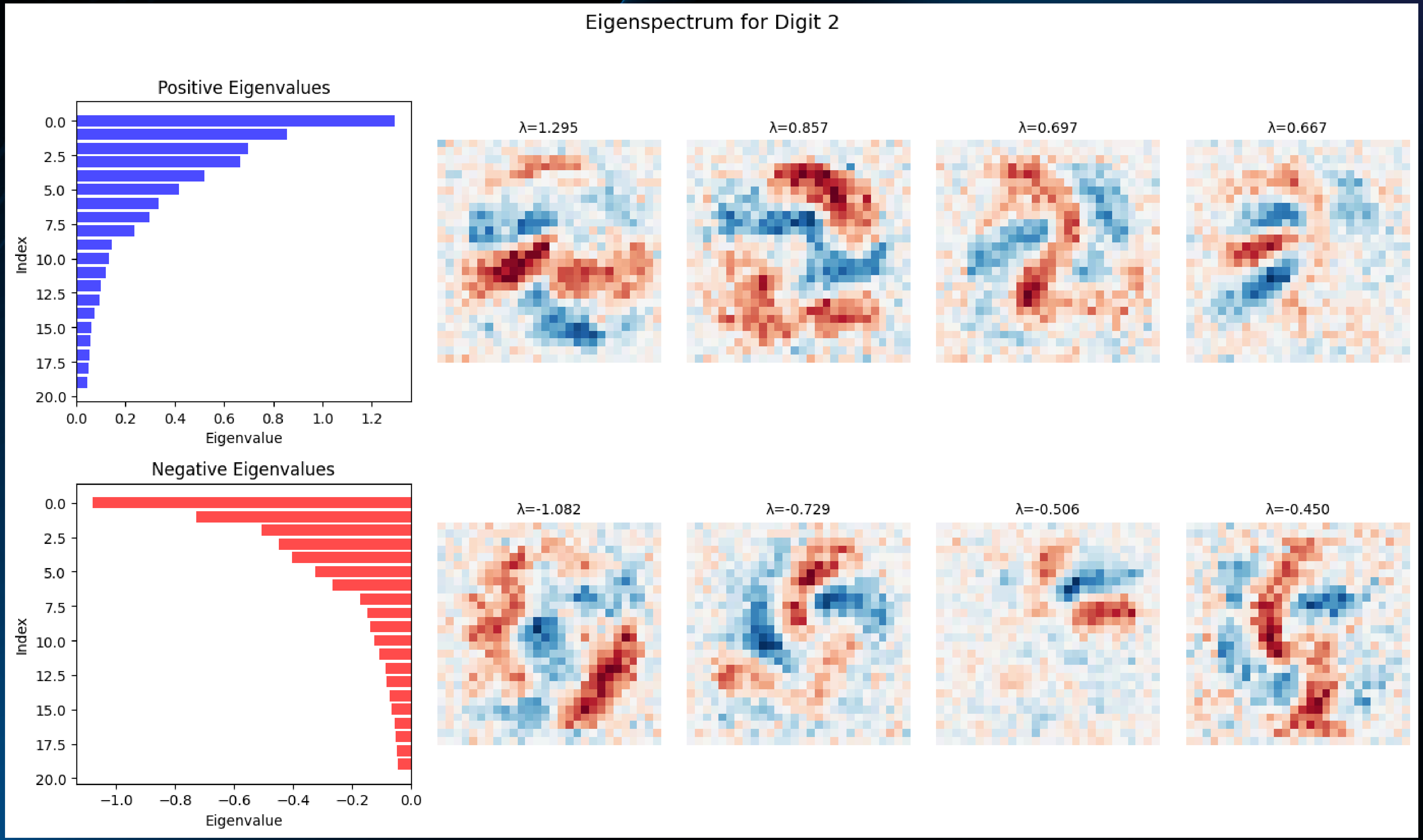
Visualize Eigenvalue Spectra



Visualize Eigenvectors



Visualize Eigenvectors



Transcoders

- Advantages

1. Works with any existing architecture
2. No changes to original model
3. Flexible sparsity control
4. Rich activation-based analysis

- Limitations

1. Requires additional training
2. Features are approximate, not exact
3. May miss some model behaviors

Bilinear MLPs

- Advantages

1. No extra training required
2. Exact mathematical interpretation
3. Features derived directly from weights
4. Can construct adversarial examples analytically

- Limitations

1. Requires architectural change
2. May have performance gap vs standard MLPs at scale
3. Only validated on smaller models

Side-by-Side Comparison

Aspect	Bilinear MLP	Transcoder
Interpretability Method	Eigendecomposition	Sparse Autoencoder
Feature Source	Eigenvectors	Encoder weights
Importance Measure	Eigenvalues	Decoder weights / activation frequency
Low-rank Structure	Yes (10-20 eigenvectors sufficient)	Yes (16-32 features sufficient)
Extra Training	No	Yes (transcoder)
Works with ReLU	No (requires bilinear)	Yes
Test Accuracy (MNIST)	~98%	~98%

Q &A