
Semantic Optimal Transport for Sparse Autoencoder Feature Matching and Circuit Compression

Tue M. Cao ^{*} [†]
caotue@ufl.edu

Nguyen Do ^{*} [†]
nguyen.do@ufl.edu

My T. Thai [†] [‡]
mythai@cise.ufl.edu

Abstract

Sparse autoencoders (SAEs) have become a central tool for interpreting language models. However, two key SAE analyses that remain difficult to scale are (1) matching semantically similar features across multi-layers and (2) compressing large feature circuits into interpretable supernodes. Although these have been treated as separate problems, we show that both are instances of a more fundamental challenge, which we frame as the estimation of semantic distances between SAE features that lie on different activation manifolds. We introduce a distributional framework for this problem, in which each feature is represented not by a single decoder vector like in the literature, but by an activation-weighted distribution over the hidden states that express it. By projecting these distributions into a shared reference space and comparing them with Wasserstein distance, our method provides a unified semantic metric for cross-layer feature comparison. We prove that our representation is invariant to activation rescaling, stable under perturbations, and recovers true matches under finite-sample margin conditions. Empirically, our method outperforms decoder-vector and LLM-based baselines and captures subtle functional distinctions between related features. Notably, our method compresses large feature circuits into interpretable supernodes *automatically*.

1 Introduction

Sparse Autoencoder (SAE) has shown great promises in understanding large language models (LLM) Cunningham et al. [2023], Gao et al. [2024] by breaking down the latent space into interpretable features. Many tools have been proposed, such as Feature Matching to understand feature persistence and evolution between layers Balagansky et al. [2024], Laptev et al. [2025], or tracing the “thoughts” of the LLM using Feature Circuits Marks et al. [2024], Dunefsky et al. [2024].

A standard tool of inspecting the thinking of a model is Feature Circuit Marks et al. [2024], a causal graph that connects features from SAEs spanned across all layers of a LLM. However, feature circuits are difficult to inspect at scale. A single circuit may contain hundreds feature nodes, each with distinct activation patterns and functional roles Lindsey et al. [2025]. Manually inspecting every node is impractical, while overly pruning can remove important intermediate mechanisms Marks et al. [2024]. In practice, researchers often compress circuits by manually grouping semantically similar feature nodes into higher-level “supernodes” Lindsey et al. [2025]. This makes the circuit easier to understand, but it relies on human labeling and hence not scalable. This motivates the need for an automatic method that can group SAE features by their semantic roles.

A separate line of works is Feature Matching Balagansky et al. [2024], Laptev et al. [2025], of which the goal is to understand feature transform across layers. These methods aim to match feature pairs with similar meaning from different layers, showing how the feature persist or evolve across layers.

^{*}Equal contribution

[†]Department of Computer and Information Science and Engineering, University of Florida

[‡]Corresponding Author

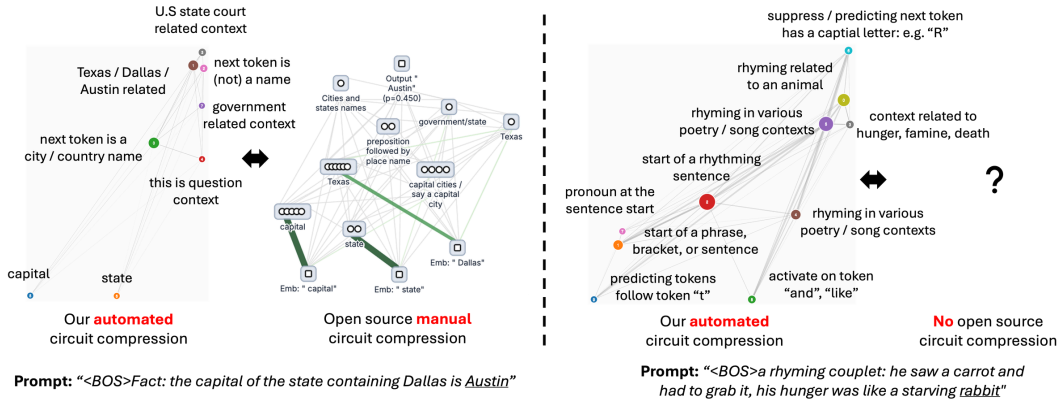


Figure 1: This figure compares our automated circuit compression and open source manual compression Lin [2023]. Our method identifies similar supernodes as open source version for "Dallas" circuit and scalable to other circuits that have no open source implementation. More details in Section 6.1.

A major flaw of the existing works is that their matching rely solely on feature decoder vectors of the SAEs trained on two different layers. This flaw is critical because the decoder vectors learned to reconstruct two different distributions. Therefore, their methods only work well on matching consecutive layers Balagansky et al. [2024] where the distributions are similar; and are inaccurate when the matching across dissimilar distributions, such as matching across multiple layers.

Although these two have been treated as separately, we identify that the core problem of both feature matching and circuit compression is approximating "semantic distance" of features across multi-manifolds. The "semantic distances" show how close the meaning of the features are, and thus, are the direct solution to both feature matching and circuit compressing. However, there are two key challenges to solve this problem. The first challenge is the different nature of the manifolds, where each manifold can have different support set and metrics, thus hindering the computation of across manifolds. The second problem is how to embed the "semantic" of features into the manifold. The main assumption of the field is to embed a feature by one vector (e.g. decoder vectors of SAEs) Balagansky et al. [2024], Bricken et al. [2023]. However, this approach lose information of the feature meaning and cannot be used with multi-manifolds. A comprehensive discussion of related literature is provided in the related works (Appendix B).

To address these challenges, we abandon the standard vector-centric paradigm and instead frame feature comparison as an Optimal Transport (OT) problem Beier et al. [2025], Imfeld et al. [2023], Melnyk et al. [2024]. By representing feature as activation-weighted distribution rather than one vector, we prevent the loss of semantic information that occurs when collapsing complex feature behaviors into a single point. Furthermore, OT provides a mathematically rigorous foundation for cross-manifold mapping, allowing us to estimate semantic distances between features even when they reside in different latent spaces. Hence, this distributional approach naturally facilitates both accurate cross-layer feature matching and feature circuit compression. Our theoretical analysis and empirical results show that:

- **Theoretical guarantees:** We prove that this distributional representation is invariant to activation rescaling and its Wasserstein distances are robust to measure perturbations. Furthermore, we establish explicit margin conditions guaranteeing that empirical top- K samples perfectly recover the true feature matches and semantic groups.
- **Cross Manifold Feature Matching:** Our method is more accurate than existing methods as well as LLM in matching features for both consecutive and across many layers. Notably, ours can identify the subtle differences in the meaning of many "digit addition" features Lindsey et al. [2025] while other failed.
- **Automatic circuit compression:** We can *automatically* compress feature circuits into intuitive supernodes, enhancing circuits' interpretability. We apply to automatically find modular features groups that have the same functional role in across many circuits.

2 Background and Problem Formulation

Sparse Autoencoders. Let $x^{(\ell)} \in \mathbb{R}^{d_\ell}$ denote the hidden state of a language model at layer ℓ . An SAE trained on layer ℓ represents $x^{(\ell)}$ using an overcomplete sparse feature vector $a^{(\ell)}(x^{(\ell)}) \in \mathbb{R}^{F_\ell}$, $F_\ell \gg d_\ell$. The encoder and decoder are given by

$$a^{(\ell)}(x^{(\ell)}) = \sigma\left(W_{\text{enc}}^{(\ell)}\left(x^{(\ell)} - b_{\text{enc}}^{(\ell)}\right)\right), \quad \hat{x}^{(\ell)} = W_{\text{dec}}^{(\ell)}a^{(\ell)}(x^{(\ell)}) + b_{\text{dec}}^{(\ell)}. \quad (1)$$

Here, $W_{\text{enc}}^{(\ell)} \in \mathbb{R}^{F_\ell \times d_\ell}$, $W_{\text{dec}}^{(\ell)} \in \mathbb{R}^{d_\ell \times F_\ell}$ are the encoder and decoder weights, and $b_{\text{enc}}^{(\ell)} \in \mathbb{R}^{d_\ell}$, $b_{\text{dec}}^{(\ell)} \in \mathbb{R}^{d_\ell}$ are the encoder and decoder biases. The function σ is a sparsity-inducing activation function, such as TopK Gao et al. [2024]. We denote $a_i^{(\ell)}(x^{(\ell)})$ for the activation of feature $i \in \mathcal{F}^{(\ell)}$ on hidden state $x^{(\ell)}$. Note that, throughout this work, we consider nonnegative SAE activations $a_i^{(\ell)}(x^{(\ell)}) \geq 0$.

Empirical activation distributions. Let $\mathcal{S}_T = \left\{(x_t^{(1)}, \dots, x_t^{(L)})\right\}_{t=1}^T$ denote a corpus of T token positions, where $x_t^{(\ell)}$ is the hidden state of token position t at layer ℓ and L is the total number of layers in the language model. For a feature $i \in \mathcal{F}^{(\ell)}$, define its top- K activation index set by $\mathcal{I}_{i,T}^{(\ell),K} = \text{TopK}_{t \in [T]} a_i^{(\ell)}(x_t^{(\ell)})$. Throughout the paper, TopK denotes a deterministic selector of K best candidates. For each selected token position $t \in \mathcal{I}_{i,T}^{(\ell),K}$, define the normalized activation weight $\hat{w}_{i,t}^{(\ell)} = a_i^{(\ell)}(x_t^{(\ell)}) / \sum_{s \in \mathcal{I}_{i,T}^{(\ell),K}} a_i^{(\ell)}(x_s^{(\ell)})$. The activation-weighted empirical distribution of feature i at layer ℓ is $\hat{\mu}_{i,T}^{(\ell)} = \sum_{t \in \mathcal{I}_{i,T}^{(\ell),K}} \hat{w}_{i,t}^{(\ell)} \delta_{x_t^{(\ell)}}$. Here, $\delta_{x_t^{(\ell)}}$ is the Dirac measure at the hidden state $x_t^{(\ell)}$. Thus, $\hat{\mu}_{i,T}^{(\ell)}$ records both the token positions where feature i activates most strongly and the relative activation strengths over those positions.

Problem Definition: Cross-Manifold Semantic Distance. A fundamental problem is to estimate the semantic distance between SAE features, once the distance is available, downstream tasks such as: cross-layer feature matching can be solved by selecting nearest neighbors, or circuit compression can be solved by grouping circuit nodes are close.

However, there are two key challenges: (1) features from different layers reside on distinct manifolds the meaning, compute the distance using decoder vectors trained of different manifolds Balagansky et al. [2024], Laptev et al. [2025] would yield inaccurate result, (2) many works Cunningham et al. [2023], Cho and Hockenmaier [2025], Laptev et al. [2025] consider the semantic of the features as one vector (e.g. decoder vectors), this approach loses the information of activation context.

To solve these challenges, we first define the problem as computing the *cross-manifold* semantic distance of *probability distributions* of features. The following formulation requires approximating true cross-manifold distance and preserving the semantic activation context of the features:

Let $\{\hat{\mu}_{i,T}^{(\ell)} : i \in \mathcal{F}^{(\ell)}\}$ denote the empirical activation distributions of SAE features at layer ℓ lying on a feature-semantic manifold $\mathcal{M}^{(\ell)} \subseteq \mathcal{P}(\mathcal{X}^{(\ell)})$; where $\mathcal{P}(\mathcal{X}^{(\ell)})$ denotes the space of probability distributions over the hidden-state space $\mathcal{X}^{(\ell)} \subseteq \mathbb{R}^{d_\ell}$. Ideally, there exists a true but unknown semantic distance between any two features, which we refer to as the oracle distance: $D_{\ell q}^*(i, j)$, $i \in \mathcal{F}^{(\ell)}$, $j \in \mathcal{F}^{(q)}$. This oracle distance is small when two features play similar semantic or functional roles and large when their activation behavior differs. Since $D_{\ell q}^*$ is not directly observable, we seek a computable discrepancy $\hat{D}_{\ell q} : \mathcal{M}^{(\ell)} \times \mathcal{M}^{(q)} \rightarrow \mathbb{R}_{\geq 0}$ that approximates this oracle semantic distance. Conceptually, this corresponds to minimizing:

$$\min_{\hat{D}} \sum_{\ell, q=1}^L \sum_{i \in \mathcal{F}^{(\ell)}} \sum_{j \in \mathcal{F}^{(q)}} \omega_{ij}^{(\ell, q)} \left[\hat{D}_{\ell q} \left(\hat{\mu}_{i,T}^{(\ell)}, \hat{\mu}_{j,T}^{(q)} \right) - D_{\ell q}^*(i, j) \right]^2, \quad (2)$$

where $\omega_{ij}^{(\ell, q)} \geq 0$ optionally weights feature pairs of interest.

3 Methodology

Given the formulation in Section 2, to compute distance across manifolds, we first project these layer-specific empirical distributions onto a shared reference space $\mathcal{R}$. Furthermore, to preserve

semantic information, we compute the semantic distance between the features' empirical distributions using the Wasserstein distance. This unified pairwise distance then serves as the foundation for downstream tasks, including cross-layer feature matching and circuit compression.

3.1 Shared Reference Space and Wasserstein Distance

To make feature activation distributions comparable across layers, we assign each token position a representation in a shared reference space $\mathcal{R}$. For a feature $i \in \mathcal{F}^{(\ell)}$ with top- K index set $\mathcal{I}_{i,T}^{(\ell),K}$ and normalized activation weights $\hat{w}_{i,t}^{(\ell)}$, we define its projected feature distribution $\hat{\mu}_{i,T}^{(\ell \rightarrow \mathcal{R})} = \sum_{t \in \mathcal{I}_{i,T}^{(\ell),K}} \hat{w}_{i,t}^{(\ell)} \delta_{z_t^{(\mathcal{R})}}$ in the shared reference space $\mathcal{R}$ by replacing each layer-specific support point $x_t^{(\ell)} \in \mathcal{X}^{(\ell)}$ with the shared-space representation $z_t^{(\mathcal{R})} \in \mathcal{R}$. This new empirical distribution on shared space still defines the feature's activation profile: it chooses the contexts and their relative weights. The practical choice of reference space $\mathcal{R}$ are detailed in Section 3.2. For readability, when comparing a target feature $i \in \mathcal{F}^{(A)}$ with a source feature $j \in \mathcal{F}^{(B)}$, we write $\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}$ for the projected target distribution and $\hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}$ for the projected source distribution.

Once two features are represented as probability distributions over the shared reference space $\mathcal{R}$, we compare them using OT. Let $c : \mathcal{R} \times \mathcal{R} \rightarrow \mathbb{R}_{\geq 0}$ be a generic ground cost on $\mathcal{R}$. For a target feature $i \in \mathcal{F}^{(A)}$ and a source feature $j \in \mathcal{F}^{(B)}$, we define their semantic distance as:

$$\hat{D}_{\mathcal{R}}(i, j) = W_c(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}), \quad W_c(\mu, \nu) = \inf_{\gamma \in \Pi(\mu, \nu)} \int_{\mathcal{R} \times \mathcal{R}} c(z, z') d\gamma(z, z'). \quad (3)$$

Here, $\Pi(\mu, \nu)$ is the set of couplings between μ and ν . The ground cost $c(z, z')$ measures how expensive it is to match two selected context representations $z, z' \in \mathcal{R}$, while the coupling γ specifies how probability mass from one feature distribution is transported to the other.

For the empirical distributions used in our method, this Wasserstein problem becomes a finite discrete transport problem. For a fixed feature pair (i, j) , the pairwise cost matrix over their selected top- K contexts has entries $C_{pq}^{(i,j)} = c(z_p^{(\mathcal{R})}, z_q^{(\mathcal{R})})$, where $p \in \mathcal{I}_{i,T}^{(A),K}$ indexes a selected context of the target feature and $q \in \mathcal{I}_{j,T}^{(B),K}$ indexes a selected context of the source feature. A discrete transport plan is a nonnegative matrix whose row and column sums match the normalized activation weights: $\Pi_{ij} = \left\{ \pi \geq 0 : \sum_{q \in \mathcal{I}_{j,T}^{(B),K}} \pi_{pq} = \hat{w}_{i,p}^{(A)} \forall p \in \mathcal{I}_{i,T}^{(A),K}, \sum_{p \in \mathcal{I}_{i,T}^{(A),K}} \pi_{pq} = \hat{w}_{j,q}^{(B)} \forall q \in \mathcal{I}_{j,T}^{(B),K} \right\}$. Therefore, the empirical Wasserstein feature distance is $\hat{D}_{\mathcal{R}}(i, j) = \min_{\pi \in \Pi_{ij}} \sum_{p \in \mathcal{I}_{i,T}^{(A),K}} \sum_{q \in \mathcal{I}_{j,T}^{(B),K}} \pi_{pq} C_{pq}^{(i,j)}$. This objective quantifies the minimum cost required to align the activation-weighted supports of the two features in $\mathcal{R}$. It serves as a purely geometric measure of semantic similarity, however, our overall framework remains entirely agnostic to the specific choice of the ground cost c . In Section 4.1, we analyze the stability of the Wasserstein score for a general cost c satisfying coordinate-wise Lipschitz conditions. Appendix E.4 instantiates this general result for several concrete score choices and compares their stability constants.

3.2 Feature Matching and Circuit Compression

We provide our choice of reference space $\mathcal{R}$ for feature matching and circuit compression problem, the full procedure for each application is provided in Algorithm 1 and 2 respectively:

Cross-layer feature matching: Given a target feature set $\mathcal{F}^{(A)}$ at layer A and source feature set $\mathcal{F}^{(B)}$ at layer B , we match each target feature to its nearest source feature:

$$\hat{M}_{A \rightarrow B}(i) = \arg \min_{j \in \mathcal{F}^{(B)}} \hat{D}_{\mathcal{R}}(i, j). \quad (4)$$

In practice, we choose the reference space $\mathcal{R}$ as the space of target layer A , where for each activation token of feature at layer B , we extract the representation at the same token in layer A . This method leverages the natural alignment via tokens between the layers without any computational overhead.

Circuit compression: Let $G = (V, E)$ be a feature circuit, where each node $u = (\ell, i) \in V$ corresponds to feature i at layer ℓ . We want to find a set of supernodes $\mathbb{C} = \{\mathcal{C}_1, \dots, \mathcal{C}_m\}$, each

supernode contains feature nodes of V , and $C_i \cap C_j = \emptyset$, $i \neq j$. We choose the partition by minimizing within-supernode semantic distance:

$$\hat{\mathbb{C}} = \arg \min_{\mathbb{C}} \sum_{C \in \mathbb{C}} \frac{1}{|C|(|\mathbb{C}| - 1)} \sum_{u, v \in C, u \neq v} \hat{D}_{\mathcal{R}}(u, v). \quad (5)$$

In practice, this objective can be optimized with standard clustering procedures using the pairwise distance matrix over circuit nodes. We choose the shared space $\mathcal{R}$ as the concatenation of hidden state of all layers. Specifically, for a feature activates on token t , the representation on $\mathcal{R}$ is $z_t^{(\mathcal{R})} = \text{Concat}(x_t^{(1)}, x_t^{(2)}, \dots, x_t^{(L)})$. This concatenation help the $\mathcal{R}$ to preserve the semantic across layers, leading to more accurate circuit compression.

4 Theoretical Analysis

We establish three theoretical guarantees for our framework (proofs in Appendix E). First, the projected feature distribution is strictly invariant to feature activation scaling. Second, the Wasserstein distances are stable under measure perturbations, with variations bounded by the Lipschitz constants of the ground cost, justifying the stability of the distances. Finally, we provide margin-based recovery conditions, proving that true feature matches and circuit compression are perfectly preserved whenever the relevant Wasserstein or Voronoi margin exceeds the estimation error.

4.1 Representation and Score Stability

Although features are compared in the shared reference space $\mathcal{R}$, their projected distributions are constructed from layer-specific SAE activations. The representation should therefore depend on which contexts a feature selects and their relative activation weights, not on the arbitrary numerical scale of the activations.

Proposition 1 (Activation rescaling invariance). *Let $i \in \mathcal{F}^{(A)}$ have activation function $a_i^{(A)} : \mathcal{X}^{(A)} \rightarrow \mathbb{R}_{\geq 0}$. For any $\alpha > 0$, define $\tilde{a}_i^{(A)}(x) = \alpha a_i^{(A)}(x)$. Under deterministic TopK selection, $\tilde{\mu}_{i,T}^{(A \rightarrow \mathcal{R})} = \hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}$. Consequently, for any source feature $j \in \mathcal{F}^{(B)}$ and any ground cost $c : \mathcal{R} \times \mathcal{R} \rightarrow \mathbb{R}_{\geq 0}$, $W_c(\tilde{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}) = W_c(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})})$.*

This proposition guarantees our representation is invariant to global activation scales, which naturally vary across layers due to differing hidden-state norms. Because our construction relies entirely on the relative ranking and proportional distribution of activations, it is unaffected by these arbitrary scalar shifts. Consequently, the method compares where a feature activates and how its activation mass is distributed across those contexts, rather than comparing raw activation magnitudes.

Let ρ be a base metric on $\mathcal{R}$. For $\mu, \tilde{\mu}, \nu, \tilde{\nu} \in \mathcal{P}(\mathcal{R})$, define $S_\mu = \text{supp}(\mu) \cup \text{supp}(\tilde{\mu})$ and $S_\nu = \text{supp}(\nu) \cup \text{supp}(\tilde{\nu})$. We say that $c : \mathcal{R} \times \mathcal{R} \rightarrow \mathbb{R}_{\geq 0}$ is coordinate-wise Lipschitz on the relevant support if there exist constants $L_x, L_y < \infty$ such that $|c(z, z') - c(\tilde{z}, \tilde{z}')| \leq L_x \rho(z, \tilde{z})$ for all $z, \tilde{z} \in S_\mu$ and $z', \tilde{z}' \in S_\nu$, and $|c(z, z') - c(z, \tilde{z}')| \leq L_y \rho(z', \tilde{z}')$ for all $z \in S_\mu$ and $z', \tilde{z}' \in S_\nu$.

Theorem 1 (Stability of Wasserstein feature scores). *Let $c : \mathcal{R} \times \mathcal{R} \rightarrow \mathbb{R}_{\geq 0}$ be coordinate-wise Lipschitz on the relevant support with constants L_x, L_y . Then $|W_c(\mu, \nu) - W_c(\tilde{\mu}, \tilde{\nu})| \leq L_x W_\rho(\mu, \tilde{\mu}) + L_y W_\rho(\nu, \tilde{\nu})$.*

Theorem 1 guarantees that the feature distance is stable under smooth distributional shifts. This robustness is critical because top- K activation sets naturally fluctuate across corpora, SAE retrains, or representation noise. The theorem ensures the Wasserstein score changes substantially only if a nontrivial probability mass moves significantly within $\mathcal{R}$, or under a highly sensitive ground cost. Thus, our method removes the need for exact token overlap, requiring only that activation distributions remain proximate in the shared reference geometry. Metric-specific constants (Euclidean, Mahalanobis, cosine, flow-aware) are detailed in Appendix E.4.

4.2 Margin-Based Recovery of Feature Matching and Circuit Compressing

Score stability alone does not guarantee that the final discrete decision is unchanged. A feature match can change if the best and second-best source features have nearly identical scores.

We therefore state margin conditions under which stable scores imply stable matching or grouping decisions. For target feature $i \in \mathcal{F}^{(A)}$ and source feature $j \in \mathcal{F}^{(B)}$, define $D_{\mathcal{R}}(i, j) = W_c(\mu_i^{(A \rightarrow \mathcal{R})}, \nu_j^{(B \rightarrow \mathcal{R})})$ and $\hat{D}_{\mathcal{R}}(i, j) = W_c(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})})$. Let $j_i^* = \arg \min_{j \in \mathcal{F}^{(B)}} D_{\mathcal{R}}(i, j)$ and $\Delta_i = \min_{j \neq j_i^*} [D_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j_i^*)]$.

Theorem 2 (Matching recovery under Wasserstein margin). *Fix $i \in \mathcal{F}^{(A)}$ and suppose j_i^* is unique with $\Delta_i > 0$. If $\sup_{j \in \mathcal{F}^{(B)}} |\hat{D}_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j)| \leq \epsilon_i$ and $\Delta_i > 2\epsilon_i$, then $\arg \min_{j \in \mathcal{F}^{(B)}} \hat{D}_{\mathcal{R}}(i, j) = j_i^*$.*

This theorem ensures the matching decision are reliable and not merely artifacts of representation noise by converting score stability into matching stability. It introduces the matching margin Δ_i , which measures how much closer the true source is compared to the nearest competing source. If empirical estimation perturbs every score by at most ϵ_i , the worst-case scenario shrinks this gap by at most $2\epsilon_i$. Consequently, whenever $\Delta_i > 2\epsilon_i$, the empirical nearest-neighbor rule is mathematically guaranteed to preserve the true match. In practice, this allows us to confidently accept large-margin matches as highly robust to finite-sample perturbations, while identifying and flagging inherently unstable matches.

For circuit compression, each semantic group can be represented by a semantic center in the shared reference space. A semantic center is a representative point of one semantic cell or supernode. Let $\xi_1, \dots, \xi_m \in \mathcal{R}$ be semantic centers, one for each supernode. For a projected distribution $\mu \in \mathcal{P}(\mathcal{R})$, define the assignment score $\mathcal{Q}_k(\mu) = \mathbb{E}_{z \sim \mu} [\rho(z, \xi_k)]$. For feature i , let $k_i^* = \arg \min_{k \in [m]} \mathcal{Q}_k(\mu_i^{(A \rightarrow \mathcal{R})})$, $\hat{k}_{i,T} = \arg \min_{k \in [m]} \mathcal{Q}_k(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})})$, and $\Gamma_i = \min_{r \neq k_i^*} [\mathcal{Q}_r(\mu_i^{(A \rightarrow \mathcal{R})}) - \mathcal{Q}_{k_i^*}(\mu_i^{(A \rightarrow \mathcal{R})})]$.

Theorem 3 (Semantic supernode recovery). *Suppose the population semantic assignment k_i^* is unique with margin $\Gamma_i > 0$. If $W_\rho(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})}) < \Gamma_i/2$, then the empirical semantic assignment recovers the population assignment: $\hat{k}_{i,T} = k_i^*$.*

To confidently compress complex feature circuits into semantic supernodes, we need guarantees that the grouping decision is stable under finite-sample noise. Theorem 3 provides such a guarantee through the Voronoi margin Γ_i , which measures the assignment-score gap between the correct semantic center and the nearest competing semantic center in $\mathcal{R}$. This margin induces a recovery radius $\Gamma_i/2$: if the empirical projected distribution satisfies $W_\rho(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})}) < \Gamma_i/2$, then the empirical semantic assignment is guaranteed to match the population assignment. Thus, large-margin features can be merged into supernodes with confidence, while small-margin features are close to semantic boundaries and should be left ungrouped or marked as uncertain. Moreover, this recovery condition requires the empirical projected distribution to be sufficiently close to its population counterpart, which naturally raises the question of how large the sample size T must be for the condition to hold with high probability. To answer this question, we refer readers to Appendix E.8 (Table 8) for high-probability recovery variants and sufficient sample complexity guarantees, and to Appendix E.7 for combined distribution-and-cost perturbation bounds.

5 Experiments

We conduct the experiments on open source JumpReLU SAEs Lieberum et al. [2024] trained on the output of MLP block of Gemma-2-2b Team [2024] and our own TopK SAEs Gao et al. [2024] trained on Residual stream of GPT2 Radford et al. [2019], the training details are in Appendix C. We implement the following baselines, implementation details are in Appendix C.2:

- SAE Match Balagansky et al. [2024] (**Match**): computes the distance between two features via L_2 of the decoder vectors. This method contains two setups: directly matching two layers, or “composition permutation” where we recursively match two consecutive layers till we reach the target layer, we denote this setup as **Match ML**.
- Feature Flow Laptev et al. [2025] (**FeatFlow**): uses cosine similarity of the decoder vectors as the feature distance. This method contains also contains two setups of direct matching and “composition permutation”, we denote this setup as **FeatFlow ML**.

- Attribution Patching Syed et al. [2023], Marks et al. [2024] (**Attrib**): selects important feature for a target feature via using attribution patching.
- LLM selection (**LLM Sel**): we prompt LLM (gpt-oss-120b OpenAI [2025]) to select which feature is the most relevant to a target feature. Note that prompting LLM to select among all pairs of features are very costly, thus we do not evaluate this method on feature matching (Section 5.1).

Lastly, Appendix D illustrates that representing feature as distribution outperforms representing as one vector for all of main experiments, showing that activation distribution preserves more semantic information.

5.1 Feature Matching Experiments

We test the feature matching under three setups, matching SAEs from: consecutive layers, middle to last layer, and first to last layer detailed in Table 3. Each setup we randomly sample 100 feature pairs to evaluate, these pairs are shared across all methods to ensure fairness. Since LLM rating fluctuates, we evaluate 5 times and report the 90% confident interval. Other details about LLM prompt and implementation are in Appendix C.3. Given the target SAE at layer A and source SAE at layer B , we follow Balagansky et al. [2024] and evaluate using:

- LLM evaluation and Accuracy (**LLM Eval** and **Acc**): We use a language model to caption two features separately given their activation tokens. Given the captions, for the LLM Eval metric, we ask LLM if the two features are “similar”, “maybe”, or “different”; and we assign 3, 2 or 1 point for the judgment respectively, the higher the point the better. For the Acc metric, we report the percentage of accurate matching (LLM judge the feature pair as “similar”). We choose gpt-oss-120b OpenAI [2025] for this experiment.
- Cross Entropy loss and Variance Explained (**CE** and **VE**): If we correctly match similar feature, we can replace the target feature in layer A with source feature in B . We denote the reconstruction of SAE A given the feature replacement as $\hat{x}_{M_{A \rightarrow B}}^{(A)}$. The CE measures Cross Entropy loss of the language model when replacing the original hidden state $x^{(A)}$ with $\hat{x}_{M_{A \rightarrow B}}^{(A)}$, while the VE measures how well $\hat{x}_{M_{A \rightarrow B}}^{(A)}$ captures $x^{(A)}$ via variance explained $Var(x^{(A)} - \hat{x}_{M_{A \rightarrow B}}^{(A)})/Var(x^{(A)})$.

The results are shown in Table 1. Our method performs well on LLM Eval and Matching Accuracy, while maintaining competitive performance on downstream cross entropy loss and variance explained. Most noticeably, our method has substantially more similar matched features across all setups compared to other baselines, suggesting the effective of using reference space. Moreover, consistently reaching strong results on CE and VE shows that our method identifies the correct feature pairs with similar functional role, not just mere similar in activated tokens. On the other hand, other methods are less consistent in CE and VE metrics as well as are less accurate in multi-layer matching.

5.2 Circuit Compressing Experiments

To evaluate, we first construct feature circuits following Marks et al. [2024] with 50 nodes and compress down to 5 supernodes. Then, for each supernode, we randomly sample half of the nodes to prompt an LLM to caption the shared meaning and the other half are mixed with random features from the remaining supernodes to create a test set. We then ask the LLM to pick the node in the test set that is from the original supernode given the caption. The more accurate prediction, the more intuitive the compression. We evaluate over 200 circuits and report 90% confident interval. Further implementation details are in Appendix C.4.

The results are in table 2. We find that the our method compresses circuit more accurately for both GPT2 and Gemma-2-2B than other methods. While compressing using LLM selection is competitive, it is more costly in large circuits and still less accurate than our method. This shows that our method are the best choice for automated circuit compressing.

5.3 Matching Feature With Subtle Meaning

We test the methods ability to recognize subtle meanings in the features, where we know the “ground truth” meanings. Specifically, we use “two number addition” feature Lindsey et al. [2025] in the

Table 1: Feature matching experiment results on GPT2 and Gemma-2-2B.

Method	GPT2				Gemma-2-2B			
	LLM Eval $\uparrow$	Acc $\uparrow$	CE $\downarrow$	VE $\uparrow$	LLM Eval $\uparrow$	Acc $\uparrow$	CE $\downarrow$	VE $\uparrow$
Layer 5 match with Layer 6					Layer 12 match with Layer 13			
Our	2.53 $\pm$ 0.04	68.2 $\pm$ 1.9	3.775	0.9866	2.32 $\pm$ 0.04	58.6 $\pm$ 2.0	2.74784	0.7196
FeatFlow	<u>2.46 $\pm$ 0.04</u>	<u>66.2 $\pm$ 1.8</u>	3.774	<u>0.9866</u>	1.51 $\pm$ 0.03	20.4 $\pm$ 1.5	2.74774	0.7181
Match	2.25 $\pm$ 0.04	55.6 $\pm$ 2.6	3.776	0.9867	1.41 $\pm$ 0.01	17.2 $\pm$ 1.5	2.74781	0.7188
Attrib	1.21 $\pm$ 0.01	8.6 $\pm$ 0.7	3.785	0.9863	<u>1.65 $\pm$ 0.04</u>	<u>27.2 $\pm$ 1.8</u>	2.74804	0.7105
Layer 5 match with Layer 11					Layer 12 match with Layer 25			
Our	1.56 $\pm$ 0.04	23.2 $\pm$ 2.2	<u>3.806</u>	<u>0.9127</u>	1.83 $\pm$ 0.04	33.6 $\pm$ 2.9	2.74809	0.9115
FeatFlow	<u>1.49 $\pm$ 0.02</u>	20.0 $\pm$ 1.7	3.810	0.9126	1.26 $\pm$ 0.01	10.0 $\pm$ 1.0	2.74813	<u>0.9114</u>
Match	<u>1.49 $\pm$ 0.03</u>	<u>22.2 $\pm$ 1.9</u>	3.800	0.9130	1.18 $\pm$ 0.03	6.6 $\pm$ 1.1	<u>2.74811</u>	<u>0.9114</u>
FeatFlow ML	1.18 $\pm$ 0.01	7.8 $\pm$ 1.1	3.808	0.9127	1.23 $\pm$ 0.04	9.4 $\pm$ 2.3	<u>2.74811</u>	<u>0.9114</u>
Match ML	1.17 $\pm$ 0.04	6.8 $\pm$ 1.9	3.808	<u>0.9127</u>	1.26 $\pm$ 0.04	11.2 $\pm$ 1.2	2.74812	<u>0.9114</u>
Attrib	1.26 $\pm$ 0.02	10.6 $\pm$ 1.2	3.955	0.8911	<u>1.34 $\pm$ 0.04</u>	<u>13.0 $\pm$ 1.3</u>	2.74820	0.9109
Layer 0 match with Layer 11					Layer 0 match with Layer 25			
Our	1.39 $\pm$ 0.03	17.2 $\pm$ 2.3	3.828	0.9111	1.83 $\pm$ 0.05	34.8 $\pm$ 2.1	<u>2.74866</u>	0.9094
FeatFlow	<u>1.34 $\pm$ 0.02</u>	<u>12.4 $\pm$ 1.0</u>	3.855	0.9095	1.23 $\pm$ 0.04	8.4 $\pm$ 1.5	2.74884	0.9091
Match	1.27 $\pm$ 0.02	10.0 $\pm$ 1.3	3.833	<u>0.9110</u>	1.21 $\pm$ 0.01	8.0 $\pm$ 0.8	2.74874	0.9094
FeatFlow ML	1.16 $\pm$ 0.04	5.4 $\pm$ 2.4	<u>3.831</u>	<u>0.9110</u>	1.11 $\pm$ 0.03	4.4 $\pm$ 1.2	2.74889	0.9085
Match ML	1.14 $\pm$ 0.03	6.0 $\pm$ 2.1	3.911	0.9063	1.13 $\pm$ 0.04	5.2 $\pm$ 1.8	2.74863	0.9094
Attrib	1.18 $\pm$ 0.01	6.2 $\pm$ 1.3	3.837	0.9105	<u>1.28 $\pm$ 0.05</u>	<u>11.6 $\pm$ 2.6</u>	2.74885	<u>0.9092</u>

Table 2: Circuit compressing experiment results on GPT2 and Gemma-2-2B.

Method	Gemma-2-2B	GPT2-small
Our	0.6151 $\pm$ 0.0153	0.6812 $\pm$ 0.0173
FeatFlow	0.5349 $\pm$ 0.0137	<u>0.6655 $\pm$ 0.0167</u>
Match	0.5563 $\pm$ 0.0200	0.5555 $\pm$ 0.0183
LLM Sel	<u>0.6019 $\pm$ 0.0131</u>	0.6572 $\pm$ 0.0171
Attrib	0.5125 $\pm$ 0.0129	0.5405 $\pm$ 0.0189

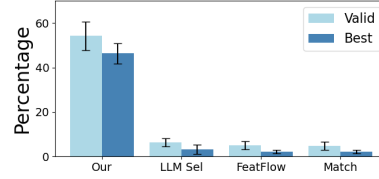


Figure 2: Digit addition experiment results. “Valid” means the user agree with the matching, and “Best” means the method has the best match among all methods.

format “ $p_1 + p_2 =$ ”. We limit $p_1, p_2 \in [0, 9]$ for a less costly experiment to run LLM Selection method. We extract the features’ activation on the “=” token over 100 datapoints ($[0, 9] \times [0, 9]$) and plot the activation pattern of each feature. The patterns tell the ground truth of feature meaning, i.e. feature: “first number is 5” would have activation pattern of “only row $p_1 = 5$ has positive values”, one such pattern is in Figure 3. Based on the patterns as the ground truth, we can identify if the matching correctly identify the functional meaning or merely “matching in math” context only.

We create blind ranking API for the methods and ask volunteers to rank which candidate feature is better given a target feature, randomly ranking 100 target features. We test on matching feature of the same layer (Layer 19 MLP from Gemma-2-2b) to ensure other methods can capture on the same manifold. Details are in Appendix C.5.

The quantitative results in Figure 2 and qualitative in Figure 3. Our methods are ranked substantially higher than other matching methods by the people in the survey. The qualitative figure 3 shows that our method correctly capture the hidden meaning of the features while others cannot.

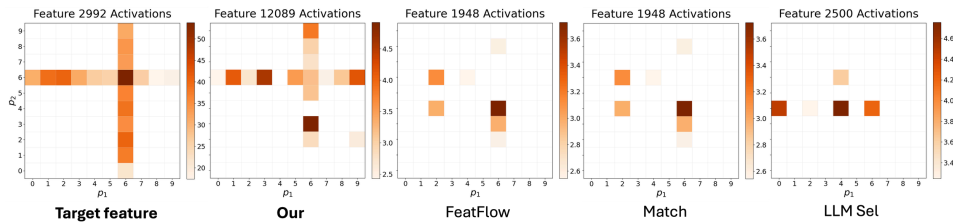


Figure 3: Qualitative example of “digit addition” (Section 5.3). Our method correctly identifies the similar feature while other methods fail.

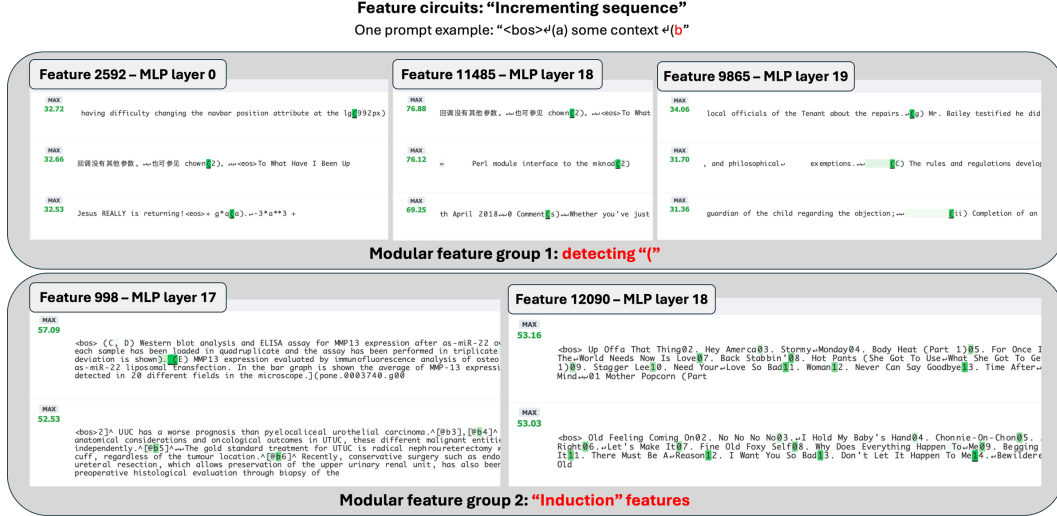


Figure 4: Two modular circuits identified from “incrementing sequences” circuits Marks et al. [2024]. We find two distinct modular function among the features, namely detecting “(” features and “Induction” features Olsson et al. [2022]. For example, feature 12090 layer 18 sees “02” of previous text (without activate on “02”), it recognizes this is increment sequence and activates on “03”, etc.

6 Applications

We demonstrate two applications of our framework. We show qualitative results of automatic circuit compression in Section 6.1. Furthermore, we can identify feature groups that consistently perform similar function role in many circuits, Section 6.2.

6.1 Case Study: Automated Circuit Compression

We demonstrate qualitative results in *automatically* compressing feature circuits to reveal the mechanisms of the model. We follow the circuit examples from Neuropedia Lin [2023] and Lindsey et al. [2025] in Figure 5, 6, 7. Specifically, we found that our compression of “Dallas circuit” (Figure 5) found similar supernodes as in open source *manual* implementation Lin [2023]. We find that the model retrieve information about Dallas as the Texas state, it then retrieve various government / U.S court related contexts and output the correct answer “Austin”. We identify compressed circuit that *do not have open source compression* in “rhyming circuit” (Figure 6), we find that the model first recognizes the poem context and thinks about a noun that rhythm and related to “hunger”. It then thinks about rhythm in “animal” context, ultimately outputting “rabbit”. Details are in Appendix C.6.

6.2 Case Study: Finding Modular Feature Group

Beyond unsupervisedly finding circuits that perform similar task Marks et al. [2024], we can further automatically detect feature group that the features consistently perform the same functional role and appear in similar circuits. Specifically, we hypothesize that features that are frequently assigned into the same supernodes have the same functional role. To check this hypothesis, we propose the following procedure: we use method in Marks et al. [2024] to identify circuits that perform similar task. We then perform circuit compression on all circuits. Lastly, we apply clustering based on how frequent the features are in the same supernodes throughout all compressions, creating groups of features that we term as “modular feature groups”. Details on the procedure are in Appendix C.7.

Figure 4, 8, 9 show the modular feature groups. For the interesting behavior of “incrementing sequences” in Marks et al. [2024], we can automatically identify two interesting groups (Figure 4). Group 1 help recognize token “(” and group 2 contain “induction features” - the core mechanism in recognizing repeated context Olsson et al. [2022]. This shows that our circuit compression correctly identifies the subtle functional role of the features, demonstrating the capability of unsupervisedly exploring complex behavior. Similar examples can be seen in Figure 8 and 9.

7 Conclusion

In this work, we introduced a distributional framework for comparing Sparse Autoencoder features across multi-layer language models. We framed cross-manifold feature comparison as an Optimal Transport problem. Our approach achieves strong multi-layer feature matching and enables the automated feature circuits compression into interpretable semantic supernodes.

References

- Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*, 2023.
- Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093*, 2024.
- Nikita Balagansky, Ian Maksimov, and Daniil Gavrilov. Mechanistic permutability: Match features across layers. *arXiv preprint arXiv:2410.07656*, 2024.
- Daniil Laptev, Nikita Balagansky, Yaroslav Aksenov, and Daniil Gavrilov. Analyze feature flow to enhance interpretation and steering in language models. *arXiv preprint arXiv:2502.03032*, 2025.
- Samuel Marks, Can Rager, Eric J Michaud, Yonatan Belinkov, David Bau, and Aaron Mueller. Sparse feature circuits: Discovering and editing interpretable causal graphs in language models. *arXiv preprint arXiv:2403.19647*, 2024.
- Jacob Dunefsky, Philippe Chlenski, and Neel Nanda. Transcoders find interpretable llm feature circuits. *Advances in Neural Information Processing Systems*, 37:24375–24410, 2024.
- Jack Lindsey, Wes Gurnee, Emmanuel Ameisen, Brian Chen, Adam Pearce, Nicholas L. Turner, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, Jonathan Marcus, Michael Sklar, Adly Templeton, Trenton Bricken, Callum McDougall, Hoagy Cunningham, Thomas Henighan, Adam Jermy, Andy Jones, Andrew Persic, Zhenyi Qi, T. Ben Thompson, Sam Zimmerman, Kelley Rivoire, Thomas Conerly, Chris Olah, and Joshua Batson. On the biology of a large language model. *Transformer Circuits Thread*, 2025. URL <https://transformer-circuits.pub/2025/attribution-graphs/biology.html>.
- Johnny Lin. Neuronpedia: Interactive reference and tooling for analyzing neural networks, 2023. URL <https://www.neuronpedia.org>. Software available from neuronpedia.org.
- Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermy, 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. <https://transformer-circuits.pub/2023/monosemantic-features/index.html>.
- Florian Beier, Moritz Piening, Robert Beinert, and Gabriele Steidl. Joint metric space embedding by unbalanced optimal transport with gromov—wasserstein marginal penalization. In *Proceedings of ICML*, volume 25, 2025.
- Moritz Imfeld, Jacopo Galdi, Marco Giordano, Thomas Hofmann, Sotiris Anagnostidis, and Sidak Pal Singh. Transformer fusion with optimal transport. In *The Twelfth International Conference on Learning Representations*, 2023.
- Igor Melnyk, Youssef Mroueh, Brian Belgodere, Mattia Rigotti, Apoorva Nitsure, Mikhail Yurochkin, Kristjan Greenewald, Jiri Navratil, and Jarret Ross. Distributional preference alignment of llms via optimal transport. *Advances in Neural Information Processing Systems*, 37:104412–104442, 2024.
- Ikhyun Cho and Julia Hockenmaier. Toward efficient sparse autoencoder-guided steering for improved in-context learning in large language models. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 28961–28973, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1474. URL <https://aclanthology.org/2025.emnlp-main.1474/>.
- Tom Lieberum, Senthoooran 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>.

- Gemma Team. Gemma. 2024. doi: 10.34740/KAGGLE/M/3301. URL <https://www.kaggle.com/m/3301>.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- Aaquib Syed, Can Rager, and Arthur Conmy. Attribution patching outperforms automated circuit discovery, 2023. URL <https://arxiv.org/abs/2310.10348>.
- OpenAI. gpt-oss-120b & gpt-oss-20b model card, 2025. URL <https://arxiv.org/abs/2508.10925>.
- Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. <https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html>.
- Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders. *arXiv preprint arXiv:2407.14435*, 2024.
- Bart Bussmann, Patrick Leask, and Neel Nanda. Batchtopk sparse autoencoders. *arXiv preprint arXiv:2412.06410*, 2024.
- Julian Minder, Clément Dumas, Caden Juang, Bilal Chughtai, and Neel Nanda. Robustly identifying concepts introduced during chat fine-tuning using crosscoders. *arXiv preprint arXiv:2504.02922*, 2025.
- Javier Ferrando, Oscar Obeso, Senthooran Rajamanoharan, and Neel Nanda. Do i know this entity? knowledge awareness and hallucinations in language models, 2025. URL <https://arxiv.org/abs/2411.14257>.
- Andrey Galichin, Alexey Dontsov, Polina Druzhinina, Anton Razzhigaev, Oleg Y. Rogov, Elena Tutubalina, and Ivan Oseledets. I have covered all the bases here: Interpreting reasoning features in large language models via sparse autoencoders, 2025. URL <https://arxiv.org/abs/2503.18878>.
- Peter Halmos, Julian Gold, Xinhao Liu, and Benjamin Raphael. Hierarchical refinement: Optimal transport to infinity and beyond. In *Forty-second International Conference on Machine Learning*, 2025.
- Jayadev Naram, Fredrik Hellström, Ziming Wang, Rebecka Jörnsten, and Giuseppe Durisi. Theoretical performance guarantees for partial domain adaptation via partial optimal transport. In *International Conference on Machine Learning*, pages 45663–45681. PMLR, 2025.
- Yanbei Liu, Chongxu Wang, Zhitao Xiao, Lei Geng, Yanwei Pang, and Xiao Wang. Hgot: Self-supervised heterogeneous graph neural network with optimal transport. In *International Conference on Machine Learning*, pages 39742–39754. PMLR, 2025.
- Jiaming Lv, Haoyuan Yang, and Peihua Li. Wasserstein distance rivals kullback-leibler divergence for knowledge distillation. *Advances in Neural Information Processing Systems*, 37:65445–65475, 2024.
- Khai Nguyen, Hai Nguyen, Tuan Pham, and Nhat Ho. Lightspeed geometric dataset distance via sliced optimal transport. In *International Conference on Machine Learning*, pages 46162–46177. PMLR, 2025.
- Clément Bonet, Christophe Vauthier, and Anna Korba. Flowing datasets with wasserstein over wasserstein gradient flows. In *International Conference on Machine Learning (ICML)*, 2025.
- Gathika Ratnayaka, James Nichols, and Qing Wang. Learning partial graph matching via optimal partial transport. In *The Thirteenth International Conference on Learning Representations*, 2024.
- Tung Le, Khai Nguyen, Shanlin Sun, Nhat Ho, and Xiaohui Xie. Integrating efficient optimal transport and functional maps for unsupervised shape correspondence learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23188–23198, 2024.
- Ryo Kishino, Hiroaki Yamagiwa, Ryo Nagata, Sho Yokoi, and Hidetoshi Shimodaira. Quantifying lexical semantic shift via unbalanced optimal transport. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15913–15933, 2025.
- Minh-Phuc Truong, Hai An Vu, Tu Vu, Nguyen Thi Ngoc Diep, Linh Ngo Van, Thien Huu Nguyen, and Trung Le. Emo: Embedding model distillation via intra-model relation and optimal transport alignments. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 7605–7617, 2025.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

Guojun Gan, Chaoqun Ma, and Jianhong Wu. *Data Clustering: Theory, Algorithms, and Applications*. Society for Industrial and Applied Mathematics, 2007. doi: 10.1137/1.9780898718348. URL <https://epubs.siam.org/doi/abs/10.1137/1.9780898718348>.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Meta AI. Llama. <https://www.llama.com/>, 2026. Accessed: May 5, 2026.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive” or “we were unable to find the license for the dataset we used”). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading “NeurIPS Paper Checklist”,**
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers.**

1. Claims

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

Answer: [Yes]

Justification: See Abstract and Introduction (Section 1).

Guidelines:

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

2. Limitations

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

Answer: [Yes]

Justification: See Appendix A.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate “Limitations” section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren’t acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

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

Answer: [Yes]

Justification: See Appendix E.

Guidelines:

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

4. Experimental result reproducibility

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

Answer: [Yes]

Justification: See Appendix C.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.

- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

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

Answer: [No]

Justification: We will provide the code once the paper get published.

Guidelines:

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

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

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

Answer: [\[Yes\]](#)

Justification: See Appendix C.

Guidelines:

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

7. Experiment statistical significance

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

Answer: [\[Yes\]](#)

Justification: We report 90% confident interval, see Section 5.

Guidelines:

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

8. Experiments compute resources

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

Answer: [\[Yes\]](#)

Justification: See Appendix C.8.

Guidelines:

- The answer [\[N/A\]](#) means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

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

Answer: [Yes]

Justification: We follow code of ethics guidelines and only use open source data and models.

Guidelines:

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

10. Broader impacts

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

Answer: [N/A]

Justification: Our method do not tie to any deployments.

Guidelines:

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

11. Safeguards

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

Answer: [N/A]

Justification: We do not release new data or model.

Guidelines:

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

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

12. Licenses for existing assets

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

Answer: [\[Yes\]](#)

Justification: We use open source dataset and models that are sufficiently cited.

Guidelines:

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

13. New assets

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

Answer: [\[N/A\]](#)

Justification: We do not release new assets.

Guidelines:

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

14. Crowdsourcing and research with human subjects

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

Answer: [\[Yes\]](#)

Justification: See Appendix C.5.

Guidelines:

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

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

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

Answer: [Yes]

Justification: Our survey poses no risk for study participants.

Guidelines:

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

16. Declaration of LLM usage

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

Answer: [N/A]

Justification: We do not use LLM for core methodology.

Guidelines:

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

Supplementary for Semantic Optimal Transport for Sparse Autoencoder Feature Matching and Circuit Compression

This supplementary material serves as the appendix to the main paper to provide full detailed pseudocode of algorithms, derivations and proofs of key theoretical results, comprehensive ablation studies and hyperparameter settings, as well as expanded discussions on implementation to reproduce results, scalability, and generalization. It is organized as follows:

A. Limitations	21
B. Related Works	23
C. Experiment details	24
C.1. Algorithm for Feature Matching and Circuit Compression	24
C.2. Baseline Implementations	25
C.3. Feature Matching Setups	25
C.4. Circuit Compression Setups	26
C.5. Digit Addition Experiment Setups	26
C.6. Automated Circuit Compression Setups	26
C.7. Finding Modular Feature Group Setups	27
C.8. SAE Training	28
C.9. LLM Prompt	28
D. Ablation Study: Why do we need feature as distribution?	29
E. Additional Theoretical Properties	30
E.1. Activation Rescaling Invariance of Projected Feature Distributions	30
E.2. Flow Cost as a Refinement of Positional Cost	31
E.3. Stability of Wasserstein Feature Distance	32
E.4. Metric-Specific Stability Corollaries	34
E.5. Stability Under Ground-Cost Perturbation	41
E.7. Matching recovery under Wasserstein margin	44
E.8. Voronoi Semantic Cell Recovery	47
E.9. Synthetic Verification of Voronoi Semantic-Cell Recovery	51

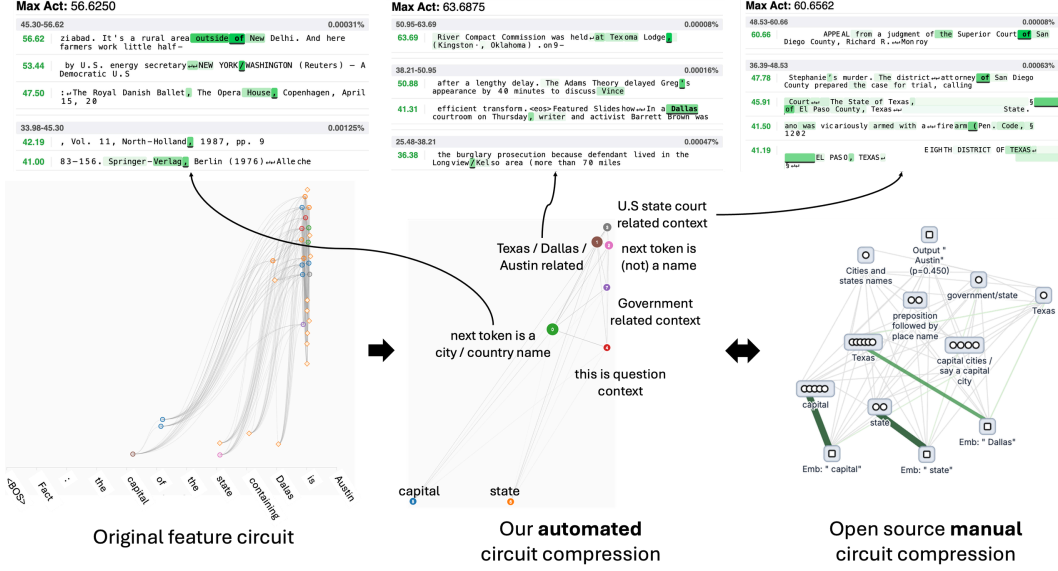


Figure 5: This figures compares compares our automated circuit compression and open source manual compression Lin [2023] on “Dallas” circuit. Our method correctly finds supernodes as the open source.

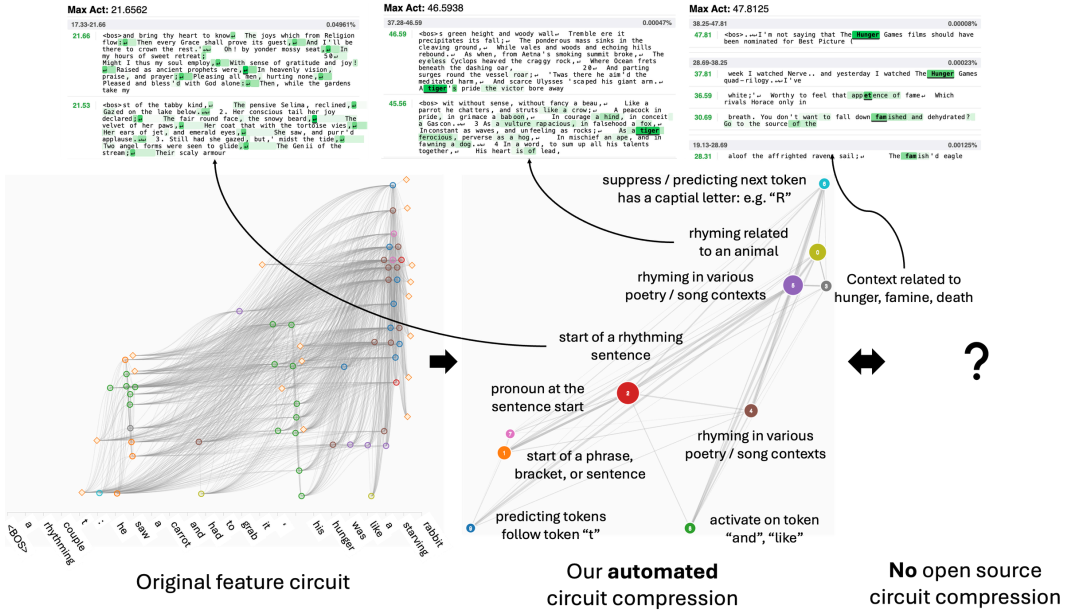


Figure 6: This figures compares compares our automated circuit compression and open source manual compression Lin [2023] on “Rhythm” circuit. Our method can scale up while open source implementation cannot.

8 Appendix

A Limitations

A primary limitation of our current framework lies in the choice of the ground cost $c(\cdot, \cdot)$ used in our Optimal Transport formulation. Currently, we compute the Wasserstein distance using the standard L_2 distance in the shared reference space. This implicitly assumes a flat, Euclidean geometry, which may not fully capture the complex, non-linear manifold structure of language model representations.

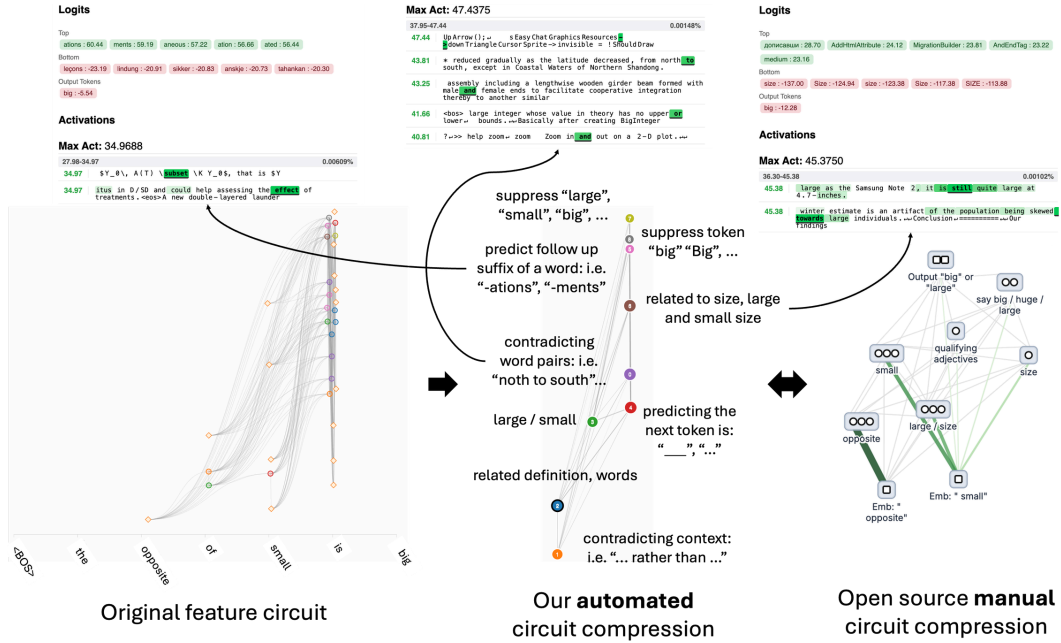


Figure 7: This figures compares compares our automated circuit compression and open source manual compression Lin [2023] on “Opposite” circuit. Our method finds supernodes that are not in open source implementation such as “predict next token is ’_’ or ’...’”.

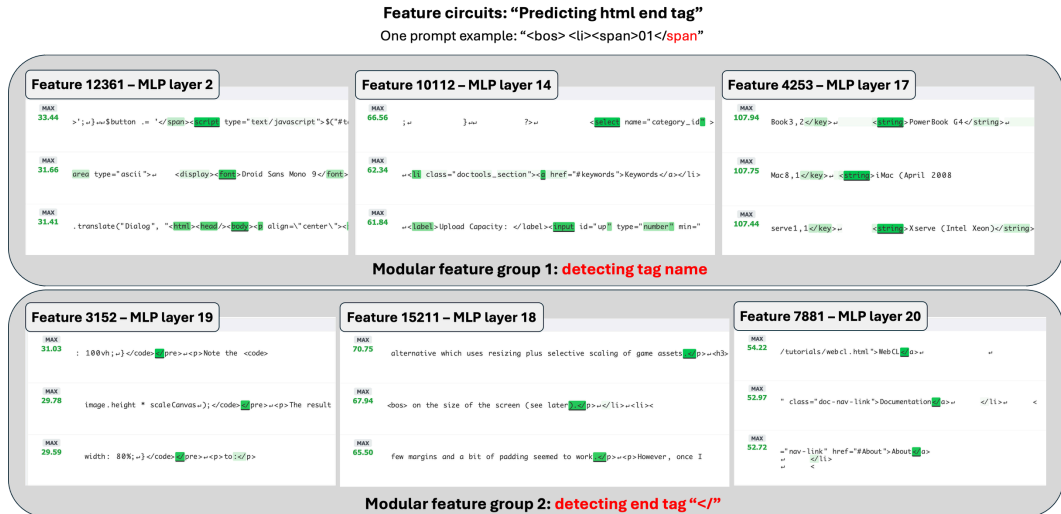


Figure 8: Two modular circuits identified form “predicting html end tags” circuits. We find two distinct modular function among the features, namely detecting tag names and detecting end tags.

Feature circuits: “Predicting token follow up an apostrophe”

One prompt example: “<bos> and who was twice her husband’s”

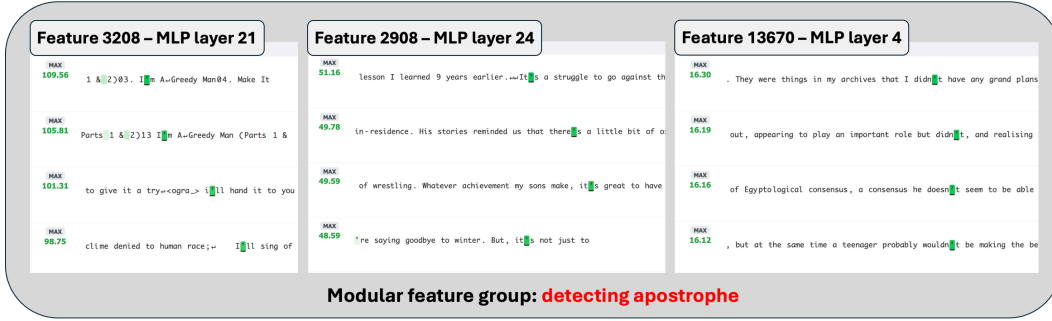


Figure 9: Two modular circuits identified form “predict token follow up an apostrophe” circuits. We find a group of features consistently detecting apostrophes.

A promising direction for future work is to replace the L_2 metric with a more general, geometry-aware ground cost. For instance, leveraging the velocity field derived from flow matching models could define a principled, geodesic-based transport cost, allowing the feature distributions to be aligned along the true continuous dynamics of the latent space.

B Related Works

Sparse Autoencoder: SAE is proposed to extract interpretable features from a latent space of one layer of a deep model Rajamanoharan et al. [2024], Gao et al. [2024], Bussmann et al. [2024], Cunningham et al. [2023]. It shows to be effective in many applications, including model steering Cho and Hockenmaier [2025], model diffing Minder et al. [2025], study hallucination Ferrando et al. [2025], study reasoning Galichin et al. [2025] etc.

Feature Matching: To understand how features evolve across layers and decision making of a model, recent works come up with feature matching Balagansky et al. [2024], Laptev et al. [2025]. The feature matching works aim to find a mapping between SAE features at one layer and features of a SAE at another layer so that the mapped features have similar meaning with the target feature. Balagansky et al. [2024] formulate the mapping as a 1-to-1 linear assignment problem that minimizes the mean squared error between target feature’s and mapped feature’s decoder vectors. Laptev et al. [2025] further propose mapping with a feature that induces the highest cosine similarity between decoder vectors. While these methods can be effective in matching feature from consecutive layers where the decoder vectors are trained on more similar latent space distributions, in matching across multi-layer, their methods are less accurate Balagansky et al. [2024] due to the distribution difference.

Feature Circuit and Circuit Compression: Feature Circuit Marks et al. [2024], Dunefsky et al. [2024], Lindsey et al. [2025] is a causal graph with nodes as features and edges show causal relation with a sink feature node, thereby revealing how features interact that affect the decision of a model. Marks et al. [2024] construct the circuit by first filter out important features to model prediction via attribution patching Syed et al. [2023]. For each filtered features, they find the edge weights by approximate the causal relations between the remaining features with attribution patching, only the strong connections are retained. Dunefsky et al. [2024] propose Transcoder, a variant of SAE, to approximate the language model as a linear interaction between features that is input independent, which aids the understanding feature interaction generalized across any input. Based on their work, Lindsey et al. [2025] identify various interesting circuits and feature interactions. The main downside of circuits is the complexity of manually interpreting hundreds of features and thousands of connections. Researchers have to group circuits’ nodes that share the same meaning into supernodes, helping to reveal high level mechanisms of the model that are often drown by the circuit complexity. However, all of this effort is done manually Lindsey et al. [2025], Lin [2023], hence, unscalable to large circuits.

Optimal Transport (OT): OT provides a geometry for comparing probability distributions by measuring the minimum cost of moving mass between them Halmos et al. [2025]. It has been widely used in machine learning when the object of interest is more naturally represented as a distribution

than as a single vector, including domain adaptation Naram et al. [2025], representation learning Liu et al. [2025], Lv et al. [2024], dataset comparison Nguyen et al. [2025], Bonet et al. [2025], graph matching Ratnayaka et al. [2024], shape correspondence Le et al. [2024], and NLP similarity Kishino et al. [2025], Truong et al. [2025]. These works show that OT is useful when the geometry of the support matters and pointwise vector comparison fails to capture important distributional structure.

Most OT-based alignment methods compare distributions over external objects, such as samples from different domains, graphs, shapes, documents, or embedding spaces. In our setting, the object being compared is an internal SAE feature of language model. Rather than treating a feature as a decoder vector, we build a distribution from the contexts on which the feature activates and the relative strength of those activations. These activation-induced distributions are then projected into a shared reference space, allowing features from different layers to be compared in a common geometry. This makes Wasserstein distance a semantic distance between feature activation patterns, and supports both cross-layer feature matching and automatic grouping of circuit nodes into semantic supernodes.

C Experiment details

In this section we detail feature matching, circuit compression, the baseline implementations, our experiment setups, SAE training, and LLM prompts.

C.1 Algorithm for Feature Matching and Circuit Compression

Feature Matching: In algorithm 1, we compute the distance for feature matching by first project the distributions of both target and source features set on the reference space. We then compute the centroids of the distribution on the space. The top-50 source features that are the closest to the target feature are retained, we discard the rest. This step is crucial to speed up the algorithm by not having to compute most of faraway source feature distributions. We then compute the top-1 feature using Wasserstein distance with Sinkhorn algorithm Cuturi [2013].

Algorithm 1 Feature Matching Algorithm.

Input: Target feature $i \in \mathcal{F}^{(A)}$, source features $\mathcal{F}^{(B)}$, and empirical distributions $\hat{\mu}_{i,T}^{(A)}$, $i \in \mathcal{F}^{(A)}$ and $\hat{\mu}_{j,T}^{(B)}$, $\forall j \in \mathcal{F}^{(B)}$.

- 1: Compute reference space projected distributions $\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}$, $i \in \mathcal{F}^{(A)}$ and $\hat{\mu}_{j,T}^{(B \rightarrow \mathcal{R})}$, $\forall j \in \mathcal{F}^{(B)}$.
- 2: Compute feature distributions' centroids $\bar{\mu}_{i,T}^{(A \rightarrow \mathcal{R})} = \sum_{t \in \mathcal{I}_{i,T}^{(A),K}} \hat{w}_{i,t}^{(A)} z_t^{(A)}$, $i \in \mathcal{F}^{(A)}$ and $\bar{\mu}_{j,T}^{(B \rightarrow \mathcal{R})} = \sum_{t \in \mathcal{I}_{j,T}^{(B),K}} \hat{w}_{j,t}^{(B)} z_t^{(B)}$, $\forall j \in \mathcal{F}^{(B)}$.
- 3: Extract top-50 source features that have the closest centroids to target feature centroid $\text{Top50}_{j \in \mathcal{F}^{(B)}} \left(-c \left(\bar{\mu}_{j,T}^{(B \rightarrow \mathcal{R})}, \bar{\mu}_{i,T}^{(A \rightarrow \mathcal{R})} \right) \right)$ $\triangleright$ We do this to filter out faraway distributions, speeding up the algorithm. The choice of top-50 is arbitrary.
- 4: Compute the Wasserstein distance between target feature projected distribution $\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}$, $i \in \mathcal{F}^{(A)}$ and source features projected distributions within the top-50 $\hat{\mu}_{p,T}^{(B \rightarrow \mathcal{R})}$, $\forall p \in \text{Top50}_{j \in \mathcal{F}^{(B)}} \left(-c \left(\bar{\mu}_{j,T}^{(B \rightarrow \mathcal{R})}, \bar{\mu}_{i,T}^{(A \rightarrow \mathcal{R})} \right) \right)$ via Sinkhorn algorithm Cuturi [2013].
- 5: $\widehat{M}_{A \rightarrow B}(i)$ is the closest source feature.

Return: $\widehat{M}_{A \rightarrow B}(i)$.

Circuit Compression: We detail our circuit compression as follows: in algorithm 2, we first project the distribution onto reference space $\mathcal{R}$. We then compute the Wasserstein distance between any node pairs in the circuit using Sinkhorn Cuturi [2013]. Lastly, we apply Agglomerative clustering Gan et al. [2007] to find the supernodes.

Algorithm 2 Circuit Compression Algorithm.

Input: Circuit $G = (V, E)$, number of supernodes $m \leq |V|$, and empirical distributions $\hat{\mu}_{i,T}^{(\ell)}, (i, \ell) \in V$.

- 1: Compute reference space projected distributions $\hat{\mu}_{i,T}^{(\ell \rightarrow \mathcal{R})}, (i, \ell) \in V$
- 2: Compute Wasserstein distance via Sinkhorn Cuturi [2013] between any two pairs $u, v \in V$ resulting in $\hat{D}_{\mathcal{R}}(u, v), \forall u, v \in V, u \neq v$.
- 3: Apply Agglomerate clustering Gan et al. [2007] to find supernodes set: $\mathbb{C} = \{\mathcal{C}_1, \dots, \mathcal{C}_m\}$.

Return: $\mathbb{C}$.

C.2 Baseline Implementations

We compute the distance matrix $\hat{D}$ for each baseline (except for LLM Selection) as follows:

SAE Match Balagansky et al. [2024]. This method computes the feature distance via the L_2 norm of the threshold-normalized decoder vectors. We define threshold $a_{\min,i}^{(A)}$ as the minimum positive activation of feature i across a corpus of T tokens. We then compute the threshold-normalized decoder vector Balagansky et al. [2024] $\widetilde{W}_{dec}^{(A)}[:, i]$ by scaling the raw decoder direction by this minimum activation:

$$\widetilde{W}_{dec}^{(A)}[:, i] = a_{\min,i}^{(A)} W_{dec}^{(A)}[:, i]. \quad (6)$$

The entries of the distance matrix are given by:

$$\left(\hat{D}_{\text{Match}}\right)_{i,j} = \left\| \widetilde{W}_{dec}^{(A)}[:, i] - \widetilde{W}_{dec}^{(B)}[:, j] \right\|_2 \quad (7)$$

Feature Flow Laptev et al. [2025]. This method relies on the cosine similarity of the decoder vectors. Because cosine similarity is invariant to positive scalar multiplication, we can compute it using the raw decoder vectors. To convert this similarity into a valid distance metric for nearest-neighbor matching, we compute:

$$\left(\hat{D}_{\text{FeatFlow}}\right)_{i,j} = 1 - \frac{\left\langle W_{dec}^{(A)}[:, i], W_{dec}^{(B)}[:, j] \right\rangle}{\left\| W_{dec}^{(A)}[:, i] \right\|_2 \left\| W_{dec}^{(B)}[:, j] \right\|_2} \quad (8)$$

Attribution Patching Syed et al. [2023], Marks et al. [2024]. We consider method that find important features via gradient. For two feature $i \in \mathcal{F}^{(A)}$ and $j \in \mathcal{F}^{(B)}$ and $A > B$, feature i has the gradient over j . We approximate the contribution of feature $a_j^{(B)}$ to feature $a_i^{(A)}$ via gradient over a dataset of tokens. The higher the contribution, the more important feature j is to i . This method is used to find relevant features from previous layers Marks et al. [2024]. Concretely, we define the distance between the two features as:

$$\left(\hat{D}_{\text{Attrib}}\right)_{i,j} = \frac{1}{T} \sum_{t=1}^T \nabla_{a_j^{(B)}} a_i^{(A)}(x_t^{(A)}) \times a_j^{(B)}(x_t^{(B)}) \quad (9)$$

LLM Selection. Given a set of source features $\mathcal{F}^{(B)}$, we prompt the LLM to caption for each feature using LLM Captioning Prompt Appendix C.9. For each target feature $i \in \mathcal{F}^{(A)}$, we ask the LLM, from all captions of feature set $\mathcal{F}^{(B)}$, to pick the caption that most resembles the activation of feature i , using LLM Selection Prompt Appendix C.9.

C.3 Feature Matching Setups

We randomly sample 100 pairs shared across all baselines where the features have at least 30 non-zero activation over 128M tokens to avoid dead features. We evaluate three main setups: consecutive layers, middle to last layers, first to last layers, shown in Table 3.

Table 3: Layer setups in feature matching experiment.

Models	GPT2		Gemma-2-2b	
	Target layer	Source layer	Target layer	Source Layer
Consecutive	6	5	13	12
Middle to last	11	5	25	12
First to last	11	0	25	0

C.4 Circuit Compression Setups

Circuit construction and compressing: We follow Marks et al. [2024] to compute feature circuits, the node scores are approximated using attribution patching instead of integrated gradient to speed up the running process. We gather prompts by chunking the validation set of The PILE dataset Gao et al. [2020] into 1000 sequences, each with 128 tokens. We extract the token sequences that the model correctly predicts the next tokens with Cross Entropy loss less than 0.1 nats. We then randomly sample 200 prompt examples shared across all baselines and compute feature circuits with 50 nodes. For each method except for LLM Sel, we compute the distance matrix between each feature pair, then use agglomerate clustering Gan et al. [2007] to form 5 supernodes. For LLM Sel, we show all the activations of features in the circuit and ask the model to caption each feature (using LLM Captioning Prompt Appendix C.9), we then ask the LLM to cluster the features that have similar caption into 5 supernodes (using LLM Selection Prompt Appendix C.9).

Evaluation: To evaluate the compression, for each supernode, we extract two equal sets of features. The first set is used to prompt a judge LLM - we use Llama-3.3-70b-instruct Meta AI [2026], where we ask the judge LLM to caption all features of the first set and give the caption for the supernode. We then use the second set, randomly mixed with equal number of features that are not in the supernode, and ask the judge to classify which feature is in the original supernode. The better the prediction, the more intuitive the compression. In our experiment, we extract 6 random features from the supernode, splitted into two sets, and 3 random features not from the supernode. We skip supernodes that do not have at least 2 features.

C.5 Digit Addition Experiment Setups

Dataset creation: We use “two number addition” feature Lindsey et al. [2025] in the format $p_1 + p_2 =$. We limit $p_1, p_2 \in [0, 9]$ for a less costly experiment to run LLM Selection method. We extract the features’ activation on the “=” token over 100 datapoints ($[0, 9] \times [0, 9]$) and plot the activation pattern of each feature. We filter features that have the number of positive activation in the range of $[5, 50]$ to avoid overly dense or dead features.

Setup: We conduct our experiment on layer 19 MLP on Gemma-2-2b. For each target feature in the filtered set, we run the matching baselines to find the most similar source feature within the same filtered set that is not the target. Note that, since this experiment is conducted on a single layer, we cannot apply Attribution Patching Syed et al. [2023] for this setup. We then create blind ranking API for all of the methods and let 10 volunteers to rate over 100 random target features.

Blind ranking API: The API example is in Figure 10. There are two choices, “valid” means that the source feature has the same meaning with the target feature, and “best” means that the source feature is the best selection among all blind ranking matches.

C.6 Automated Circuit Compression Setups

We use three prompts shown on Neuropedia for Gemma-2-2b, namely: “<BOS>Fact: the capital of the state containing Dallas is Austin”, “<BOS>A rhyming couplet: he saw a carrot and had to grab it, his hunger was like a starving rabbit”, and “<BOS>The opposite of small is large”. Following the use of corrupt prompt Marks et al. [2024], we use corrupt prompt of “Fact: the city of the country containing Paris is” and not use corrupt prompt for the other two circuits (this means the corrupt activation of each feature is zero Marks et al. [2024]). We compute the circuit using attribution patching to find important nodes, the number of nodes and supernodes are in Table 4.

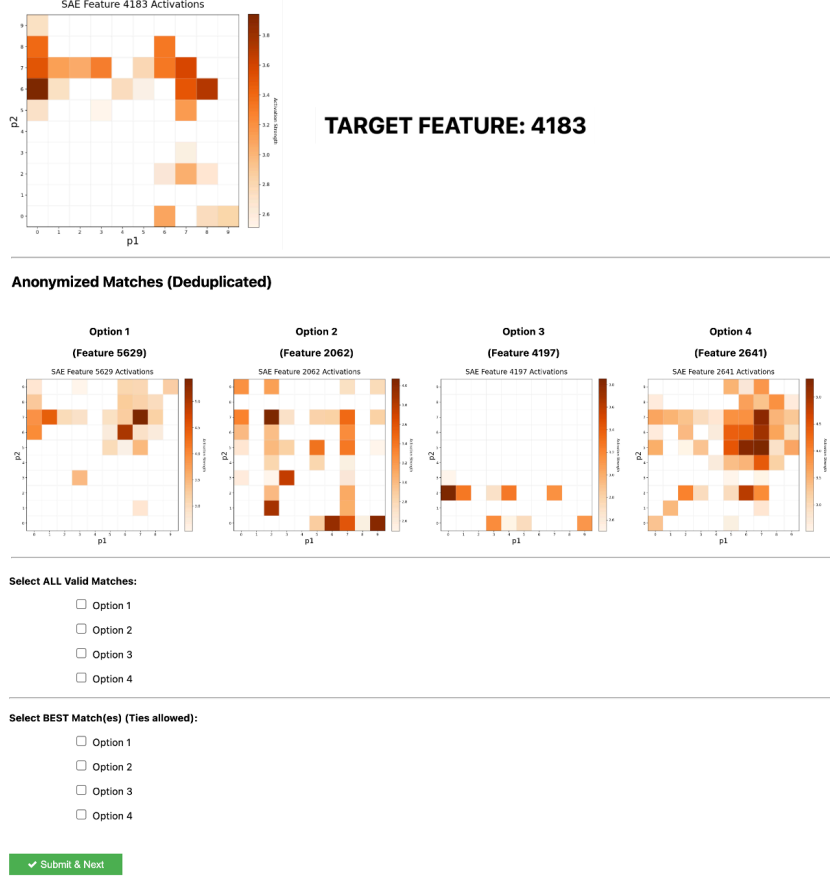


Figure 10: Blind ranking API for “Digit Addition” experiment (Section 5.3). There are two options of “Valid” and “Best” for a valid matching and best overall pick respectively. The volunteers can choose multiple choices or skip if no feature candidate matches.

Table 4: Exemplar circuit setups for Section 6.1.

Models	Gemma-2-2b	
	Nodes	Supernodes
“Dallas” circuit	50	7
“Rhythm” circuit	100	10
“Opposite” circuit	40	9

C.7 Finding Modular Feature Group Setups

Behavior clustering: The following procedure follows Marks et al. [2024]. We gather prompts by chunking the validation set of The PILE dataset Gao et al. [2020] into 1000 sequences, each with 128 tokens. We extract the token sequences that the model correctly predicts the next tokens with Cross Entropy loss less than 0.1 nats. For each prompt, we extract the last token representation (which is the token with low Cross Entropy loss) of every layers and concatenate into one vector. We then use these vectors to cluster using K-means clustering. We identify around 18000 prompt examples that are clustered into 500 groups.

Finding Modular Feature: We select a few interesting groups such as in Figure 8, 4, and 9. For each prompt example in a group, we then find a circuit (using method in Marks et al. [2024]) with attribution patching to select important nodes) with 200 nodes and compress it down to 10 supernodes. For each node pairs u, v in the circuit, we track the following Jaccard similarity: $\frac{\text{number_of_co_occurrence}}{\text{number_of_u} + \text{number_of_v} - \text{number_of_co_occurrence}}$ where $\text{number_of_co_occurrence}$ counts

when u and v are in the same supernode, and $number_of_u$ tracks the number of u appears in the circuit group. After this, we form an “edge” matrix between the all feature node pairs, and we apply Spectral clustering to find the modular group. Optionally, we can filter out “edge” between feature pairs that have low co-occurrence rate before clustering to avoid unimportant pairs.

C.8 SAE Training

We train TopK SAE Gao et al. [2024] on The Pile training set for 500M tokens on GPT2 at “pre-residual-stream” (before entering transformer blocks) of every layers. The learning rate is 10^{-4} , batch size of 2048 tokens, and auxiliary loss coefficient is 0.03125 and use Adam Optimizer. We run on A100 GPU.

C.9 LLM Prompt

LLM Compressing Prompt: ""

You are an expert AI interpretability researcher. Your task is to analyze a list of Sparse Autoencoder (SAE) features extracted from a transformer language model and group them into semantically meaningful clusters.

DATA PROVIDED:

- *Feature ID: Unique identifier for the feature.*
- *Act Context: Shows text intervals where the feature fired (activating tokens). THIS IS THE GOLD STANDARD for interpretation. Look for patterns (e.g., tracking plural nouns, specific concepts, or syntax).*
- *Logits: Shows tokens the feature promotes or suppresses. Use this to confirm the feature’s meaning.*

YOUR TASK:

- 1. Analyze the "Act Context" and "Logits" for every feature provided.*
- 2. Write a short, general 1-sentence caption explaining what each feature detects or represents.*
- 3. Group features that track similar concepts, syntactical roles, or semantic meanings into the same cluster.*
- 4. Output the result STRICTLY as a JSON object matching the schema below. Do not include any markdown formatting outside the JSON block.*

CRITICAL RULES:

- *EVERY feature ID provided in the prompt MUST be assigned to exactly one cluster. Do not leave any feature behind.*
- *You MUST group the features into EXACTLY $n_clusters$ clusters. Number the $cluster_id$ from 0 to $\{n_clusters - 1\}$.*
- *Each cluster has AT LEAST one feature. ""*

LLM Captioning Prompt: ""

You are an AI interpretability assistant analyzing neural network features. You must respond in valid JSON.

Here are the highest activating text snippets for Feature {feat_idx}:{context_str_joined}

Describe what concept, pattern, or semantic meaning this feature activates on in EXACTLY ONE SHORT SENTENCE.

Do not add any conversational filler.

Output your response strictly as a JSON object with a single key 'caption'. ""

LLM Selection Prompt: ""

You are an AI interpretability assistant analyzing neural network features. You must respond in valid JSON.

You are given a full catalog of features and their single-sentence descriptions.

FULL FEATURE CATALOG: {full_catalog_str}

Your task is to find the top {k} most semantically related features for the following target features.

TARGET FEATURES TO MATCH: {target_str}

RULES:

1. Return EXACTLY {k} matches for each target feature.
2. DO NOT match a target feature to itself.
3. Output your response STRICTLY as a JSON object where keys are the target feature IDs (as strings) and values are lists of {k} integer feature IDs. ""

D Ablation Study: Why do we need feature as distribution?

In this section, we run ablation experiments showing that representing feature as distribution preserves more semantic information than using one vector. We implement the following baseline, which first project the activation hidden states to the reference space but then average the distribution into one singular vector. By comparing with this baseline, we will show that our method surpass representing via one vector, even when projecting onto the same manifold:

Naive Manifold Projection (Naive). This baseline projects features into the shared reference space $\mathcal{R}$ but ignores the full distributional geometry. Instead of computing the Wasserstein transport plan, it collapses each projected feature distribution into a single activation-weighted centroid. For a source feature $j \in \mathcal{F}^{(B)}$, the centroid is defined as the expected position of its activated contexts in the reference space:

$$\bar{z}_j^{(B)} = \sum_{t \in \{t, a_{j,t}^{(B)} \geq 0\}} \hat{w}_{j,t}^{(B)} z_t^{(\mathcal{R})} \quad (10)$$

We similarly define the centroid $\bar{z}_i^{(A)}$ for a target feature $i \in \mathcal{F}^{(A)}$. The distance matrix is then computed by evaluating the ground cost $c(\cdot, \cdot)$ strictly between these centroids, discarding the multi-modal activation structure:

$$\left(\hat{D}_{\text{Naive}} \right)_{i,j} = c \left(\bar{z}_i^{(A)}, \bar{z}_j^{(B)} \right) \quad (11)$$

We run the same experiments as in Section 5.1, 5.2, 5.3; the results are in Table 5, Table 6, and Figure 11 respectively. We find that for feature matching, our method have stronger performance on the majority of the setups. Similarly, for circuit compression and digit addition, our method is better than naive method. This shows that represent features as distributions preserve more semantic information than representing with one vector, even in the setup of projecting to the same manifold.

Table 5: Feature matching experiment results when comparing with Naive method.

Method	GPT2				Gemma-2-2B			
	LLM Eval $\uparrow$	Acc $\uparrow$	CE $\downarrow$	VE $\uparrow$	LLM Eval $\uparrow$	Acc $\uparrow$	CE $\downarrow$	VE $\uparrow$
	Layer 5 match with Layer 6				Layer 12 match with Layer 13			
Our	2.53 $\pm$ 0.04	68.2 $\pm$ 1.9	3.775	0.9866	2.32 $\pm$ 0.04	58.6 $\pm$ 2.0	2.74784	0.7196
Our Naive	2.51 $\pm$ 0.04	66.4 $\pm$ 2.3	3.777	0.9865	2.22 $\pm$ 0.04	55.2 $\pm$ 2.1	2.74799	0.7183
	Layer 5 match with Layer 11				Layer 12 match with Layer 25			
Our	1.56 $\pm$ 0.04	23.2 $\pm$ 2.2	3.806	0.9127	1.83 $\pm$ 0.04	33.6 $\pm$ 2.9	2.74809	0.9115
Our Naive	1.56 $\pm$ 0.04	22.4 $\pm$ 2.3	3.822	0.9117	1.94 $\pm$ 0.02	39.6 $\pm$ 0.7	2.74806	0.9115
	Layer 0 match with Layer 11				Layer 0 match with Layer 25			
Our	1.39 $\pm$ 0.03	17.2 $\pm$ 2.3	3.828	0.9111	1.83 $\pm$ 0.05	34.8 $\pm$ 2.1	2.74866	0.9094
Our Naive	1.48 $\pm$ 0.01	19.6 $\pm$ 1.1	3.820	0.9116	1.78 $\pm$ 0.03	32.6 $\pm$ 1.6	2.74862	0.9094

Table 6: Circuit compression result comparing with Naive method.

Method	Gemma-2-2B	GPT2-small
Our	0.6151 $\pm$ 0.0153	0.6812 $\pm$ 0.0173
Our Naive	0.5950 $\pm$ 0.0128	0.6421 $\pm$ 0.0177

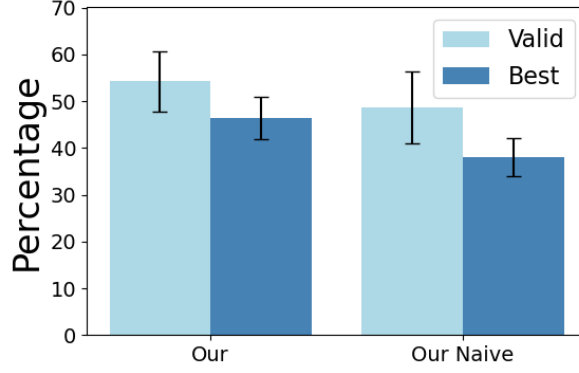


Figure 11: Digit addition comparing with Naive method. “Valid” means the user agree with the matching, and “Best” means the method has the best match among all methods.

E Additional Theoretical Properties

E.1 Activation Rescaling Invariance of Projected Feature Distributions

Proposition 1 (Activation rescaling invariance of projected feature distributions). *Let $i \in \mathcal{F}^{(A)}$ be a target feature with activation function $a_i^{(A)} : \mathcal{X}^{(A)} \rightarrow \mathbb{R}_{\geq 0}$. For any $\alpha > 0$, define $\tilde{a}_i^{(A)}(x) = \alpha a_i^{(A)}(x)$. Under the deterministic TopK convention, the projected empirical distribution induced by $\tilde{a}_i^{(A)}$ is identical to the one induced by $a_i^{(A)}$: $\tilde{\mu}_{i,T}^{(A \rightarrow \mathcal{R})} = \hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}$. Consequently, for any source feature $j \in \mathcal{F}^{(B)}$ and any ground cost $c : \mathcal{R} \times \mathcal{R} \rightarrow \mathbb{R}_{\geq 0}$, $W_c(\tilde{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}) = W_c(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})})$.*

Proof. Let $\mathcal{I} = \mathcal{I}_{i,T}^{(A),K}$ denote the top- K index set selected by the original activation values $a_i^{(A)}(x_t^{(A)})$, and let $\tilde{\mathcal{I}} = \tilde{\mathcal{I}}_{i,T}^{(A),K}$ denote the top- K index set selected by the rescaled activation values $\tilde{a}_i^{(A)}(x_t^{(A)}) = \alpha a_i^{(A)}(x_t^{(A)})$. We first show that the selected indices are unchanged. For any token positions $p, q \in [T]$, because $\alpha > 0$,

$$\begin{aligned} a_i^{(A)}(x_p^{(A)}) \geq a_i^{(A)}(x_q^{(A)}) &\iff \alpha a_i^{(A)}(x_p^{(A)}) \geq \alpha a_i^{(A)}(x_q^{(A)}) \\ &\iff \tilde{a}_i^{(A)}(x_p^{(A)}) \geq \tilde{a}_i^{(A)}(x_q^{(A)}). \end{aligned}$$

Thus every pairwise ordering relation among token activations is preserved. Moreover, equality is also preserved:

$$a_i^{(A)}(x_p^{(A)}) = a_i^{(A)}(x_q^{(A)}) \iff \tilde{a}_i^{(A)}(x_p^{(A)}) = \tilde{a}_i^{(A)}(x_q^{(A)}).$$

Therefore the rescaling neither changes the ranking nor creates or removes ties. Since TopK uses a deterministic tie-breaking convention, the same ordered top- K candidates are selected before and after rescaling. Hence $\tilde{\mathcal{I}} = \mathcal{I}$. We next compare the normalized activation weights. By construction,

for every selected token position $t \in \mathcal{I} = \tilde{\mathcal{I}}$,

$$\begin{aligned}
\tilde{w}_{i,t}^{(A)} &= \frac{\tilde{a}_i^{(A)}(x_t^{(A)})}{\sum_{s \in \tilde{\mathcal{I}}} \tilde{a}_i^{(A)}(x_s^{(A)})} \\
&= \frac{\alpha a_i^{(A)}(x_t^{(A)})}{\sum_{s \in \mathcal{I}} \alpha a_i^{(A)}(x_s^{(A)})} \quad \text{since } \tilde{\mathcal{I}} = \mathcal{I} \\
&= \frac{\alpha a_i^{(A)}(x_t^{(A)})}{\alpha \sum_{s \in \mathcal{I}} a_i^{(A)}(x_s^{(A)})} \\
&= \frac{a_i^{(A)}(x_t^{(A)})}{\sum_{s \in \mathcal{I}} a_i^{(A)}(x_s^{(A)})} = \hat{w}_{i,t}^{(A)}.
\end{aligned}$$

Thus, global positive rescaling preserves not only which token positions are selected, but also the probability mass assigned to every selected token position. It remains to compare the projected empirical distributions. In both constructions, the token position t is projected to the same support point $z_t^{(\mathcal{R})} \in \mathcal{R}$. Since the selected index set and the normalized weights are both unchanged, we have

$$\begin{aligned}
\tilde{\mu}_{i,T}^{(A \rightarrow \mathcal{R})} &= \sum_{t \in \tilde{\mathcal{I}}} \tilde{w}_{i,t}^{(A)} \delta_{z_t^{(\mathcal{R})}} \\
&= \sum_{t \in \mathcal{I}} \tilde{w}_{i,t}^{(A)} \delta_{z_t^{(\mathcal{R})}} \quad \text{since } \tilde{\mathcal{I}} = \mathcal{I} \\
&= \sum_{t \in \mathcal{I}} \hat{w}_{i,t}^{(A)} \delta_{z_t^{(\mathcal{R})}} \quad \text{since } \tilde{w}_{i,t}^{(A)} = \hat{w}_{i,t}^{(A)} \text{ for all } t \in \mathcal{I} \\
&= \hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}.
\end{aligned}$$

Therefore the projected empirical distribution induced by the rescaled activation function is exactly the same probability distribution as the original projected empirical distribution. Finally, Wasserstein distance depends only on its input distributions and the ground cost. Since the source distribution is identical in the two comparisons and the source distribution $\hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}$ is fixed,

$$W_c\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right) = W_c\left(\tilde{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right).$$

This proves both the invariance of the projected feature distribution and the invariance of all downstream Wasserstein scores involving this source feature. $\square$

Interpretation. This proposition shows that the projected activation-weighted representation depends on the relative activation profile of a feature, not on its raw activation scale. Positive global rescaling preserves the selected token positions, normalized activation weights, projected empirical distribution, and all downstream Wasserstein scores. This is important because SAE features from different layers or training runs may have different activation scales, while their semantic identity should be determined by where they activate strongly and by their relative activation pattern.

E.2 Flow Cost as a Refinement of Positional Cost

We state the result in the shared reference space $\mathcal{R}$, where all projected feature distributions are compared. In particular, one may take $\mu = \hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}$ and $\nu = \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}$.

Proposition 2 (Flow cost refines positional cost). *Let $\phi : \mathcal{R} \rightarrow \mathbb{R}^p$ be a positional representation map. For $m = 1, \dots, M$, let $\bar{u}_\theta(\tau_m, \cdot) : \mathcal{R} \rightarrow \mathbb{R}^s$ denote the normalized flow field evaluated at time point τ_m . For $z, z' \in \mathcal{R}$, define the positional cost $c_{\text{pos}}(z, z') = \|\phi(z) - \phi(z')\|_2^2$, the directional cost $c_{\text{dir}}(z, z') = \frac{1}{M} \sum_{m=1}^M \|\bar{u}_\theta(\tau_m, z) - \bar{u}_\theta(\tau_m, z')\|_2^2$, and the composite flow cost*

$$c_{\text{flow}}(z, z') = \lambda_x c_{\text{pos}}(z, z') + \lambda_u c_{\text{dir}}(z, z'), \quad \lambda_x, \lambda_u \geq 0.$$

Then, for any $\mu, \nu \in \mathcal{P}(\mathcal{R})$,

$$W_{c_{\text{flow}}}(\mu, \nu) \geq \lambda_x W_{c_{\text{pos}}}(\mu, \nu), \quad W_{c_{\text{flow}}}(\mu, \nu) \geq \lambda_u W_{c_{\text{dir}}}(\mu, \nu),$$

and more strongly, $W_{c_{\text{flow}}}(\mu, \nu) \geq \lambda_x W_{c_{\text{pos}}}(\mu, \nu) + \lambda_u W_{c_{\text{dir}}}(\mu, \nu)$.

Proof. By the definition of Wasserstein cost with ground cost c_{flow} , and by the decomposition $c_{\text{flow}} = \lambda_x c_{\text{pos}} + \lambda_u c_{\text{dir}}$, we have

$$\begin{aligned} W_{c_{\text{flow}}}(\mu, \nu) &= \inf_{\pi \in \Pi(\mu, \nu)} \mathbb{E}_{(z, z') \sim \pi} [c_{\text{flow}}(z, z')] \\ &= \inf_{\pi \in \Pi(\mu, \nu)} \mathbb{E}_{(z, z') \sim \pi} [\lambda_x c_{\text{pos}}(z, z') + \lambda_u c_{\text{dir}}(z, z')] \\ &= \inf_{\pi \in \Pi(\mu, \nu)} \{ \lambda_x \mathbb{E}_{\pi} [c_{\text{pos}}(z, z')] + \lambda_u \mathbb{E}_{\pi} [c_{\text{dir}}(z, z')] \}. \end{aligned}$$

For any fixed coupling $\pi \in \Pi(\mu, \nu)$, its expected positional cost is at least the optimal positional transport cost, and its expected directional cost is at least the optimal directional transport cost:

$$\begin{aligned} \mathbb{E}_{\pi} [c_{\text{pos}}(z, z')] &\geq \inf_{\gamma \in \Pi(\mu, \nu)} \mathbb{E}_{\gamma} [c_{\text{pos}}(z, z')] = W_{c_{\text{pos}}}(\mu, \nu), \\ \mathbb{E}_{\pi} [c_{\text{dir}}(z, z')] &\geq \inf_{\gamma \in \Pi(\mu, \nu)} \mathbb{E}_{\gamma} [c_{\text{dir}}(z, z')] = W_{c_{\text{dir}}}(\mu, \nu). \end{aligned}$$

Since $\lambda_x, \lambda_u \geq 0$, multiplying these two inequalities preserves their directions. Hence, for every coupling $\pi \in \Pi(\mu, \nu)$,

$$\begin{aligned} \lambda_x \mathbb{E}_{\pi} [c_{\text{pos}}(z, z')] + \lambda_u \mathbb{E}_{\pi} [c_{\text{dir}}(z, z')] \\ \geq \lambda_x W_{c_{\text{pos}}}(\mu, \nu) + \lambda_u W_{c_{\text{dir}}}(\mu, \nu). \end{aligned}$$

The right-hand side is independent of π . Therefore, taking the infimum over all couplings on the left-hand side gives

$$\begin{aligned} W_{c_{\text{flow}}}(\mu, \nu) &= \inf_{\pi \in \Pi(\mu, \nu)} \{ \lambda_x \mathbb{E}_{\pi} [c_{\text{pos}}(z, z')] + \lambda_u \mathbb{E}_{\pi} [c_{\text{dir}}(z, z')] \} \\ &\geq \lambda_x W_{c_{\text{pos}}}(\mu, \nu) + \lambda_u W_{c_{\text{dir}}}(\mu, \nu). \end{aligned}$$

This proves the stronger inequality. The two individual lower bounds follow from nonnegativity:

$$\begin{aligned} W_{c_{\text{flow}}}(\mu, \nu) &\geq \lambda_x W_{c_{\text{pos}}}(\mu, \nu) + \lambda_u W_{c_{\text{dir}}}(\mu, \nu) \geq \lambda_x W_{c_{\text{pos}}}(\mu, \nu), \\ W_{c_{\text{flow}}}(\mu, \nu) &\geq \lambda_x W_{c_{\text{pos}}}(\mu, \nu) + \lambda_u W_{c_{\text{dir}}}(\mu, \nu) \geq \lambda_u W_{c_{\text{dir}}}(\mu, \nu). \end{aligned}$$

Thus the composite flow cost lower-bounds both weighted component Wasserstein costs and also lower-bounds their sum. $\square$

Interpretation. This proposition shows that the flow-aware ground cost is a refinement of positional geometry in the shared reference space $\mathcal{R}$. The positional term c_{pos} compares where two projected activation distributions lie, while the directional term c_{dir} compares how the normalized flow field behaves around those reference-space points. The composite cost c_{flow} therefore adds a directional penalty on top of positional matching. The inequality implies that matching under c_{flow} is at least as strict as matching under either weighted component alone. In particular, two projected feature distributions may be close positionally but far under the flow-aware cost if their directional fields disagree. When $\lambda_u = 0$, the cost reduces to ordinary positional Wasserstein matching: $c_{\text{flow}}(z, z') = \lambda_x c_{\text{pos}}(z, z')$.

E.3 Stability of Wasserstein Feature Distance

We prove the stability result in the shared reference space. Throughout this section, let $(\mathcal{R}, \rho)$ be the metric space where all projected feature distributions live. Let $\mu, \tilde{\mu}, \nu, \tilde{\nu} \in \mathcal{P}(\mathcal{R})$, where (μ, ν) denotes the original pair of projected feature distributions and $(\tilde{\mu}, \tilde{\nu})$ denotes the perturbed pair. Define the relevant supports

$$\mathcal{S}_{\mu} = \text{supp}(\mu) \cup \text{supp}(\tilde{\mu}), \quad \mathcal{S}_{\nu} = \text{supp}(\nu) \cup \text{supp}(\tilde{\nu}).$$

A ground cost $c : \mathcal{R} \times \mathcal{R} \rightarrow \mathbb{R}_{\geq 0}$ is coordinate-wise Lipschitz on the relevant support if there exist constants $L_x, L_y < \infty$ such that

$$|c(z, z') - c(\bar{z}, \bar{z}')| \leq L_x \rho(z, \bar{z}), \quad \forall z, \bar{z} \in \mathcal{S}_{\mu}, \forall z', \bar{z}' \in \mathcal{S}_{\nu},$$

and

$$|c(z, z') - c(z, \bar{z}')| \leq L_y \rho(z', \bar{z}'), \quad \forall z \in \mathcal{S}_{\mu}, \forall z', \bar{z}' \in \mathcal{S}_{\nu}.$$

Theorem 1 (Stability of Wasserstein feature distance). Let $(\mathcal{R}, \rho)$ be the shared reference metric space. Let $c : \mathcal{R} \times \mathcal{R} \rightarrow \mathbb{R}_{\geq 0}$ be coordinate-wise Lipschitz on the relevant support with constants L_x, L_y . Then, for any projected feature distributions $\mu, \tilde{\mu}, \nu, \tilde{\nu} \in \mathcal{P}(\mathcal{R})$,

$$|W_c(\mu, \nu) - W_c(\tilde{\mu}, \tilde{\nu})| \leq L_x W_\rho(\mu, \tilde{\mu}) + L_y W_\rho(\nu, \tilde{\nu}),$$

where $W_\rho(\mu, \tilde{\mu}) = \inf_{\alpha \in \Pi(\mu, \tilde{\mu})} \mathbb{E}_{(z, \tilde{z}) \sim \alpha} [\rho(z, \tilde{z})]$ is the Wasserstein-1 distance induced by the base metric ρ .

Proof. We first prove the one-sided inequality

$$W_c(\mu, \nu) - W_c(\tilde{\mu}, \tilde{\nu}) \leq L_x W_\rho(\mu, \tilde{\mu}) + L_y W_\rho(\nu, \tilde{\nu}).$$

Fix any $\eta > 0$. By the definition of infimum, choose couplings $\alpha_\eta \in \Pi(\mu, \tilde{\mu})$, $\beta_\eta \in \Pi(\tilde{\nu}, \nu)$, and $\tilde{\pi}_\eta \in \Pi(\tilde{\mu}, \tilde{\nu})$ such that

$$\mathbb{E}_{\alpha_\eta}[\rho(z, \tilde{z})] \leq W_\rho(\mu, \tilde{\mu}) + \eta,$$

$$\mathbb{E}_{\beta_\eta}[\rho(\tilde{z}', z')] \leq W_\rho(\tilde{\nu}, \nu) + \eta = W_\rho(\nu, \tilde{\nu}) + \eta,$$

$$\mathbb{E}_{\tilde{\pi}_\eta}[c(\tilde{z}, \tilde{z}')] \leq W_c(\tilde{\mu}, \tilde{\nu}) + \eta.$$

The first coupling connects μ to $\tilde{\mu}$, the middle coupling connects $\tilde{\mu}$ to $\tilde{\nu}$, and the third coupling connects $\tilde{\nu}$ to ν . Since these couplings share the required intermediate marginals, the gluing lemma gives a joint distribution $\Lambda_\eta \in \mathcal{P}(\mathcal{R}^4)$ over $(z, \tilde{z}, \tilde{z}', z')$ such that

$$(z, \tilde{z})_{\#} \Lambda_\eta = \alpha_\eta, \quad (\tilde{z}, \tilde{z}')_{\#} \Lambda_\eta = \tilde{\pi}_\eta, \quad (\tilde{z}', z')_{\#} \Lambda_\eta = \beta_\eta.$$

Let $\pi_\eta = (z, z')_{\#} \Lambda_\eta$. Since the first marginal of Λ_η is μ and the fourth marginal is ν , we have $\pi_\eta \in \Pi(\mu, \nu)$. Hence

$$\begin{aligned} W_c(\mu, \nu) &= \inf_{\kappa \in \Pi(\mu, \nu)} \mathbb{E}_\kappa[c(z, z')] \\ &\leq \mathbb{E}_{\pi_\eta}[c(z, z')] = \mathbb{E}_{\Lambda_\eta}[c(z, z')]. \end{aligned}$$

For every $(z, \tilde{z}, \tilde{z}', z')$ in the support of Λ_η , we have $z, \tilde{z} \in \mathcal{S}_\mu$ and $z', \tilde{z}' \in \mathcal{S}_\nu$. Therefore, by coordinate-wise Lipschitzness,

$$\begin{aligned} c(z, z') - c(\tilde{z}, \tilde{z}') &= [c(z, z') - c(\tilde{z}, z')] + [c(\tilde{z}, z') - c(\tilde{z}, \tilde{z}')] \\ &\leq |c(z, z') - c(\tilde{z}, z')| + |c(\tilde{z}, z') - c(\tilde{z}, \tilde{z}')| \\ &\leq L_x \rho(z, \tilde{z}) + L_y \rho(z', \tilde{z}') \\ &= L_x \rho(z, \tilde{z}) + L_y \rho(\tilde{z}', z'). \end{aligned}$$

Equivalently,

$$c(z, z') \leq c(\tilde{z}, \tilde{z}') + L_x \rho(z, \tilde{z}) + L_y \rho(\tilde{z}', z').$$

Taking expectation under Λ_η and using its prescribed marginals gives

$$\begin{aligned} \mathbb{E}_{\Lambda_\eta}[c(z, z')] &\leq \mathbb{E}_{\Lambda_\eta}[c(\tilde{z}, \tilde{z}')] + L_x \mathbb{E}_{\Lambda_\eta}[\rho(z, \tilde{z})] + L_y \mathbb{E}_{\Lambda_\eta}[\rho(\tilde{z}', z')] \\ &= \mathbb{E}_{\tilde{\pi}_\eta}[c(\tilde{z}, \tilde{z}')] + L_x \mathbb{E}_{\alpha_\eta}[\rho(z, \tilde{z})] + L_y \mathbb{E}_{\beta_\eta}[\rho(\tilde{z}', z')] \\ &\leq [W_c(\tilde{\mu}, \tilde{\nu}) + \eta] + L_x [W_\rho(\mu, \tilde{\mu}) + \eta] + L_y [W_\rho(\nu, \tilde{\nu}) + \eta] \\ &= W_c(\tilde{\mu}, \tilde{\nu}) + L_x W_\rho(\mu, \tilde{\mu}) + L_y W_\rho(\nu, \tilde{\nu}) + (1 + L_x + L_y)\eta. \end{aligned}$$

Combining this with $W_c(\mu, \nu) \leq \mathbb{E}_{\Lambda_\eta}[c(z, z')]$ yields

$$W_c(\mu, \nu) - W_c(\tilde{\mu}, \tilde{\nu}) \leq L_x W_\rho(\mu, \tilde{\mu}) + L_y W_\rho(\nu, \tilde{\nu}) + (1 + L_x + L_y)\eta.$$

Since $\eta > 0$ was arbitrary, letting $\eta \downarrow 0$ gives

$$W_c(\mu, \nu) - W_c(\tilde{\mu}, \tilde{\nu}) \leq L_x W_\rho(\mu, \tilde{\mu}) + L_y W_\rho(\nu, \tilde{\nu}).$$

For the reverse direction, apply the same argument with (μ, ν) and $(\tilde{\mu}, \tilde{\nu})$ interchanged. The relevant supports are unchanged because they are unions of the same original and perturbed supports. Hence

$$\begin{aligned} W_c(\tilde{\mu}, \tilde{\nu}) - W_c(\mu, \nu) &\leq L_x W_\rho(\tilde{\mu}, \mu) + L_y W_\rho(\tilde{\nu}, \nu) \\ &= L_x W_\rho(\mu, \tilde{\mu}) + L_y W_\rho(\nu, \tilde{\nu}), \end{aligned}$$

where the last equality uses symmetry of W_ρ . Combining the two one-sided bounds gives

$$|W_c(\mu, \nu) - W_c(\tilde{\mu}, \tilde{\nu})| \leq L_x W_\rho(\mu, \tilde{\mu}) + L_y W_\rho(\nu, \tilde{\nu}).$$

This completes the proof. $\square$

Interpretation. This theorem shows that Wasserstein distances between projected feature distributions are stable under small perturbations in the shared reference space. In our setting, one can instantiate $\mu = \hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}$ and $\nu = \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}$. If the projected empirical distribution of the target feature changes by $W_\rho(\mu, \tilde{\mu})$ and the projected empirical distribution of the source feature changes by $W_\rho(\nu, \tilde{\nu})$, then the Wasserstein feature score changes by at most the corresponding Lipschitz-weighted sum. Thus, the method is stable as long as the activation-induced projected distributions remain close in the shared reference geometry.

E.4 Metric-Specific Stability Corollaries

We now instantiate Theorem 1 for several commonly used ground costs on the shared reference space $\mathcal{R}$. Throughout this section, let $\mu, \tilde{\mu}, \nu, \tilde{\nu} \in \mathcal{P}(\mathcal{R})$ be projected feature distributions, and define

$$\delta_\rho := W_\rho(\mu, \tilde{\mu}) + W_\rho(\nu, \tilde{\nu}).$$

Here, μ and ν denote the original target-side and source-side projected feature distributions, while $\tilde{\mu}$ and $\tilde{\nu}$ denote their perturbed versions. We use z for a target-side reference-space point and z' for a source-side reference-space point. Later, $\bar{z}$ and $\bar{z}'$ denote perturbed reference-space points used to verify coordinate-wise Lipschitz conditions.

Let

$$\mathcal{S}_\mu := \text{supp}(\mu) \cup \text{supp}(\tilde{\mu}), \quad \mathcal{S}_\nu := \text{supp}(\nu) \cup \text{supp}(\tilde{\nu}).$$

Thus, $z \in \mathcal{S}_\mu$ ranges over target-side support points and $z' \in \mathcal{S}_\nu$ ranges over source-side support points. If a ground cost $c : \mathcal{R} \times \mathcal{R} \rightarrow \mathbb{R}_{\geq 0}$ satisfies

$$|W_c(\mu, \nu) - W_c(\tilde{\mu}, \tilde{\nu})| \leq K_c \delta_\rho,$$

then K_c is its raw worst-case stability constant. Raw constants are scale-dependent: replacing c by αc also replaces K_c by αK_c . Therefore, when possible, we use an on-support range bound $B_c \geq \sup_{(z, z') \in \mathcal{S}_\mu \times \mathcal{S}_\nu} c(z, z')$, and compare costs by the normalized stability constant $\bar{K}_c := K_c/B_c$. A smaller $\bar{K}_c$ means that, relative to the numerical scale of the cost, the Wasserstein feature distance is less sensitive to perturbations of the projected feature distributions.

Common support assumptions. Let $\phi : \mathcal{R} \rightarrow \mathbb{R}^p$ be a representation map on the shared reference space. Assume that on the relevant support,

$$\|\phi(z)\|_2 \leq R_\phi, \quad \forall z \in \mathcal{S}_\mu \cup \mathcal{S}_\nu,$$

and, when cosine cost is used, assume additionally that

$$\|\phi(z)\|_2 \geq r_\phi > 0, \quad \forall z \in \mathcal{S}_\mu \cup \mathcal{S}_\nu.$$

Assume that ϕ is L_ϕ -Lipschitz with respect to ρ :

$$\|\phi(z) - \phi(\bar{z})\|_2 \leq L_\phi \rho(z, \bar{z}), \quad \forall z, \bar{z} \in \mathcal{S}_\mu \cup \mathcal{S}_\nu.$$

For flow-direction costs, define

$$q_m(z) := \bar{u}_\theta(\tau_m, z) = \frac{u_\theta(\tau_m, z)}{\|u_\theta(\tau_m, z)\|_2 + \varepsilon_{\text{norm}}}, \quad m = 1, \dots, M.$$

Assume that each $q_m : \mathcal{R} \rightarrow \mathbb{R}^s$ is L_u -Lipschitz on the relevant support:

$$\|q_m(z) - q_m(\bar{z})\|_2 \leq L_u \rho(z, \bar{z}), \quad \forall m \in [M],$$

and since q_m is normalized, assume $\|q_m(z)\|_2 \leq 1$.

Corollary 1 (Stability under Euclidean positional cost). *Consider $c_{\text{Euc}}(z, z') = \|\phi(z) - \phi(z')\|_2$. Then $|W_{c_{\text{Euc}}}(\mu, \nu) - W_{c_{\text{Euc}}}(\tilde{\mu}, \tilde{\nu})| \leq L_\phi \delta_\rho$. Hence $K_{\text{Euc}} = L_\phi$. Moreover, $B_{\text{Euc}} = 2R_\phi$ and $\bar{K}_{\text{Euc}} = L_\phi/(2R_\phi)$.*

Proof. We verify the coordinate-wise Lipschitz constants and the range bound. For the first coordinate,

$$\begin{aligned} |c_{\text{Euc}}(z, z') - c_{\text{Euc}}(\bar{z}, \bar{z}')| &= |\|\phi(z) - \phi(z')\|_2 - \|\phi(\bar{z}) - \phi(\bar{z}')\|_2| \\ &\leq \|(\phi(z) - \phi(z')) - (\phi(\bar{z}) - \phi(\bar{z}'))\|_2 \\ &= \|\phi(z) - \phi(\bar{z})\|_2 \\ &\leq L_\phi \rho(z, \bar{z}). \end{aligned}$$

The same argument in the second coordinate gives

$$\begin{aligned} |c_{\text{Euc}}(z, z') - c_{\text{Euc}}(z, \bar{z}')| &= |\|\phi(z) - \phi(z')\|_2 - \|\phi(z) - \phi(\bar{z}')\|_2| \\ &\leq \|(\phi(z) - \phi(z')) - (\phi(z) - \phi(\bar{z}'))\|_2 \\ &= \|\phi(z') - \phi(\bar{z}')\|_2 \leq L_\phi \rho(z', \bar{z}'). \end{aligned}$$

Thus c_{Euc} is coordinate-wise Lipschitz with $L_x = L_y = L_\phi$. Applying Theorem 1 gives

$$|W_{c_{\text{Euc}}}(\mu, \nu) - W_{c_{\text{Euc}}}(\tilde{\mu}, \tilde{\nu})| \leq L_\phi W_\rho(\mu, \tilde{\mu}) + L_\phi W_\rho(\nu, \tilde{\nu}) = L_\phi \delta_\rho.$$

For the range bound, for every $(z, z') \in \mathcal{S}_\mu \times \mathcal{S}_\nu$,

$$\begin{aligned} c_{\text{Euc}}(z, z') &= \|\phi(z) - \phi(z')\|_2 \leq \|\phi(z)\|_2 + \|\phi(z')\|_2 \\ &\leq R_\phi + R_\phi = 2R_\phi. \end{aligned}$$

Hence $B_{\text{Euc}} = 2R_\phi$ and $\bar{K}_{\text{Euc}} = K_{\text{Euc}}/B_{\text{Euc}} = L_\phi/(2R_\phi)$. $\square$

Interpretation. Euclidean cost is stable when the representation map ϕ is smooth. Its normalized stability depends on $L_\phi/(2R_\phi)$.

Corollary 2 (Stability under squared Euclidean positional cost). *Consider $c_{\text{sq}}(z, z') = \|\phi(z) - \phi(z')\|_2^2$. Then $|W_{c_{\text{sq}}}(\mu, \nu) - W_{c_{\text{sq}}}(\tilde{\mu}, \tilde{\nu})| \leq 4R_\phi L_\phi \delta_\rho$. Hence $K_{\text{sq}} = 4R_\phi L_\phi$. Moreover, $B_{\text{sq}} = 4R_\phi^2$ and $\bar{K}_{\text{sq}} = L_\phi/R_\phi$.*

Proof. For the first coordinate, use $\|a\|_2^2 - \|b\|_2^2 = \langle a + b, a - b \rangle$ with $a = \phi(z) - \phi(z')$ and $b = \phi(\bar{z}) - \phi(\bar{z}')$. Then

$$\begin{aligned} |c_{\text{sq}}(z, z') - c_{\text{sq}}(\bar{z}, \bar{z}')| &= |\|\phi(z) - \phi(z')\|_2^2 - \|\phi(\bar{z}) - \phi(\bar{z}')\|_2^2| \\ &= |\langle \phi(z) - \phi(z') + \phi(\bar{z}) - \phi(\bar{z}'), \phi(z) - \phi(\bar{z}') \rangle| \\ &\leq (\|\phi(z) - \phi(z')\|_2 + \|\phi(\bar{z}) - \phi(\bar{z}')\|_2) \|\phi(z) - \phi(\bar{z}')\|_2 \\ &\leq (\|\phi(z)\|_2 + \|\phi(z')\|_2 + \|\phi(\bar{z})\|_2 + \|\phi(\bar{z}')\|_2) L_\phi \rho(z, \bar{z}) \\ &\leq (R_\phi + R_\phi + R_\phi + R_\phi) L_\phi \rho(z, \bar{z}) \\ &= 4R_\phi L_\phi \rho(z, \bar{z}). \end{aligned}$$

The same computation in the second coordinate gives

$$\begin{aligned} |c_{\text{sq}}(z, z') - c_{\text{sq}}(z, \bar{z}')| &\leq (\|\phi(z) - \phi(z')\|_2 + \|\phi(z) - \phi(\bar{z}')\|_2) \|\phi(z') - \phi(\bar{z}')\|_2 \\ &\leq 4R_\phi L_\phi \rho(z', \bar{z}'). \end{aligned}$$

Thus c_{sq} is coordinate-wise Lipschitz with $L_x = L_y = 4R_\phi L_\phi$, so Theorem 1 yields

$$|W_{c_{\text{sq}}}(\mu, \nu) - W_{c_{\text{sq}}}(\tilde{\mu}, \tilde{\nu})| \leq 4R_\phi L_\phi \delta_\rho.$$

For the range bound,

$$\begin{aligned} c_{\text{sq}}(z, z') &= \|\phi(z) - \phi(z')\|_2^2 \leq (\|\phi(z)\|_2 + \|\phi(z')\|_2)^2 \\ &\leq (2R_\phi)^2 = 4R_\phi^2. \end{aligned}$$

Hence $B_{\text{sq}} = 4R_\phi^2$ and $\bar{K}_{\text{sq}} = K_{\text{sq}}/B_{\text{sq}} = (4R_\phi L_\phi)/(4R_\phi^2) = L_\phi/R_\phi$. $\square$

Interpretation. Squared Euclidean cost has a larger raw stability constant than Euclidean cost. After normalization, $\bar{K}_{\text{sq}} = 2\bar{K}_{\text{Euc}}$.

Corollary 3 (Stability under Mahalanobis cost). *Let $M_{\text{Mah}} \in \mathbb{R}^{m \times p}$ and consider $c_{\text{Mah}}(z, z') = \|M_{\text{Mah}}(\phi(z) - \phi(z'))\|_2$. Then $|W_{c_{\text{Mah}}}(\mu, \nu) - W_{c_{\text{Mah}}}(\tilde{\mu}, \tilde{\nu})| \leq \|M_{\text{Mah}}\|_{\text{op}} L_\phi \delta_\rho$. Hence $K_{\text{Mah}} = \|M_{\text{Mah}}\|_{\text{op}} L_\phi$. Moreover, $B_{\text{Mah}} = 2R_\phi \|M_{\text{Mah}}\|_{\text{op}}$ and $\bar{K}_{\text{Mah}} = L_\phi/(2R_\phi)$.*

Proof. For the first coordinate,

$$\begin{aligned}
|c_{\text{Mah}}(z, z') - c_{\text{Mah}}(\bar{z}, z')| &= |\|M_{\text{Mah}}(\phi(z) - \phi(z'))\|_2 - \|M_{\text{Mah}}(\phi(\bar{z}) - \phi(z'))\|_2| \\
&\leq \|M_{\text{Mah}}(\phi(z) - \phi(z')) - M_{\text{Mah}}(\phi(\bar{z}) - \phi(z'))\|_2 \\
&= \|M_{\text{Mah}}(\phi(z) - \phi(\bar{z}))\|_2 \\
&\leq \|M_{\text{Mah}}\|_{\text{op}} \|\phi(z) - \phi(\bar{z})\|_2 \\
&\leq \|M_{\text{Mah}}\|_{\text{op}} L_\phi \rho(z, \bar{z}).
\end{aligned}$$

Similarly,

$$\begin{aligned}
|c_{\text{Mah}}(z, z') - c_{\text{Mah}}(z, \bar{z}')| &\leq \|M_{\text{Mah}}(\phi(z') - \phi(\bar{z}'))\|_2 \\
&\leq \|M_{\text{Mah}}\|_{\text{op}} L_\phi \rho(z', \bar{z}').
\end{aligned}$$

Therefore $L_x = L_y = \|M_{\text{Mah}}\|_{\text{op}} L_\phi$, and Theorem 1 gives

$$|W_{c_{\text{Mah}}}(\mu, \nu) - W_{c_{\text{Mah}}}(\tilde{\mu}, \tilde{\nu})| \leq \|M_{\text{Mah}}\|_{\text{op}} L_\phi \delta_\rho.$$

For the range bound,

$$\begin{aligned}
c_{\text{Mah}}(z, z') &= \|M_{\text{Mah}}(\phi(z) - \phi(z'))\|_2 \\
&\leq \|M_{\text{Mah}}\|_{\text{op}} \|\phi(z) - \phi(z')\|_2 \\
&\leq \|M_{\text{Mah}}\|_{\text{op}} (\|\phi(z)\|_2 + \|\phi(z')\|_2) \\
&\leq 2R_\phi \|M_{\text{Mah}}\|_{\text{op}}.
\end{aligned}$$

Thus $B_{\text{Mah}} = 2R_\phi \|M_{\text{Mah}}\|_{\text{op}}$ and $\bar{K}_{\text{Mah}} = (\|M_{\text{Mah}}\|_{\text{op}} L_\phi) / (2R_\phi \|M_{\text{Mah}}\|_{\text{op}}) = L_\phi / (2R_\phi)$. $\square$

Interpretation. Raw Mahalanobis stability is amplified by $\|M_{\text{Mah}}\|_{\text{op}}$, but this trivial scaling cancels after range normalization.

Corollary 4 (Stability under squared Mahalanobis cost). *Let $M_{\text{Mah}} \in \mathbb{R}^{m \times p}$ and consider $c_{\text{sqMah}}(z, z') = \|M_{\text{Mah}}(\phi(z) - \phi(z'))\|_2^2$. Then $|W_{c_{\text{sqMah}}}(\mu, \nu) - W_{c_{\text{sqMah}}}(\tilde{\mu}, \tilde{\nu})| \leq 4R_\phi \|M_{\text{Mah}}\|_{\text{op}}^2 L_\phi \delta_\rho$. Hence $K_{\text{sqMah}} = 4R_\phi \|M_{\text{Mah}}\|_{\text{op}}^2 L_\phi$. Moreover, $B_{\text{sqMah}} = 4R_\phi^2 \|M_{\text{Mah}}\|_{\text{op}}^2$ and $\bar{K}_{\text{sqMah}} = L_\phi / R_\phi$.*

Proof. For the first coordinate, again using $\|a\|_2^2 - \|b\|_2^2 = \langle a+b, a-b \rangle$ with $a = M_{\text{Mah}}(\phi(z) - \phi(z'))$ and $b = M_{\text{Mah}}(\phi(\bar{z}) - \phi(z'))$, we get

$$\begin{aligned}
|c_{\text{sqMah}}(z, z') - c_{\text{sqMah}}(\bar{z}, z')| &= |\|M_{\text{Mah}}(\phi(z) - \phi(z'))\|_2^2 - \|M_{\text{Mah}}(\phi(\bar{z}) - \phi(z'))\|_2^2| \\
&\leq (\|M_{\text{Mah}}(\phi(z) - \phi(z'))\|_2 + \|M_{\text{Mah}}(\phi(\bar{z}) - \phi(z'))\|_2) \|M_{\text{Mah}}(\phi(z) - \phi(\bar{z}))\|_2 \\
&\leq (\|M_{\text{Mah}}\|_{\text{op}} \|\phi(z) - \phi(z')\|_2 + \|M_{\text{Mah}}\|_{\text{op}} \|\phi(\bar{z}) - \phi(z')\|_2) \|M_{\text{Mah}}\|_{\text{op}} \|\phi(z) - \phi(\bar{z})\|_2 \\
&\leq (2R_\phi \|M_{\text{Mah}}\|_{\text{op}} + 2R_\phi \|M_{\text{Mah}}\|_{\text{op}}) \|M_{\text{Mah}}\|_{\text{op}} L_\phi \rho(z, \bar{z}) \\
&= 4R_\phi \|M_{\text{Mah}}\|_{\text{op}}^2 L_\phi \rho(z, \bar{z}).
\end{aligned}$$

The same argument in the second coordinate gives

$$|c_{\text{sqMah}}(z, z') - c_{\text{sqMah}}(z, \bar{z}')| \leq 4R_\phi \|M_{\text{Mah}}\|_{\text{op}}^2 L_\phi \rho(z', \bar{z}').$$

Therefore c_{sqMah} is coordinate-wise Lipschitz with $L_x = L_y = 4R_\phi \|M_{\text{Mah}}\|_{\text{op}}^2 L_\phi$, giving the stated stability bound. For the range,

$$\begin{aligned}
c_{\text{sqMah}}(z, z') &= \|M_{\text{Mah}}(\phi(z) - \phi(z'))\|_2^2 \\
&\leq \|M_{\text{Mah}}\|_{\text{op}}^2 \|\phi(z) - \phi(z')\|_2^2 \\
&\leq \|M_{\text{Mah}}\|_{\text{op}}^2 (2R_\phi)^2 = 4R_\phi^2 \|M_{\text{Mah}}\|_{\text{op}}^2.
\end{aligned}$$

Thus $\bar{K}_{\text{sqMah}} = (4R_\phi \|M_{\text{Mah}}\|_{\text{op}}^2 L_\phi) / (4R_\phi^2 \|M_{\text{Mah}}\|_{\text{op}}^2) = L_\phi / R_\phi$. $\square$

Interpretation. Squared Mahalanobis behaves like squared Euclidean after normalization.

Corollary 5 (Stability under cosine cost). *Define $\psi(z) = \phi(z)/\|\phi(z)\|_2$ and consider $c_{\cos}(z, z') = 1 - \langle \psi(z), \psi(z') \rangle$. Then $|W_{c_{\cos}}(\mu, \nu) - W_{c_{\cos}}(\tilde{\mu}, \tilde{\nu})| \leq (2L_\phi/r_\phi)\delta_\rho$. Hence $K_{\cos} = 2L_\phi/r_\phi$. Since $c_{\cos}(z, z') \in [0, 2]$, we may take $B_{\cos} = 2$ and $\bar{K}_{\cos} = L_\phi/r_\phi$.*

Proof. First, for any u, v with $\|u\|_2, \|v\|_2 \geq r_\phi$,

$$\begin{aligned} \left\| \frac{u}{\|u\|_2} - \frac{v}{\|v\|_2} \right\|_2 &= \left\| \frac{u-v}{\|u\|_2} + v \left(\frac{1}{\|u\|_2} - \frac{1}{\|v\|_2} \right) \right\|_2 \\ &\leq \frac{\|u-v\|_2}{\|u\|_2} + \|v\|_2 \frac{|\|v\|_2 - \|u\|_2|}{\|u\|_2 \|v\|_2} \\ &\leq \frac{\|u-v\|_2}{r_\phi} + \frac{|\|v\|_2 - \|u\|_2|}{r_\phi} \\ &\leq \frac{\|u-v\|_2}{r_\phi} + \frac{\|u-v\|_2}{r_\phi} = \frac{2\|u-v\|_2}{r_\phi}. \end{aligned}$$

Applying this with $u = \phi(z)$ and $v = \phi(\bar{z})$ gives

$$\|\psi(z) - \psi(\bar{z})\|_2 \leq \frac{2\|\phi(z) - \phi(\bar{z})\|_2}{r_\phi} \leq \frac{2L_\phi}{r_\phi} \rho(z, \bar{z}).$$

Hence, for the first coordinate,

$$\begin{aligned} |c_{\cos}(z, z') - c_{\cos}(\bar{z}, z')| &= |1 - \langle \psi(z), \psi(z') \rangle - (1 - \langle \psi(\bar{z}), \psi(z') \rangle)| \\ &= |\langle \psi(\bar{z}) - \psi(z), \psi(z') \rangle| \\ &\leq \|\psi(\bar{z}) - \psi(z)\|_2 \|\psi(z')\|_2 \\ &= \|\psi(\bar{z}) - \psi(z)\|_2 \\ &\leq \frac{2L_\phi}{r_\phi} \rho(z, \bar{z}). \end{aligned}$$

The second coordinate is identical. Thus $L_x = L_y = 2L_\phi/r_\phi$, and Theorem 1 gives the stability bound. Finally, since $\psi(z)$ and $\psi(z')$ are unit vectors,

$$-1 \leq \langle \psi(z), \psi(z') \rangle \leq 1 \implies 0 \leq c_{\cos}(z, z') \leq 2.$$

Thus $B_{\cos} = 2$ and $\bar{K}_{\cos} = (2L_\phi/r_\phi)/2 = L_\phi/r_\phi$. □

Interpretation. Cosine cost becomes sensitive when representation norms approach zero; this is captured by the factor $1/r_\phi$.

Corollary 6 (Stability under pure flow-direction cost). *Consider $c_{\text{dir}}(z, z') = M^{-1} \sum_{m=1}^M \|q_m(z) - q_m(z')\|_2$. Then $|W_{c_{\text{dir}}}(\mu, \nu) - W_{c_{\text{dir}}}(\tilde{\mu}, \tilde{\nu})| \leq L_u \delta_\rho$. Hence $K_{\text{dir}} = L_u$. Since $c_{\text{dir}}(z, z') \leq 2$, we may take $B_{\text{dir}} = 2$ and $\bar{K}_{\text{dir}} = L_u/2$.*

Proof. For the first coordinate,

$$\begin{aligned}
& |c_{\text{dir}}(z, z') - c_{\text{dir}}(\bar{z}, z')| \\
&= \left| \frac{1}{M} \sum_{m=1}^M \|q_m(z) - q_m(z')\|_2 - \frac{1}{M} \sum_{m=1}^M \|q_m(\bar{z}) - q_m(z')\|_2 \right| \\
&\leq \frac{1}{M} \sum_{m=1}^M \left| \|q_m(z) - q_m(z')\|_2 - \|q_m(\bar{z}) - q_m(z')\|_2 \right| \\
&\leq \frac{1}{M} \sum_{m=1}^M \|(q_m(z) - q_m(z')) - (q_m(\bar{z}) - q_m(z'))\|_2 \\
&= \frac{1}{M} \sum_{m=1}^M \|q_m(z) - q_m(\bar{z})\|_2 \\
&\leq \frac{1}{M} \sum_{m=1}^M L_u \rho(z, \bar{z}) = L_u \rho(z, \bar{z}).
\end{aligned}$$

The same argument gives

$$|c_{\text{dir}}(z, z') - c_{\text{dir}}(z, \bar{z}')| \leq L_u \rho(z', \bar{z}').$$

Thus $L_x = L_y = L_u$, and Theorem 1 gives

$$|W_{c_{\text{dir}}}(\mu, \nu) - W_{c_{\text{dir}}}(\tilde{\mu}, \tilde{\nu})| \leq L_u \delta_\rho.$$

For the range,

$$\begin{aligned}
c_{\text{dir}}(z, z') &= \frac{1}{M} \sum_{m=1}^M \|q_m(z) - q_m(z')\|_2 \\
&\leq \frac{1}{M} \sum_{m=1}^M (\|q_m(z)\|_2 + \|q_m(z')\|_2) \\
&\leq \frac{1}{M} \sum_{m=1}^M 2 = 2.
\end{aligned}$$

Hence $B_{\text{dir}} = 2$ and $\bar{K}_{\text{dir}} = L_u/2$. □

Interpretation. Pure flow-direction stability is controlled only by the smoothness L_u of the normalized velocity field.

Corollary 7 (Stability under squared pure flow-direction cost). *Consider $c_{\text{dir}}^{(2)}(z, z') = M^{-1} \sum_{m=1}^M \|q_m(z) - q_m(z')\|_2^2$. Then $|W_{c_{\text{dir}}^{(2)}}(\mu, \nu) - W_{c_{\text{dir}}^{(2)}}(\tilde{\mu}, \tilde{\nu})| \leq 4L_u \delta_\rho$. Hence $K_{\text{dir}}^{(2)} = 4L_u$. Since $c_{\text{dir}}^{(2)}(z, z') \leq 4$, we may take $B_{\text{dir}}^{(2)} = 4$ and $\bar{K}_{\text{dir}}^{(2)} = L_u$.*

Proof. For each m , using again $\|a\|_2^2 - \|b\|_2^2 = \langle a + b, a - b \rangle$ with $a = q_m(z) - q_m(z')$ and $b = q_m(\bar{z}) - q_m(z')$, we obtain

$$\begin{aligned}
& \left| \|q_m(z) - q_m(z')\|_2^2 - \|q_m(\bar{z}) - q_m(z')\|_2^2 \right| \\
&= |\langle q_m(z) - q_m(z') + q_m(\bar{z}) - q_m(z'), q_m(z) - q_m(\bar{z}) \rangle| \\
&\leq (\|q_m(z) - q_m(z')\|_2 + \|q_m(\bar{z}) - q_m(z')\|_2) \|q_m(z) - q_m(\bar{z})\|_2 \\
&\leq (\|q_m(z)\|_2 + \|q_m(z')\|_2 + \|q_m(\bar{z})\|_2 + \|q_m(z')\|_2) L_u \rho(z, \bar{z}) \\
&\leq (1 + 1 + 1 + 1) L_u \rho(z, \bar{z}) = 4L_u \rho(z, \bar{z}).
\end{aligned}$$

Averaging over m gives the first-coordinate Lipschitz bound:

$$|c_{\text{dir}}^{(2)}(z, z') - c_{\text{dir}}^{(2)}(\bar{z}, z')| \leq \frac{1}{M} \sum_{m=1}^M 4L_u \rho(z, \bar{z}) = 4L_u \rho(z, \bar{z}).$$

The second coordinate is identical. Thus $L_x = L_y = 4L_u$, and the stability bound follows from Theorem 1. For the range,

$$\begin{aligned} c_{\text{dir}}^{(2)}(z, z') &= \frac{1}{M} \sum_{m=1}^M \|q_m(z) - q_m(z')\|_2^2 \\ &\leq \frac{1}{M} \sum_{m=1}^M (\|q_m(z)\|_2 + \|q_m(z')\|_2)^2 \\ &\leq \frac{1}{M} \sum_{m=1}^M 4 = 4. \end{aligned}$$

Hence $B_{\text{dir}}^{(2)} = 4$ and $\overline{K}_{\text{dir}}^{(2)} = 4L_u/4 = L_u$. $\square$

Interpretation. Squaring the directional discrepancy increases the normalized stability constant from $L_u/2$ to L_u .

Corollary 8 (Stability under composite unsquared flow cost). *Consider $c_{\text{comp}}(z, z') = \lambda_x c_{\text{Euc}}(z, z') + \lambda_u c_{\text{dir}}(z, z')$ with $\lambda_x, \lambda_u \geq 0$. Then $|W_{c_{\text{comp}}}(\mu, \nu) - W_{c_{\text{comp}}}(\tilde{\mu}, \tilde{\nu})| \leq (\lambda_x L_\phi + \lambda_u L_u) \delta_\rho$. Hence $K_{\text{comp}} = \lambda_x L_\phi + \lambda_u L_u$. Moreover, $B_{\text{comp}} = 2(\lambda_x R_\phi + \lambda_u)$ and $\overline{K}_{\text{comp}} = (\lambda_x L_\phi + \lambda_u L_u)/(2(\lambda_x R_\phi + \lambda_u))$.*

Proof. For the first coordinate, using nonnegativity of λ_x, λ_u and the coordinate-wise bounds already proved,

$$\begin{aligned} &|c_{\text{comp}}(z, z') - c_{\text{comp}}(\bar{z}, z')| \\ &= |\lambda_x [c_{\text{Euc}}(z, z') - c_{\text{Euc}}(\bar{z}, z')] + \lambda_u [c_{\text{dir}}(z, z') - c_{\text{dir}}(\bar{z}, z')]| \\ &\leq \lambda_x |c_{\text{Euc}}(z, z') - c_{\text{Euc}}(\bar{z}, z')| + \lambda_u |c_{\text{dir}}(z, z') - c_{\text{dir}}(\bar{z}, z')| \\ &\leq \lambda_x L_\phi \rho(z, \bar{z}) + \lambda_u L_u \rho(z, \bar{z}) \\ &= (\lambda_x L_\phi + \lambda_u L_u) \rho(z, \bar{z}). \end{aligned}$$

The same argument in the second coordinate gives the same constant. Hence

$$L_x = L_y = \lambda_x L_\phi + \lambda_u L_u,$$

and Theorem 1 gives the claimed stability bound. For the range,

$$\begin{aligned} c_{\text{comp}}(z, z') &= \lambda_x c_{\text{Euc}}(z, z') + \lambda_u c_{\text{dir}}(z, z') \\ &\leq \lambda_x (2R_\phi) + \lambda_u (2) = 2(\lambda_x R_\phi + \lambda_u). \end{aligned}$$

Thus

$$\overline{K}_{\text{comp}} = \frac{\lambda_x L_\phi + \lambda_u L_u}{2(\lambda_x R_\phi + \lambda_u)}.$$

$\square$

Interpretation. The composite unsquared cost interpolates between positional stability and directional stability.

Corollary 9 (Stability under composite squared flow cost). *Consider $c_{\text{flow}}(z, z') = \lambda_x c_{\text{sq}}(z, z') + \lambda_u c_{\text{dir}}^{(2)}(z, z')$ with $\lambda_x, \lambda_u \geq 0$. Then $|W_{c_{\text{flow}}}(\mu, \nu) - W_{c_{\text{flow}}}(\tilde{\mu}, \tilde{\nu})| \leq (4\lambda_x R_\phi L_\phi + 4\lambda_u L_u) \delta_\rho$. Hence $K_{\text{flow}} = 4\lambda_x R_\phi L_\phi + 4\lambda_u L_u$. Moreover, $B_{\text{flow}} = 4(\lambda_x R_\phi^2 + \lambda_u)$ and $\overline{K}_{\text{flow}} = (\lambda_x R_\phi L_\phi + \lambda_u L_u)/(\lambda_x R_\phi^2 + \lambda_u)$.*

Proof. For the first coordinate,

$$\begin{aligned} &|c_{\text{flow}}(z, z') - c_{\text{flow}}(\bar{z}, z')| \\ &= |\lambda_x [c_{\text{sq}}(z, z') - c_{\text{sq}}(\bar{z}, z')] + \lambda_u [c_{\text{dir}}^{(2)}(z, z') - c_{\text{dir}}^{(2)}(\bar{z}, z')]| \\ &\leq \lambda_x |c_{\text{sq}}(z, z') - c_{\text{sq}}(\bar{z}, z')| + \lambda_u |c_{\text{dir}}^{(2)}(z, z') - c_{\text{dir}}^{(2)}(\bar{z}, z')| \\ &\leq \lambda_x (4R_\phi L_\phi) \rho(z, \bar{z}) + \lambda_u (4L_u) \rho(z, \bar{z}) \\ &= (4\lambda_x R_\phi L_\phi + 4\lambda_u L_u) \rho(z, \bar{z}). \end{aligned}$$

Table 7: Raw and normalized worst-case stability constants for common ground costs on the shared reference space $\mathcal{R}$. The raw constant K_c is scale-dependent. The normalized constant $\bar{K}_c = K_c/B_c$ removes the trivial cost-scaling effect using an on-support range bound B_c .

Cost	Raw constant K_c	Range bound B_c	Normalized constant $\bar{K}_c$
Euclidean	L_ϕ	$2R_\phi$	$\frac{L_\phi}{2R_\phi}$
Squared Euclidean	$4R_\phi L_\phi$	$4R_\phi^2$	$\frac{L_\phi}{R_\phi}$
Mahalanobis	$\ A\ _{\text{op}} L_\phi$	$2R_\phi \ A\ _{\text{op}}$	$\frac{L_\phi}{2R_\phi}$
Squared Mahalanobis	$4R_\phi \ A\ _{\text{op}}^2 L_\phi$	$4R_\phi^2 \ A\ _{\text{op}}^2$	$\frac{L_\phi}{R_\phi}$
Cosine	$\frac{2L_\phi}{r_\phi}$	2	$\frac{L_\phi}{r_\phi}$
Pure flow-direction	L_u	2	$\frac{L_u}{2}$
Squared flow-direction	$4L_u$	4	L_u
Composite unsquared flow	$\lambda_x L_\phi + \lambda_u L_u$	$2(\lambda_x R_\phi + \lambda_u)$	$\frac{\lambda_x L_\phi + \lambda_u L_u}{2(\lambda_x R_\phi + \lambda_u)}$
Composite squared flow	$4\lambda_x R_\phi L_\phi + 4\lambda_u L_u$	$4(\lambda_x R_\phi^2 + \lambda_u)$	$\frac{\lambda_x R_\phi L_\phi + \lambda_u L_u}{\lambda_x R_\phi^2 + \lambda_u}$

The second coordinate is identical. Hence $L_x = L_y = 4\lambda_x R_\phi L_\phi + 4\lambda_u L_u$, and the stability bound follows from Theorem 1. For the range,

$$\begin{aligned} c_{\text{flow}}(z, z') &= \lambda_x c_{\text{sq}}(z, z') + \lambda_u c_{\text{dir}}^{(2)}(z, z') \\ &\leq \lambda_x (4R_\phi^2) + \lambda_u (4) = 4(\lambda_x R_\phi^2 + \lambda_u). \end{aligned}$$

Therefore,

$$\bar{K}_{\text{flow}} = \frac{4\lambda_x R_\phi L_\phi + 4\lambda_u L_u}{4(\lambda_x R_\phi^2 + \lambda_u)} = \frac{\lambda_x R_\phi L_\phi + \lambda_u L_u}{\lambda_x R_\phi^2 + \lambda_u}.$$

□

Interpretation. The squared flow cost matches squared-distance geometries, but it has larger raw stability constants than the corresponding unsquared composite cost.

Scale-invariant comparison for feature matching. Even normalized constants do not fully determine which cost is better for feature matching, because the matching guarantee depends on both the stability constant and the Wasserstein matching margin. For a ground cost c , define the margin-normalized robustness ratio $\chi_i(c) := K_c \delta_\rho / \Delta_i^c$, where Δ_i^c is the Wasserstein matching margin under c . The nearest-neighbor match is certified whenever $\chi_i(c) < 1/2$. This ratio is invariant to positive cost rescaling: if $c' = \alpha c$ with $\alpha > 0$, then

$$K_{c'} = \alpha K_c, \quad \Delta_i^{c'} = \alpha \Delta_i^c, \quad \chi_i(c') = \frac{\alpha K_c \delta_\rho}{\alpha \Delta_i^c} = \chi_i(c).$$

Therefore, for feature matching, the meaningful comparison is not the raw constant K_c , but the margin-normalized ratio $\chi_i(c)$.

Euclidean versus pure flow-direction stability. Using normalized constants, pure flow-direction cost is more stable than Euclidean positional cost only when $\bar{K}_{\text{dir}} < \bar{K}_{\text{Euc}}$. Since

$$\bar{K}_{\text{dir}} = \frac{L_u}{2}, \quad \bar{K}_{\text{Euc}} = \frac{L_\phi}{2R_\phi},$$

this condition is equivalent to $L_u < L_\phi / R_\phi$. Thus, pure flow-direction cost is more stable only when the normalized velocity field varies more smoothly than the representation geometry after accounting for the representation scale R_ϕ .

E.5 Stability Under Ground-Cost Perturbation

Theorem E.5 (Stability under ground-cost perturbation). *Let $\mu, \nu \in \mathcal{P}(\mathcal{R})$ be projected feature distributions, and let $c, \tilde{c} : \mathcal{R} \times \mathcal{R} \rightarrow \mathbb{R}_{\geq 0}$ be two ground costs. Define the relevant transport support by $\mathcal{S}_{\mu, \nu} := \text{supp}(\mu) \times \text{supp}(\nu)$. Suppose c and $\tilde{c}$ are uniformly close on this support:*

$$\varepsilon_c := \sup_{(z, z') \in \mathcal{S}_{\mu, \nu}} |c(z, z') - \tilde{c}(z, z')| < \infty.$$

Then $|W_c(\mu, \nu) - W_{\tilde{c}}(\mu, \nu)| \leq \varepsilon_c$.

Proof. For any coupling $\pi \in \Pi(\mu, \nu)$, the support of π is contained in $\text{supp}(\mu) \times \text{supp}(\nu)$. Hence,

$$|\mathbb{E}_\pi[c(z, z')] - \mathbb{E}_\pi[\tilde{c}(z, z')]| \leq \mathbb{E}_\pi[|c(z, z') - \tilde{c}(z, z')|] \leq \varepsilon_c.$$

Equivalently, for every $\pi \in \Pi(\mu, \nu)$,

$$\mathbb{E}_\pi[c(z, z')] \leq \mathbb{E}_\pi[\tilde{c}(z, z')] + \varepsilon_c, \quad \mathbb{E}_\pi[\tilde{c}(z, z')] \leq \mathbb{E}_\pi[c(z, z')] + \varepsilon_c.$$

We first show

$$W_c(\mu, \nu) - W_{\tilde{c}}(\mu, \nu) \leq \varepsilon_c.$$

Let $\eta > 0$, and choose an η -optimal coupling $\tilde{\pi}_\eta \in \Pi(\mu, \nu)$ for $W_{\tilde{c}}(\mu, \nu)$, so that

$$\mathbb{E}_{\tilde{\pi}_\eta}[\tilde{c}(z, z')] \leq W_{\tilde{c}}(\mu, \nu) + \eta.$$

Since $\tilde{\pi}_\eta$ is feasible for $W_c(\mu, \nu)$,

$$\begin{aligned} W_c(\mu, \nu) &\leq \mathbb{E}_{\tilde{\pi}_\eta}[c(z, z')] \\ &\leq \mathbb{E}_{\tilde{\pi}_\eta}[\tilde{c}(z, z')] + \varepsilon_c \\ &\leq W_{\tilde{c}}(\mu, \nu) + \eta + \varepsilon_c. \end{aligned}$$

Taking $\eta \downarrow 0$ gives

$$W_c(\mu, \nu) - W_{\tilde{c}}(\mu, \nu) \leq \varepsilon_c.$$

The reverse inequality follows identically. Let $\pi_\eta \in \Pi(\mu, \nu)$ be an η -optimal coupling for $W_c(\mu, \nu)$. Then

$$\begin{aligned} W_{\tilde{c}}(\mu, \nu) &\leq \mathbb{E}_{\pi_\eta}[\tilde{c}(z, z')] \\ &\leq \mathbb{E}_{\pi_\eta}[c(z, z')] + \varepsilon_c \\ &\leq W_c(\mu, \nu) + \eta + \varepsilon_c. \end{aligned}$$

Taking $\eta \downarrow 0$ yields

$$W_{\tilde{c}}(\mu, \nu) - W_c(\mu, \nu) \leq \varepsilon_c.$$

Combining the two one-sided bounds,

$$|W_c(\mu, \nu) - W_{\tilde{c}}(\mu, \nu)| \leq \varepsilon_c.$$

□

Interpretation. This theorem isolates the effect of changing the ground cost while keeping the projected feature distributions fixed. In our method, the distributions $\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}$ and $\hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}$ live in the same shared reference space $\mathcal{R}$, but the cost used to compare their support points may be Euclidean, flow-aware, learned, or approximated for computational reasons. The theorem says that if the approximate cost $\tilde{c}$ is uniformly close to the intended cost c on the actually transported support, then the resulting Wasserstein score changes by at most the same amount. Thus, learned or approximate ground costs are safe to use when their error is controlled on the empirical support of the projected distributions.

E.6 Matching Recovery Under Wasserstein Margin

Definitions. Fix a target feature $i \in \mathcal{F}^{(A)}$ and a source feature set $\mathcal{F}^{(B)}$. For each source feature $j \in \mathcal{F}^{(B)}$, define the oracle Wasserstein score in the shared reference space $\mathcal{R}$ by $D_{\mathcal{R}}(i, j) = W_c(\mu_i^{(A \rightarrow \mathcal{R})}, \nu_j^{(B \rightarrow \mathcal{R})})$, where $\mu_i^{(A \rightarrow \mathcal{R})}$ and $\nu_j^{(B \rightarrow \mathcal{R})}$ denote the population projected feature distributions of the target and source features. The empirical score computed from finite projected activation distributions is

$$\hat{D}_{\mathcal{R}}(i, j) = W_c(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}).$$

Let $j_i^* = \arg \min_{j \in \mathcal{F}^{(B)}} D_{\mathcal{R}}(i, j)$ be the oracle nearest-neighbor match of target feature i . We assume that this minimizer is unique. The Wasserstein matching margin is

$$\Delta_i = \min_{j \neq j_i^*} [D_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j_i^*)].$$

Thus, Δ_i measures how much closer the oracle match j_i^* is than the nearest competing source feature. Finally, define the uniform empirical score perturbation by

$$\epsilon_i = \sup_{j \in \mathcal{F}^{(B)}} |\hat{D}_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j)|.$$

This quantity upper-bounds the largest score error over all candidate source features for the fixed target feature i .

Theorem 2 (Matching recovery under Wasserstein margin). Fix $i \in \mathcal{F}^{(A)}$ and suppose j_i^* is unique with $\Delta_i > 0$. If

$$\sup_{j \in \mathcal{F}^{(B)}} |\hat{D}_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j)| \leq \epsilon_i \quad \text{and} \quad \Delta_i > 2\epsilon_i,$$

then

$$\arg \min_{j \in \mathcal{F}^{(B)}} \hat{D}_{\mathcal{R}}(i, j) = j_i^*.$$

Proof. Because j_i^* is the unique oracle minimizer, every competing source $j \neq j_i^*$ satisfies

$$D_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j_i^*) \geq \Delta_i.$$

The uniform perturbation condition gives, for every $j \in \mathcal{F}^{(B)}$,

$$-\epsilon_i \leq \hat{D}_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j) \leq \epsilon_i.$$

Equivalently,

$$\hat{D}_{\mathcal{R}}(i, j) \geq D_{\mathcal{R}}(i, j) - \epsilon_i, \quad \hat{D}_{\mathcal{R}}(i, j) \leq D_{\mathcal{R}}(i, j) + \epsilon_i.$$

Now fix any competing source $j \neq j_i^*$. We compare its empirical score with the empirical score of the oracle match by adding and subtracting the oracle scores:

$$\begin{aligned} & \hat{D}_{\mathcal{R}}(i, j) - \hat{D}_{\mathcal{R}}(i, j_i^*) \\ &= [\hat{D}_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j)] + [D_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j_i^*)] + [D_{\mathcal{R}}(i, j_i^*) - \hat{D}_{\mathcal{R}}(i, j_i^*)] \\ &\geq [-\epsilon_i] + \Delta_i + [-\epsilon_i] \\ &= \Delta_i - 2\epsilon_i. \end{aligned}$$

If $\Delta_i > 2\epsilon_i$, then

$$\hat{D}_{\mathcal{R}}(i, j) - \hat{D}_{\mathcal{R}}(i, j_i^*) > 0, \quad \forall j \neq j_i^*.$$

Hence every competing source has strictly larger empirical score than j_i^* :

$$\hat{D}_{\mathcal{R}}(i, j) > \hat{D}_{\mathcal{R}}(i, j_i^*), \quad \forall j \neq j_i^*.$$

Therefore j_i^* remains the unique empirical nearest neighbor:

$$\arg \min_{j \in \mathcal{F}^{(B)}} \hat{D}_{\mathcal{R}}(i, j) = j_i^*.$$

□

Interpretation. This theorem converts score stability into decision stability. The margin Δ_i measures how much the oracle best match is separated from the nearest competing source. The perturbation ϵ_i controls how much any empirical score can deviate from its oracle value. The factor 2 appears because, in the worst case, the empirical score of the true match can increase by ϵ_i while the score of a competitor can decrease by ϵ_i . Thus the oracle gap can shrink by at most $2\epsilon_i$.

Combined distribution and cost perturbations. The theorem above is stated in terms of an abstract score perturbation ϵ_i . We now derive such a perturbation bound when both the projected feature distributions and the ground cost are perturbed. Fix a target feature $i \in \mathcal{F}^{(A)}$ and a source feature $j \in \mathcal{F}^{(B)}$. The oracle score is

$$D_{\mathcal{R}}(i, j) = W_c\left(\mu_i^{(A \rightarrow \mathcal{R})}, \nu_j^{(B \rightarrow \mathcal{R})}\right).$$

Suppose the empirical computation uses projected empirical distributions and an approximate ground cost $\tilde{c} : \mathcal{R} \times \mathcal{R} \rightarrow \mathbb{R}_{\geq 0}$. Define the perturbed empirical score by

$$\widehat{D}_{\mathcal{R}}^{\tilde{c}}(i, j) = W_{\tilde{c}}\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right).$$

The distribution perturbation error for pair (i, j) is

$$\delta_{i,j} := L_x W_{\rho}\left(\mu_i^{(A \rightarrow \mathcal{R})}, \hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right) + L_y W_{\rho}\left(\nu_j^{(B \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right),$$

where L_x, L_y are the coordinate-wise Lipschitz constants of the reference ground cost c on the relevant support. The cost perturbation error is

$$\varepsilon_{c,i,j} := \sup_{(z,z') \in \text{supp}(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}) \times \text{supp}(\hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})})} |c(z, z') - \tilde{c}(z, z')|.$$

This is the maximum pointwise discrepancy between the reference cost and the approximate cost on the empirical supports actually used by the OT problem. Define the worst-case combined perturbation for target feature i by

$$\bar{\epsilon}_i := \max_{j \in \mathcal{F}^{(B)}} [\delta_{i,j} + \varepsilon_{c,i,j}].$$

Corollary 10 (Stable matching under distribution and cost perturbations). *Let j_i^* be the unique oracle nearest-neighbor match under $D_{\mathcal{R}}(i, j)$, and let Δ_i be its Wasserstein matching margin. If*

$$\Delta_i > 2\bar{\epsilon}_i,$$

then nearest-neighbor matching with the perturbed empirical scores recovers the oracle match:

$$\arg \min_{j \in \mathcal{F}^{(B)}} \widehat{D}_{\mathcal{R}}^{\tilde{c}}(i, j) = j_i^*.$$

Proof. Fix any source feature $j \in \mathcal{F}^{(B)}$. We decompose the total score error into a distribution perturbation term and a cost perturbation term by adding and subtracting the intermediate score computed on empirical distributions with the reference cost c :

$$\begin{aligned} & \left| D_{\mathcal{R}}(i, j) - \widehat{D}_{\mathcal{R}}^{\tilde{c}}(i, j) \right| \\ &= \left| W_c\left(\mu_i^{(A \rightarrow \mathcal{R})}, \nu_j^{(B \rightarrow \mathcal{R})}\right) - W_{\tilde{c}}\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right) \right| \\ &\leq \left| W_c\left(\mu_i^{(A \rightarrow \mathcal{R})}, \nu_j^{(B \rightarrow \mathcal{R})}\right) - W_c\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right) \right| \\ &\quad + \left| W_c\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right) - W_{\tilde{c}}\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right) \right|. \end{aligned}$$

By Theorem 1, the first term is bounded by

$$\begin{aligned} & \left| W_c\left(\mu_i^{(A \rightarrow \mathcal{R})}, \nu_j^{(B \rightarrow \mathcal{R})}\right) - W_c\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right) \right| \\ &\leq L_x W_{\rho}\left(\mu_i^{(A \rightarrow \mathcal{R})}, \hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right) + L_y W_{\rho}\left(\nu_j^{(B \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right) \\ &= \delta_{i,j}. \end{aligned}$$

It remains to bound the cost perturbation term. Let

$$\hat{\mu} = \hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \quad \hat{\nu} = \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}$$

inside this paragraph only. By definition of $\varepsilon_{c,i,j}$, for every coupling $\pi \in \Pi(\hat{\mu}, \hat{\nu})$,

$$\begin{aligned} |\mathbb{E}_\pi[c(z, z')] - \mathbb{E}_\pi[\tilde{c}(z, z')]| &= \left| \int_{\mathcal{R} \times \mathcal{R}} [c(z, z') - \tilde{c}(z, z')] d\pi(z, z') \right| \\ &\leq \int_{\mathcal{R} \times \mathcal{R}} |c(z, z') - \tilde{c}(z, z')| d\pi(z, z') \\ &\leq \varepsilon_{c,i,j}. \end{aligned}$$

Hence, for every $\pi \in \Pi(\hat{\mu}, \hat{\nu})$,

$$\mathbb{E}_\pi[c(z, z')] \leq \mathbb{E}_\pi[\tilde{c}(z, z')] + \varepsilon_{c,i,j}, \quad \mathbb{E}_\pi[\tilde{c}(z, z')] \leq \mathbb{E}_\pi[c(z, z')] + \varepsilon_{c,i,j}.$$

Taking the infimum of the first inequality over $\pi \in \Pi(\hat{\mu}, \hat{\nu})$ gives

$$\begin{aligned} W_c(\hat{\mu}, \hat{\nu}) &= \inf_{\pi \in \Pi(\hat{\mu}, \hat{\nu})} \mathbb{E}_\pi[c(z, z')] \\ &\leq \inf_{\pi \in \Pi(\hat{\mu}, \hat{\nu})} \{\mathbb{E}_\pi[\tilde{c}(z, z')] + \varepsilon_{c,i,j}\} \\ &= W_{\tilde{c}}(\hat{\mu}, \hat{\nu}) + \varepsilon_{c,i,j}. \end{aligned}$$

Taking the infimum of the second inequality gives the reverse direction:

$$\begin{aligned} W_{\tilde{c}}(\hat{\mu}, \hat{\nu}) &= \inf_{\pi \in \Pi(\hat{\mu}, \hat{\nu})} \mathbb{E}_\pi[\tilde{c}(z, z')] \\ &\leq \inf_{\pi \in \Pi(\hat{\mu}, \hat{\nu})} \{\mathbb{E}_\pi[c(z, z')] + \varepsilon_{c,i,j}\} \\ &= W_c(\hat{\mu}, \hat{\nu}) + \varepsilon_{c,i,j}. \end{aligned}$$

Combining the two one-sided inequalities yields

$$|W_c(\hat{\mu}, \hat{\nu}) - W_{\tilde{c}}(\hat{\mu}, \hat{\nu})| \leq \varepsilon_{c,i,j}.$$

Therefore,

$$\begin{aligned} \left| D_{\mathcal{R}}(i, j) - \widehat{D}_{\mathcal{R}}^{\tilde{c}}(i, j) \right| &\leq \delta_{i,j} + \varepsilon_{c,i,j} \\ &\leq \bar{\varepsilon}_i, \quad \forall j \in \mathcal{F}^{(B)}. \end{aligned}$$

Thus the perturbed empirical scores satisfy the uniform perturbation condition of Theorem 2 with $\varepsilon_i = \bar{\varepsilon}_i$. Since $\Delta_i > 2\bar{\varepsilon}_i$, applying Theorem 2 gives

$$\arg \min_{j \in \mathcal{F}^{(B)}} \widehat{D}_{\mathcal{R}}^{\tilde{c}}(i, j) = j_i^*.$$

□

Interpretation. This corollary separates the two perturbation sources that appear in practice. The first source is distributional: finite top- K sampling, corpus variation, or SAE retraining may change the projected feature distributions. This contributes the term $\delta_{i,j}$. The second source is geometric: the ground cost may be approximate, learned, or computed from a proxy representation. This contributes the term $\varepsilon_{c,i,j}$. The result shows that these errors add at the score level. Matching remains stable whenever the oracle Wasserstein margin Δ_i is larger than twice the worst-case combined error over source features.

E.7 Matching recovery under Wasserstein margin

Definitions. Fix a target feature $i \in \mathcal{F}^{(A)}$ and a source feature set $\mathcal{F}^{(B)}$. For each source feature $j \in \mathcal{F}^{(B)}$, define the oracle Wasserstein score in the shared reference space $\mathcal{R}$ by

$$D_{\mathcal{R}}(i, j) = W_c\left(\mu_i^{(A \rightarrow \mathcal{R})}, \nu_j^{(B \rightarrow \mathcal{R})}\right),$$

and define the empirical Wasserstein score by

$$\widehat{D}_{\mathcal{R}}(i, j) = W_c\left(\widehat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \widehat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right).$$

Let

$$j_i^* = \arg \min_{j \in \mathcal{F}^{(B)}} D_{\mathcal{R}}(i, j)$$

be the oracle nearest-neighbor match of target feature i , and suppose this minimizer is unique. Define the Wasserstein matching margin by

$$\Delta_i = \min_{j \neq j_i^*} [D_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j_i^*)].$$

Finally, define the uniform empirical score perturbation by

$$\epsilon_i = \sup_{j \in \mathcal{F}^{(B)}} \left| \widehat{D}_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j) \right|.$$

Theorem 2 (Matching recovery under Wasserstein margin). *Fix $i \in \mathcal{F}^{(A)}$ and suppose j_i^* is unique with $\Delta_i > 0$. If $\sup_{j \in \mathcal{F}^{(B)}} |\widehat{D}_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j)| \leq \epsilon_i$ and $\Delta_i > 2\epsilon_i$, then $\arg \min_{j \in \mathcal{F}^{(B)}} \widehat{D}_{\mathcal{R}}(i, j) = j_i^*$.*

Proof. Since j_i^* is the unique minimizer of the oracle scores, for every $j \neq j_i^*$, $D_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j_i^*) \geq \Delta_i$. By the uniform perturbation bound, for every $j \in \mathcal{F}^{(B)}$,

$$\left| \widehat{D}_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j) \right| \leq \epsilon_i.$$

Therefore, for every $j \neq j_i^*$,

$$\widehat{D}_{\mathcal{R}}(i, j) \geq D_{\mathcal{R}}(i, j) - \epsilon_i, \quad \widehat{D}_{\mathcal{R}}(i, j_i^*) \leq D_{\mathcal{R}}(i, j_i^*) + \epsilon_i.$$

Subtracting the two inequalities gives

$$\begin{aligned} \widehat{D}_{\mathcal{R}}(i, j) - \widehat{D}_{\mathcal{R}}(i, j_i^*) &\geq (D_{\mathcal{R}}(i, j) - \epsilon_i) - (D_{\mathcal{R}}(i, j_i^*) + \epsilon_i) \\ &= D_{\mathcal{R}}(i, j) - D_{\mathcal{R}}(i, j_i^*) - 2\epsilon_i \\ &\geq \Delta_i - 2\epsilon_i. \end{aligned}$$

If $\Delta_i > 2\epsilon_i$, then

$$\widehat{D}_{\mathcal{R}}(i, j) > \widehat{D}_{\mathcal{R}}(i, j_i^*), \quad \forall j \neq j_i^*.$$

Thus j_i^* is the unique minimizer of the empirical scores:

$$\arg \min_{j \in \mathcal{F}^{(B)}} \widehat{D}_{\mathcal{R}}(i, j) = j_i^*.$$

□

Interpretation. This theorem converts score stability into decision stability. The Wasserstein stability theorem controls how much the scores can move; the margin condition ensures that this movement is not large enough to change the nearest-neighbor decision. The factor 2 appears because the oracle best match may become worse by at most ϵ_i , while a competing match may become better by at most ϵ_i .

Combined distribution and cost perturbations. The matching recovery theorem is stated in terms of an abstract score perturbation ϵ_i . We now show how such a perturbation bound follows when both the projected feature distributions and the ground cost are perturbed.

Fix a target feature $i \in \mathcal{F}^{(A)}$ and a source feature $j \in \mathcal{F}^{(B)}$. Recall the oracle Wasserstein score

$$D_{\mathcal{R}}(i, j) = W_c\left(\mu_i^{(A \rightarrow \mathcal{R})}, \nu_j^{(B \rightarrow \mathcal{R})}\right).$$

When both the projected distributions and the ground cost are perturbed, define the approximate empirical score

$$\widehat{D}_{\mathcal{R}}^{\tilde{c}}(i, j) = W_{\tilde{c}}\left(\widehat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \widehat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right).$$

Here, c is the reference ground cost, while $\tilde{c}$ is an approximate or learned ground cost. For each pair (i, j) , define the projected-distribution perturbation error

$$\delta_{i,j} := L_x W_\rho\left(\mu_i^{(A \rightarrow \mathcal{R})}, \hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right) + L_y W_\rho\left(\nu_j^{(B \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right),$$

where L_x and L_y are the coordinate-wise Lipschitz constants of c on the relevant support. Define the ground-cost perturbation error by

$$\varepsilon_{c,i,j} := \sup_{(x,y) \in \text{supp}(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}) \times \text{supp}(\hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})})} |c(x, y) - \tilde{c}(x, y)|.$$

Finally, define the worst-case combined perturbation for target feature i by

$$\bar{\epsilon}_i := \max_{j \in \mathcal{F}^{(B)}} [\delta_{i,j} + \varepsilon_{c,i,j}].$$

Corollary 11 (Stable matching under distribution and cost perturbations). *Let j_i^* be the unique oracle nearest-neighbor match under $D_{\mathcal{R}}(i, j)$, and let Δ_i be its Wasserstein matching margin. If*

$$\Delta_i > 2\bar{\epsilon}_i,$$

then nearest-neighbor matching with the perturbed empirical scores recovers the oracle match:

$$\arg \min_{j \in \mathcal{F}^{(B)}} \hat{D}_{\mathcal{R}}^{\tilde{c}}(i, j) = j_i^*.$$

Proof. For each source feature $j \in \mathcal{F}^{(B)}$, add and subtract the intermediate score that uses the perturbed distributions but the original ground cost: $W_c\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right)$. Then

$$\begin{aligned} & \left| D_{\mathcal{R}}(i, j) - \hat{D}_{\mathcal{R}}^{\tilde{c}}(i, j) \right| \\ &= \left| W_c\left(\mu_i^{(A \rightarrow \mathcal{R})}, \nu_j^{(B \rightarrow \mathcal{R})}\right) - W_{\tilde{c}}\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right) \right| \\ &\leq \left| W_c\left(\mu_i^{(A \rightarrow \mathcal{R})}, \nu_j^{(B \rightarrow \mathcal{R})}\right) - W_c\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right) \right| \\ &\quad + \left| W_c\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right) - W_{\tilde{c}}\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right) \right|. \end{aligned}$$

By Theorem 1, the first term is bounded by

$$L_x W_\rho\left(\mu_i^{(A \rightarrow \mathcal{R})}, \hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right) + L_y W_\rho\left(\nu_j^{(B \rightarrow \mathcal{R})}, \hat{\nu}_{j,T}^{(B \rightarrow \mathcal{R})}\right) = \delta_{i,j}.$$

By Theorem 2, the second term is bounded by $\varepsilon_{c,i,j}$. Therefore,

$$\left| D_{\mathcal{R}}(i, j) - \hat{D}_{\mathcal{R}}^{\tilde{c}}(i, j) \right| \leq \delta_{i,j} + \varepsilon_{c,i,j} \leq \bar{\epsilon}_i, \quad \forall j \in \mathcal{F}^{(B)}.$$

Thus the perturbed empirical scores satisfy the uniform perturbation condition of Theorem 2 with $\epsilon_i = \bar{\epsilon}_i$. Since $\Delta_i > 2\bar{\epsilon}_i$, Theorem 2 gives

$$\arg \min_{j \in \mathcal{F}^{(B)}} \hat{D}_{\mathcal{R}}^{\tilde{c}}(i, j) = j_i^*.$$

□

Interpretation. This corollary combines the two perturbation sources that arise in practice. The projected empirical distributions may differ from the oracle projected distributions because they are estimated from finite top- K activations, and the ground cost may also be approximate, learned, or flow-aware. The result says that these errors add: the total score error is controlled by the distribution perturbation term $\delta_{i,j}$ plus the cost perturbation term $\varepsilon_{c,i,j}$. Matching is stable whenever the oracle Wasserstein margin Δ_i is larger than twice the worst-case combined error. Thus, a match is reliable when the projected distributions are accurate, the cost approximation is accurate on the empirical transport support, and the best source feature is separated from its nearest competitor by a nontrivial margin.

E.8 Voronoi Semantic Cell Recovery

We state the recovery result directly in the shared reference space $(\mathcal{R}, \rho)$, where all projected feature distributions live. This avoids introducing an additional semantic map and makes the grouping analysis align with the method, which assigns circuit nodes using distances between projected feature distributions in $\mathcal{R}$.

Semantic centers and cells. Let $\xi_1, \dots, \xi_m \in \mathcal{R}$ be semantic centers. Here, m is the number of semantic supernodes in the compressed circuit, and ξ_k is the representative center of the k -th semantic supernode. These centers induce a Voronoi partition of the shared reference space:

$$\text{Cell}_k := \{z \in \mathcal{R} : \rho(z, \xi_k) \leq \rho(z, \xi_r), \forall r \in [m]\}.$$

Thus, a point $z \in \mathcal{R}$ belongs to Cell_k if it is at least as close to ξ_k as to every other semantic center.

Population and empirical projected feature distributions. Fix a target feature $i \in \mathcal{F}^{(A)}$. Let $\mu_i^{(A \rightarrow \mathcal{R})} \in \mathcal{P}(\mathcal{R})$ denote the population projected feature distribution, and let $\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})} \in \mathcal{P}(\mathcal{R})$ denote the empirical projected feature distribution constructed from T token positions. In our construction,

$$\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})} = \sum_{t \in \mathcal{I}_{i,T}^{(A),K}} \hat{w}_{i,t}^{(A)} \delta_{z_t^{(\mathcal{R})}}.$$

The target-layer activation function selects the top- K token indices $\mathcal{I}_{i,T}^{(A),K}$ and normalized weights $\hat{w}_{i,t}^{(A)}$, while the support points $z_t^{(\mathcal{R})}$ lie in the shared reference space.

Semantic-center assignment scores. Since a feature is represented by a distribution rather than a single point, we assign it to a semantic center by its expected distance to that center. For any projected distribution $\eta \in \mathcal{P}(\mathcal{R})$, define

$$\mathcal{Q}_k(\eta) := \mathbb{E}_{z \sim \eta} [\rho(z, \xi_k)].$$

Here, $\mathcal{Q}_k(\eta)$ is the average distance from the feature distribution η to the k -th semantic center. The population and empirical assignments of feature i are

$$k_i^* := \arg \min_{k \in [m]} \mathcal{Q}_k(\mu_i^{(A \rightarrow \mathcal{R})}), \quad \hat{k}_{i,T} := \arg \min_{k \in [m]} \mathcal{Q}_k(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}).$$

For the empirical distribution, the score has the finite form

$$\mathcal{Q}_k(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}) = \sum_{t \in \mathcal{I}_{i,T}^{(A),K}} \hat{w}_{i,t}^{(A)} \rho(z_t^{(\mathcal{R})}, \xi_k).$$

Thus, empirical assignment compares the weighted average distance from the feature's selected activation contexts to each semantic center.

Voronoi assignment margin. Assume that the population assignment k_i^* is unique. Define the Voronoi assignment margin by

$$\Gamma_i := \min_{r \neq k_i^*} \left[\mathcal{Q}_r(\mu_i^{(A \rightarrow \mathcal{R})}) - \mathcal{Q}_{k_i^*}(\mu_i^{(A \rightarrow \mathcal{R})}) \right].$$

The condition $\Gamma_i > 0$ means that the population projected distribution of feature i is strictly closer, in expected distance, to its assigned semantic center $\xi_{k_i^*}$ than to every competing center.

Theorem 3 (Voronoi semantic cell recovery). Suppose the population semantic assignment k_i^* is unique with margin $\Gamma_i > 0$. If

$$W_\rho(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})}) < \frac{\Gamma_i}{2},$$

then the empirical semantic assignment recovers the population assignment: $\hat{k}_{i,T} = k_i^*$.

Proof. For each semantic center ξ_k , define the point-to-center distance function $h_k : \mathcal{R} \rightarrow \mathbb{R}_{\geq 0}$ by $h_k(z) = \rho(z, \xi_k)$. The first step is to show that h_k is 1-Lipschitz with respect to ρ . For any $z, z' \in \mathcal{R}$, the reverse triangle inequality gives

$$\begin{aligned} |h_k(z) - h_k(z')| &= |\rho(z, \xi_k) - \rho(z', \xi_k)| \\ &\leq \rho(z, z'). \end{aligned}$$

Thus, moving a point by at most $\rho(z, z')$ can change its distance to any fixed semantic center by at most $\rho(z, z')$.

We next lift this pointwise Lipschitz property to distributions. Let $\eta, \eta' \in \mathcal{P}(\mathcal{R})$ be any two projected feature distributions, and let $\gamma \in \Pi(\eta, \eta')$ be any coupling between them. Because the first marginal of γ is η and the second marginal is η' , we can rewrite the score difference under the same coupling:

$$\begin{aligned} \mathcal{Q}_k(\eta) - \mathcal{Q}_k(\eta') &= \int_{\mathcal{R}} h_k(z) d\eta(z) - \int_{\mathcal{R}} h_k(z') d\eta'(z') \\ &= \int_{\mathcal{R} \times \mathcal{R}} [h_k(z) - h_k(z')] d\gamma(z, z'). \end{aligned}$$

Therefore,

$$\begin{aligned} |\mathcal{Q}_k(\eta) - \mathcal{Q}_k(\eta')| &= \left| \int_{\mathcal{R} \times \mathcal{R}} [h_k(z) - h_k(z')] d\gamma(z, z') \right| \\ &\leq \int_{\mathcal{R} \times \mathcal{R}} |h_k(z) - h_k(z')| d\gamma(z, z') \\ &\leq \int_{\mathcal{R} \times \mathcal{R}} \rho(z, z') d\gamma(z, z'). \end{aligned}$$

Since the above inequality holds for every coupling $\gamma \in \Pi(\eta, \eta')$, taking the infimum over couplings gives

$$|\mathcal{Q}_k(\eta) - \mathcal{Q}_k(\eta')| \leq W_\rho(\eta, \eta').$$

Apply this bound to $\eta = \hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}$ and $\eta' = \mu_i^{(A \rightarrow \mathcal{R})}$. Define the empirical distribution error

$$\varepsilon_i := W_\rho\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})}\right).$$

Then, for every semantic center $k \in [m]$,

$$\left| \mathcal{Q}_k\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right) - \mathcal{Q}_k\left(\mu_i^{(A \rightarrow \mathcal{R})}\right) \right| \leq \varepsilon_i.$$

Now fix any competing center $r \neq k_i^*$. We compare the empirical score of the competitor r with the empirical score of the population winner k_i^* . Add and subtract the corresponding population scores:

$$\begin{aligned} &\mathcal{Q}_r\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right) - \mathcal{Q}_{k_i^*}\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right) \\ &= \left[\mathcal{Q}_r\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right) - \mathcal{Q}_r\left(\mu_i^{(A \rightarrow \mathcal{R})}\right) \right] \\ &\quad + \left[\mathcal{Q}_r\left(\mu_i^{(A \rightarrow \mathcal{R})}\right) - \mathcal{Q}_{k_i^*}\left(\mu_i^{(A \rightarrow \mathcal{R})}\right) \right] \\ &\quad + \left[\mathcal{Q}_{k_i^*}\left(\mu_i^{(A \rightarrow \mathcal{R})}\right) - \mathcal{Q}_{k_i^*}\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right) \right]. \end{aligned}$$

Each of the three terms has a direct interpretation: the first is the empirical perturbation of the competitor score, the second is the population Voronoi gap, and the third is the empirical perturbation of the winning score. By the definition of Γ_i and the uniform score perturbation bound above,

$$\begin{aligned} \mathcal{Q}_r\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right) - \mathcal{Q}_{k_i^*}\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right) &\geq [-\varepsilon_i] + \Gamma_i + [-\varepsilon_i] \\ &= \Gamma_i - 2\varepsilon_i. \end{aligned}$$

If $\varepsilon_i < \Gamma_i/2$, then $\Gamma_i - 2\varepsilon_i > 0$. Hence, for every competing center $r \neq k_i^*$,

$$\mathcal{Q}_r\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right) > \mathcal{Q}_{k_i^*}\left(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}\right).$$

Therefore k_i^* remains the unique minimizer of the empirical assignment scores:

$$\begin{aligned}\hat{k}_{i,T} &= \arg \min_{k \in [m]} \mathcal{Q}_k(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}) \\ &= k_i^*.\end{aligned}$$

This proves the claim. $\square$

Eventual exact recovery. Assume the conditions of Theorem 3 and suppose

$$W_\rho(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})}) \longrightarrow 0 \quad \text{as } T \rightarrow \infty.$$

If $\Gamma_i > 0$, then there exists $T_i < \infty$ such that $\hat{k}_{i,T} = k_i^*$ for all $T \geq T_i$.

Proof. Since the Wasserstein estimation error converges to zero and $\Gamma_i/2 > 0$, there exists $T_i < \infty$ such that, for every $T \geq T_i$,

$$W_\rho(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})}) < \frac{\Gamma_i}{2}.$$

The condition of Theorem 3 is therefore satisfied for all $T \geq T_i$, so

$$\hat{k}_{i,T} = k_i^*, \quad \forall T \geq T_i.$$

$\square$

High-probability Voronoi recovery. Assume that for some rate function $r_T(\delta)$,

$$\mathbb{P}\left(W_\rho(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})}) \leq r_T(\delta)\right) \geq 1 - \delta.$$

If $r_T(\delta) < \Gamma_i/2$, then $\mathbb{P}(\hat{k}_{i,T} = k_i^*) \geq 1 - \delta$.

Proof. Define the high-probability event

$$\mathcal{E}_T := \left\{W_\rho(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})}) \leq r_T(\delta)\right\}.$$

By assumption, $\mathbb{P}(\mathcal{E}_T) \geq 1 - \delta$. On the event $\mathcal{E}_T$, the rate condition gives

$$W_\rho(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})}) \leq r_T(\delta) < \frac{\Gamma_i}{2}.$$

Thus Theorem 3 implies $\hat{k}_{i,T} = k_i^*$ on $\mathcal{E}_T$. Therefore,

$$\begin{aligned}\mathbb{P}(\hat{k}_{i,T} = k_i^*) &\geq \mathbb{P}(\mathcal{E}_T) \\ &\geq 1 - \delta.\end{aligned}$$

$\square$

Interpretation. This theorem shows that semantic grouping is stable when the empirical projected feature distribution remains sufficiently close to the population projected distribution. The margin Γ_i measures how much smaller the assignment score of the correct semantic center is than the score of the nearest competing center. Since each assignment score $\mathcal{Q}_k$ changes by at most $W_\rho(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})})$, the gap between the correct and competing centers can shrink by at most twice this amount. Thus recovery is controlled by the margin-normalized error

$$\frac{W_\rho(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})})}{\Gamma_i}.$$

For circuit compression, this means that large-margin features can be grouped confidently into semantic supernodes, while small-margin features are close to a semantic boundary and should be treated as ambiguous or left uncompressed.

Rates under the reference-space Voronoi recovery theorem. The recovery condition depends on how accurately the empirical projected feature distribution approximates its population counterpart in the shared reference space. Specifically, Theorem 3 shows that semantic-cell recovery is guaranteed whenever

$$W_\rho\left(\widehat{\mu}_{i,n}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})}\right) < \frac{\Gamma_i}{2}.$$

Thus, any high-probability convergence bound of the form

$$\mathbb{P}\left(W_\rho\left(\widehat{\mu}_{i,n}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})}\right) \leq r_n(\delta)\right) \geq 1 - \delta$$

implies recovery with probability at least $1 - \delta$ whenever $r_n(\delta) < \Gamma_i/2$. Here, n denotes the effective number of samples used to construct the empirical projected distribution. If the empirical distribution is built from all sampled contexts, then $n = T$; if it is built only from the selected top- K contexts, then $n = K$ unless an additional top- K population convergence argument is used.

Rate in one dimension. If the projected feature distribution has effective dimension $d = 1$, empirical Wasserstein convergence typically satisfies

$$W_\rho(\widehat{\mu}_{i,T}, \mu_i) = O_p(T^{-1/2}).$$

A high-probability form can be written as $W_\rho(\widehat{\mu}_{i,T}, \mu_i) \leq C_1 \sqrt{\frac{\log(1/\delta)}{T}}$ with probability at least $1 - \delta$. Therefore, Voronoi recovery is guaranteed whenever

$$\begin{aligned} C_1 \sqrt{\frac{\log(1/\delta)}{T}} &< \frac{\Gamma_i}{2} \\ \iff T &> \frac{4C_1^2}{\Gamma_i^2} \log \frac{1}{\delta}. \end{aligned}$$

Thus, for $d = 1$, $T = O\left(\frac{1}{\Gamma_i^2} \log \frac{1}{\delta}\right)$.

Rate in two dimensions. If the projected feature distribution has effective dimension $d = 2$, empirical Wasserstein convergence has the critical logarithmic rate

$$W_\rho(\widehat{\mu}_{i,T}, \mu_i) = O_p\left(\frac{\log T}{\sqrt{T}}\right).$$

A high-probability form can be written as

$$W_\rho(\widehat{\mu}_{i,T}, \mu_i) \leq C_2 \frac{\log T + \log(1/\delta)}{\sqrt{T}}.$$

Thus, semantic-cell recovery is guaranteed whenever

$$\begin{aligned} C_2 \frac{\log T + \log(1/\delta)}{\sqrt{T}} &< \frac{\Gamma_i}{2} \\ \iff \frac{\sqrt{T}}{\log T + \log(1/\delta)} &> \frac{2C_2}{\Gamma_i}. \end{aligned}$$

Equivalently, up to logarithmic factors, the sufficient sample size scales as $T = \widetilde{O}\left(\frac{1}{\Gamma_i^2}\right)$.

Rate in dimension $d > 2$. If the projected feature distribution has effective dimension $d > 2$, empirical Wasserstein convergence typically satisfies $W_\rho(\widehat{\mu}_{i,T}, \mu_i) = O_p(T^{-1/d})$. A high-probability bound may be written as

$$W_\rho(\widehat{\mu}_{i,T}, \mu_i) \leq C_d T^{-1/d} + C_\delta \sqrt{\frac{\log(1/\delta)}{T}}.$$

For $d > 2$, the dominant term is typically $T^{-1/d}$, while the high-probability concentration term scales as $T^{-1/2}$ up to logarithmic factors. Since $T^{-1/2}$ decays faster than $T^{-1/d}$ when $d > 2$, the

Table 8: Typical empirical Wasserstein convergence rates and sufficient sample complexities for Voronoi semantic recovery in the shared reference space. Here Γ_i is the Voronoi assignment margin, $\delta \in (0, 1)$ is the failure probability, and T is the number of samples used to construct the empirical projected distribution.

Setting	Wasserstein rate $r(T)$	Sufficient sample complexity
$d = 1$	$O(T^{-1/2})$	$O\left(\frac{1}{\Gamma_i^2} \log \frac{1}{\delta}\right)$
$d = 2$	$O\left(\frac{\log T}{\sqrt{T}}\right)$	$\tilde{O}\left(\frac{1}{\Gamma_i^2} \log \frac{1}{\delta}\right)$
$d > 2$	$O(T^{-1/d})$	$O\left(\frac{1}{\Gamma_i^d}\right)$
Finite support S	$O\left(D_{\max} \sqrt{\frac{S + \log(1/\delta)}{T}}\right)$	$O\left(\frac{D_{\max}^2}{\Gamma_i^2} (S + \log \frac{1}{\delta})\right)$

dominant sample-complexity scaling is governed by $T^{-1/d}$. Ignoring the lower-order concentration term, recovery is guaranteed whenever

$$\begin{aligned}
C_d T^{-1/d} &< \frac{\Gamma_i}{2} \\
\iff T^{1/d} &> \frac{2C_d}{\Gamma_i} \\
\iff T &> \left(\frac{2C_d}{\Gamma_i}\right)^d.
\end{aligned}$$

Thus, for $d > 2$, $T = O\left[\left(\frac{1}{\Gamma_i}\right)^d\right]$.

Finite-support rate. If the population projected feature distribution is supported on at most S reference-space atoms,

$$\mu_i = \sum_{s=1}^S p_s \delta_{u_s},$$

and the empirical distribution has the same support with empirical weights $\hat{p}_s$, then

$$W_\rho(\hat{\mu}_{i,T}, \mu_i) \leq D_{\max} \|\hat{p} - p\|_1, \quad D_{\max} := \max_{a,b \in [S]} \rho(u_a, u_b).$$

By multinomial concentration,

$$\|\hat{p} - p\|_1 = O_p\left(\sqrt{\frac{S + \log(1/\delta)}{T}}\right).$$

Hence,

$$W_\rho(\hat{\mu}_{i,T}, \mu_i) = O_p\left(D_{\max} \sqrt{\frac{S + \log(1/\delta)}{T}}\right).$$

Voronoi recovery is guaranteed if

$$\begin{aligned}
D_{\max} \sqrt{\frac{S + \log(1/\delta)}{T}} &< \frac{\Gamma_i}{2} \\
\iff T &> \frac{4D_{\max}^2}{\Gamma_i^2} \left(S + \log \frac{1}{\delta}\right).
\end{aligned}$$

Thus the finite-support sample complexity scales as $T = O\left(\frac{D_{\max}^2}{\Gamma_i^2} [S + \log \frac{1}{\delta}]\right)$.

E.9 Synthetic Verification of Voronoi Semantic-Cell Recovery

We provide a synthetic experiment to verify the mechanism of Theorem 3. The goal is not to evaluate feature matching performance, but to isolate the geometric statement of the theorem: if the empirical projected feature distribution remains within the Wasserstein stability radius of the population projected distribution, then the semantic-cell assignment is preserved.

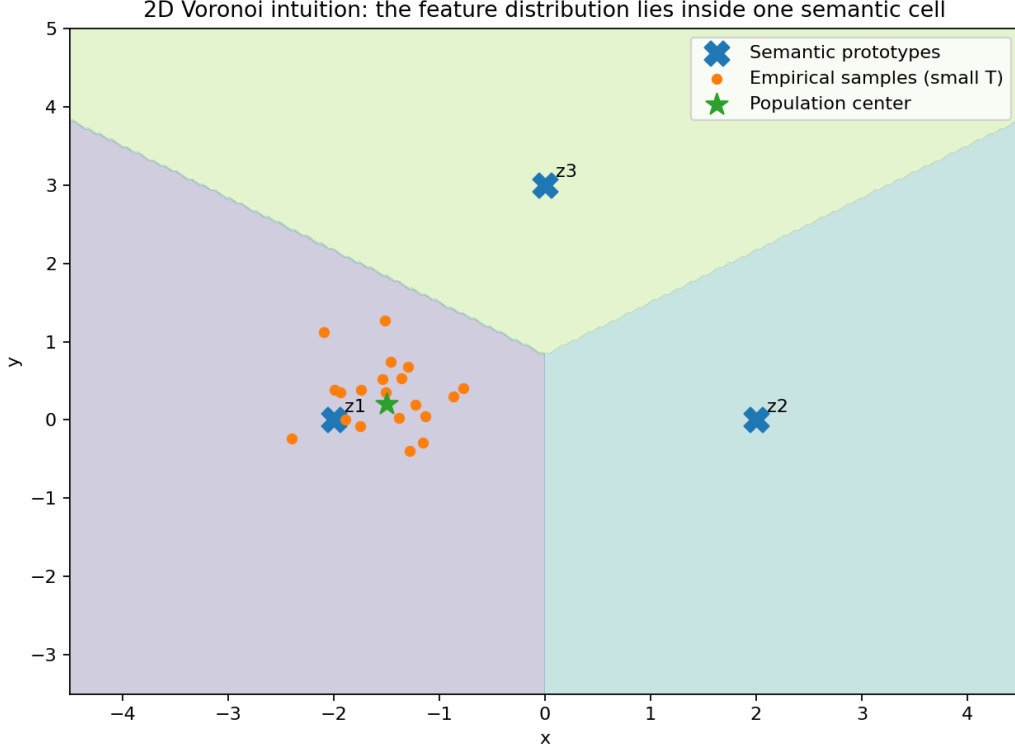


Figure 12: **Voronoi-based semantic cell intuition.** We visualize a synthetic semantic representation space with three prototypes z_1, z_2, z_3 , which induce a Voronoi partition of the space. The orange points represent empirical samples from a feature distribution, while the star denotes the population center. In this example, the distribution lies inside the Voronoi cell of z_1 , meaning that its semantic assignment is stable as long as sampling or representation perturbations do not move the empirical distribution across the decision boundary. This figure illustrates the geometric intuition behind the Voronoi recovery theorem: if a feature distribution is separated from competing cells by a positive margin, then sufficiently accurate empirical estimates recover the same semantic cell.

Setup. We consider the shared reference space $\mathcal{R} = \mathbb{R}$ with base metric $\rho(z, z') = |z - z'|$. The population projected feature distribution is $\mu = \mathcal{N}(m, \sigma^2)$ with $m = -1$ and $\sigma = 0.6$. We use two semantic centers $\xi_1 = -2$ and $\xi_2 = 2$. For any projected distribution $\eta \in \mathcal{P}(\mathcal{R})$, the assignment score to center ξ_k is

$$\mathcal{Q}_k(\eta) = \mathbb{E}_{z \sim \eta} [\rho(z, \xi_k)].$$

The population semantic assignment is therefore

$$k^* = \arg \min_{k \in \{1, 2\}} \mathcal{Q}_k(\mu), \quad \Gamma = \mathcal{Q}_2(\mu) - \mathcal{Q}_1(\mu),$$

where $\Gamma > 0$ means that the population distribution is assigned to ξ_1 with positive Voronoi margin. Since the experiment is one-dimensional, the population scores can be computed in closed form using

$$\mathcal{Q}_k(\mu) = \mathbb{E}_{Z \sim \mathcal{N}(m, \sigma^2)} |Z - \xi_k|.$$

For each sample size T , we draw samples $z_1, \dots, z_T \sim \mu$ and construct the empirical distribution $\hat{\mu}_T = T^{-1} \sum_{t=1}^T \delta_{z_t}$. The empirical assignment is

$$\hat{k}_T = \arg \min_{k \in \{1, 2\}} \mathcal{Q}_k(\hat{\mu}_T), \quad \mathcal{Q}_k(\hat{\mu}_T) = \frac{1}{T} \sum_{t=1}^T |z_t - \xi_k|.$$

Theorem 3 predicts that

$$W_\rho(\hat{\mu}_T, \mu) < \frac{\Gamma}{2} \implies \hat{k}_T = k^*.$$

In one dimension, we estimate

$$W_\rho(\hat{\mu}_T, \mu) = \int_{\mathbb{R}} |\hat{F}_T(t) - F(t)| dt,$$

where $\hat{F}_T$ is the empirical CDF and F is the Gaussian CDF.

Voronoi-cell intuition. Figure 12 illustrates the geometric meaning of the semantic-cell recovery theorem. The semantic centers partition the reference space into Voronoi cells, and the population feature distribution lies inside one cell with positive separation from the nearest boundary. The key implication is that semantic grouping is not determined by exact sample locations, but by whether the empirical distribution remains inside the same margin-defined region. Thus, small sampling fluctuations are harmless when the population distribution is well separated from competing semantic centers. This is the geometric reason why large-margin features can be compressed into the same supernode confidently.

Convergence of assignment scores. Figure 13 shows that the empirical assignment scores $Q_1(\hat{\mu}_T)$ and $Q_2(\hat{\mu}_T)$ concentrate around their population values $Q_1(\mu)$ and $Q_2(\mu)$ as T increases. Since $Q_1(\mu) < Q_2(\mu)$, the population assignment is $k^* = 1$. The empirical assignment becomes stable once the empirical score gap remains positive:

$$Q_2(\hat{\mu}_T) - Q_1(\hat{\mu}_T) > 0.$$

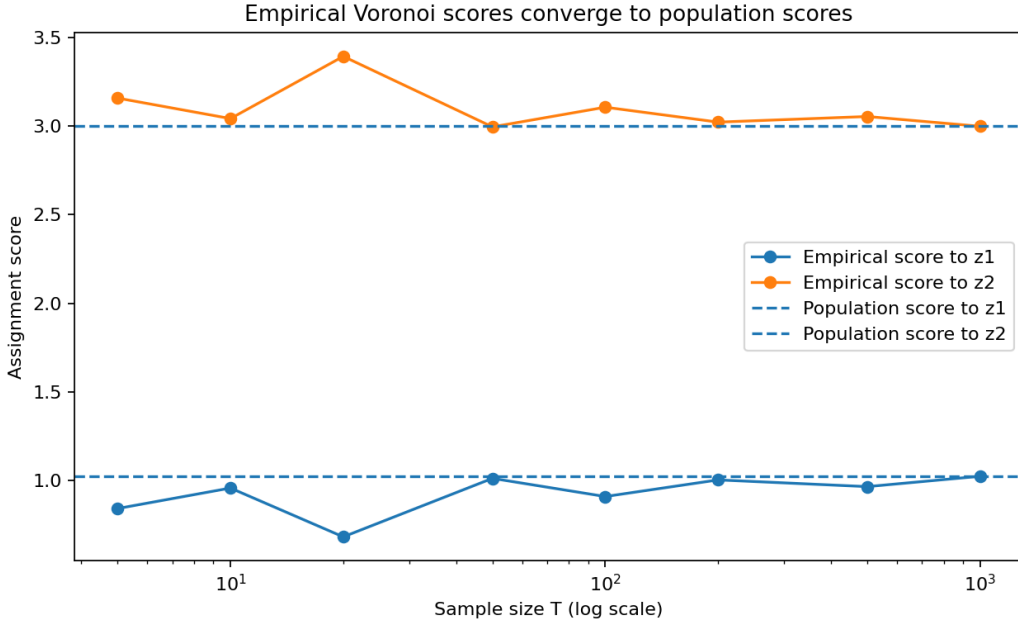


Figure 13: **Convergence of empirical Voronoi assignment scores.** For each semantic center $\xi_k \in \mathcal{R}$, the assignment score is $Q_k(\mu) = \int_{\mathcal{R}} \rho(z, \xi_k) d\mu(z)$, and the empirical score $Q_k(\hat{\mu})$ is computed from samples. The dashed horizontal lines show the population scores $Q_1(\mu)$ and $Q_2(\mu)$, while the solid curves show the empirical scores as the sample size T increases. The empirical scores concentrate around their population values, and the score to the correct semantic center ξ_1 remains smaller than the score to the competing semantic center ξ_2 . This supports the theorem’s mechanism: semantic recovery follows once the empirical score perturbation is small relative to the population assignment margin.

Verification of the recovery certificate. Figure 14 compares the empirical Wasserstein error $W_\rho(\hat{\mu}_T, \mu)$ against the theorem threshold $\Gamma/2$. As sample size increases, the Wasserstein error

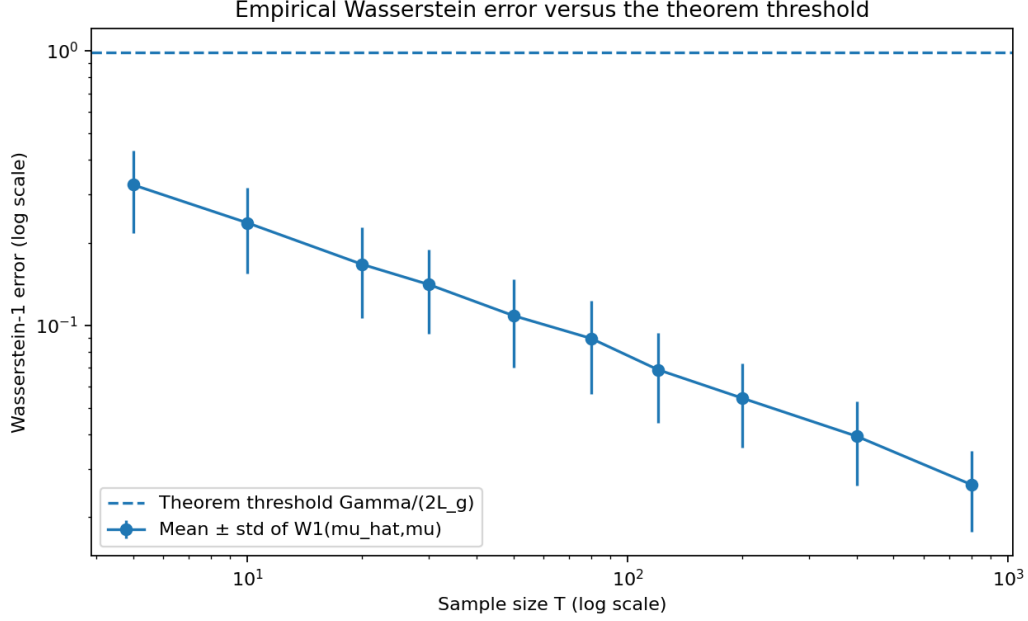


Figure 14: **Empirical Wasserstein error versus the Voronoi recovery threshold.** The blue curve shows the empirical Wasserstein error $W_\rho(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})})$ as a function of sample size T , averaged over Monte Carlo trials with error bars indicating one standard deviation. The dashed horizontal line is the theoretical recovery threshold $\Gamma_i/2$, where Γ_i is the population Voronoi assignment margin of feature i . The Voronoi recovery theorem guarantees correct semantic-cell recovery whenever $W_\rho(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}, \mu_i^{(A \rightarrow \mathcal{R})}) < \Gamma_i/2$. Thus, the plot visualizes the certificate condition: as T grows, the empirical projected distribution converges toward the population projected distribution, making the sufficient recovery condition easier to satisfy.

decreases. Once

$$W_\rho(\hat{\mu}_T, \mu) < \frac{\Gamma}{2},$$

Theorem 3 certifies that the empirical semantic-cell assignment must equal the population assignment. This verifies the theorem’s central mechanism: semantic recovery follows from Wasserstein convergence plus a positive assignment margin.

Empirical margin convergence. Figure 15 plots the empirical margin

$$\hat{\Gamma}_T = \mathcal{Q}_2(\hat{\mu}_T) - \mathcal{Q}_1(\hat{\mu}_T)$$

against the population margin

$$\Gamma = \mathcal{Q}_2(\mu) - \mathcal{Q}_1(\mu).$$

The empirical margin concentrates around Γ as T increases. This gives a direct empirical view of why the assignment stabilizes: the empirical distribution may fluctuate at small sample sizes, but the gap between the assigned center and the nearest competing center converges to the positive population margin.

Conclusion. The synthetic experiment confirms the qualitative behavior predicted by Theorem 3. Larger sample sizes reduce $W_\rho(\hat{\mu}_T, \mu)$, the empirical assignment scores converge to their population scores, and the empirical margin converges to the population Voronoi margin. Therefore, once the empirical Wasserstein error falls below the stability radius $\Gamma/2$, the semantic-cell assignment is certified to recover the population assignment. This supports the use of margin-aware semantic grouping for circuit compression: features with large Voronoi margins are stable under finite-sample perturbations, while features near semantic boundaries require more samples or should be treated as ambiguous.

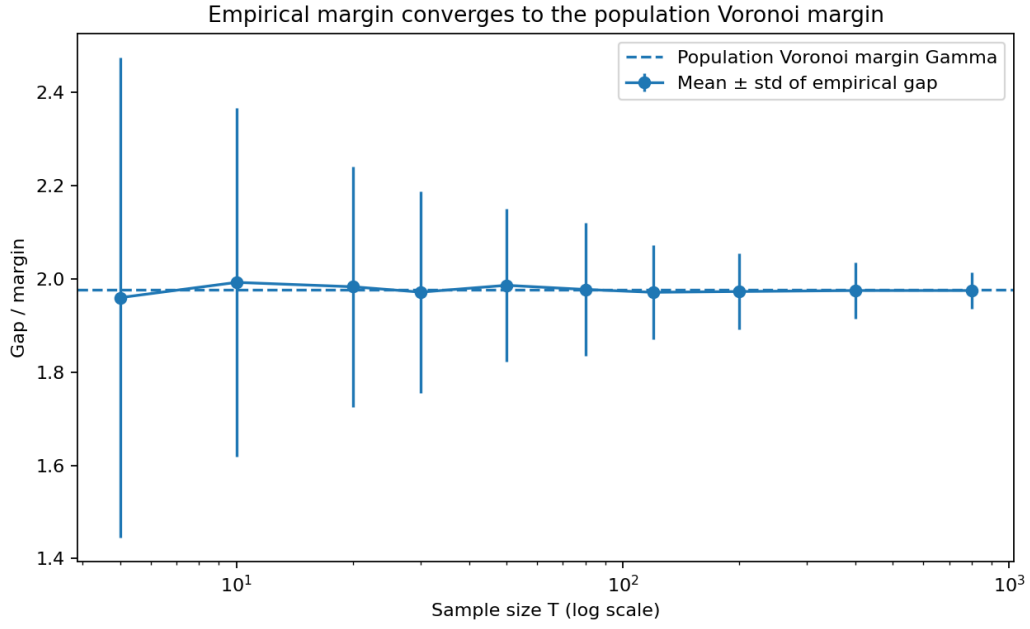


Figure 15: **Convergence of the empirical Voronoi margin.** The dashed horizontal line shows the population Voronoi margin $\Gamma_i = Q_2(\mu_i^{(A \rightarrow \mathcal{R})}) - Q_1(\mu_i^{(A \rightarrow \mathcal{R})})$, where ξ_1 is the correct semantic center and ξ_2 is the nearest competing semantic center. The solid curve shows the empirical margin $Q_2(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})}) - Q_1(\hat{\mu}_{i,T}^{(A \rightarrow \mathcal{R})})$ averaged over Monte Carlo trials, with error bars indicating one standard deviation. As the sample size T increases, the empirical margin concentrates around the population margin. This illustrates why semantic assignment becomes stable with more samples: once the empirical margin remains positive and sufficiently far from zero, the correct Voronoi cell is preserved.

W2: Can Δ_i and ϵ_i be estimated on real features to make the recovery theorem empirically checkable? Yes. Although the population quantities are unobservable, they can be approximated from finite corpora to obtain a feature-level stability certificate.

- **Estimating the margin.** For target feature i , we define $\hat{\Delta}_i = \hat{D}_{i,(2)}^{\text{ref}} - \hat{D}_{i,(1)}^{\text{ref}}$, where the two terms are the second-best and best OT scores on a large reference corpus. A larger $\hat{\Delta}_i$ means that the observed winner is more clearly separated from its closest competitor.
- **Estimating score perturbation.** We divide an independent evaluation corpus into B disjoint splits, each containing T tokens. On split b , we compute $\hat{\epsilon}_{i,T,b} = \max_j |\hat{D}_{ij,T}^b - \hat{D}_{ij}^{\text{ref}}|$. This measures how much the feature-matching scores change when estimated from a finite corpus.
- **Avoiding test leakage.** When split b is evaluated, its threshold is estimated only from the remaining splits: $\hat{\epsilon}_{i,T,b}^U = \max_{b' \neq b} \hat{\epsilon}_{i,T,b'}$. Thus, the tested split is not used to determine its own certificate.
- **Empirical certificate.** The held-out match is approximately certified when $\hat{\Delta}_i > 2\hat{\epsilon}_{i,T,b}^U$. The factor of two covers the worst case in which the winner’s score increases by ϵ_i while a competitor’s score decreases by the same amount.
- **Reported metrics.** Overall agreement is the fraction of held-out matches agreeing with the reference match. Certified coverage is the fraction satisfying the certificate condition. Certified agreement is the recovery rate among only the certified matches.

Table 9: Approximate empirical recovery certificates for Gemma-2-2B layer 13 $\rightarrow$ layer 12.

T	Med. $\hat{\Delta}$	Med. $\hat{\epsilon}^U$	Med. $\frac{\hat{\Delta}}{2\hat{\epsilon}^U}$	Overall agr.	Cert. coverage	Cert. agr.
1M	0.83	1.43	0.29	70.4%	6.2%	96.8%
2M	0.84	1.11	0.38	77.2%	10.2%	98.0%
4M	0.85	0.83	0.51	84.0%	18.2%	97.8%
8M	0.86	0.63	0.68	89.6%	30.6%	98.7%
16M	0.87	0.48	0.91	93.8%	45.2%	98.7%
32M	0.88	0.37	1.19	96.6%	63.8%	99.1%
64M	0.89	0.29	1.54	98.4%	80.2%	99.3%

- **The estimated quantities behave as expected.** As T increases, $\hat{\Delta}_i$ remains stable while $\hat{\epsilon}_{i,T,b}^U$ decreases. The median certificate ratio therefore rises from 0.29 to 1.54.
- **The certificate becomes less selective with more data.** Certified coverage increases from 6.2% at 1M tokens to 80.2% at 64M tokens.
- **Certified matches are highly reliable.** At $T = 8M$, only 30.6% of matches are certified, but 98.7% of the certified matches agree with the reference. The certificate therefore abstains on ambiguous features and retains a more stable subset.

Thus, $\hat{\Delta}_i$ and $\hat{\epsilon}_{i,T,b}^U$ make the recovery condition approximately checkable on real features. This is a finite-corpus stability certificate, not an exact guarantee for the unknown population quantities.

R4: Sample size for margin recovery. Thank you. We will clarify the following points:

- **What App. E.8 provides.** App. E.8 does not give one universal sample size for all features. Instead, the sufficient sample size depends on the feature-specific semantic margin: $T_i = O(\Gamma_i^{-d})$ in the generic d -dimensional case and $T_i = O(D_{\max}^2 \Gamma_i^{-2} [S + \log(1/\delta)])$ under finite support. Thus, large-margin features require fewer samples, while features near a semantic boundary may require substantially more.
- **Goal of the experiment.** We test whether the sufficient condition $W_1(\hat{\mu}_{i,T_{\text{scan}}}^C, \hat{\mu}_i^A) < \hat{\Gamma}_i/2$ can be satisfied on real Gemma features at practical corpus sizes. This determines whether the guarantee is empirically useful rather than only a formal asymptotic statement.

- **Why the held-out setup is appropriate.** Since the population distribution and margin are unobservable, split A defines the reference distributions and assignments, split B identifies the 57 features whose assignments are stable on independent data, and split C is used only for final evaluation. Thus, the test data are not used to define their own reference assignments or certificates.
- **Sample-size definitions.** T_{scan} is the number of corpus tokens processed. T_{eff} is the number of positive-activation contexts actually used to construct each empirical feature distribution.
- **Meaning of the reported metrics.** Recovery is agreement with the split-A reference assignment. Certified coverage is the fraction satisfying the sufficient Wasserstein condition. Certified agreement is the recovery rate among certified trials.

T_{scan}	Med. T_{eff}	Trials	Recovery	Cert. coverage	Cert. agreement
500K	282,704	285	98.2%	62.1%	100.0%
1M	568,996	285	98.6%	66.7%	100.0%
2M	1,137,725	285	98.2%	68.8%	100.0%
4M	2,273,999	228	98.7%	69.3%	100.0%
8M	4,548,756	114	98.2%	70.2%	100.0%
16M	9,097,512	57	98.2%	70.2%	100.0%

- **Main result.** The sufficient condition is already non-vacuous at the smallest tested budget of 500K tokens, where it certifies 62.1% of held-out trials. Every certified assignment recovers the reference assignment, giving 100% certified agreement and zero certified failures.
- **Why coverage is lower than recovery.** The Wasserstein condition is sufficient but not necessary. Therefore, an empirical assignment may be correct even when the theorem abstains from certifying it.
- **Why coverage plateaus.** Certified coverage increases to 70.2% at 8M tokens and then plateaus. This is consistent with the feature-dependent theory: small-margin features near a semantic boundary may remain uncertified even with large samples.
- **Large-margin subset.** On the large-margin subset pre-specified using split B, all held-out trials are correctly recovered and certified at the smallest tested budget of 500K tokens. This supports the predicted inverse relation between the required sample size and the semantic margin.
- **Scope of the conclusion.** Since we did not test budgets below 500K, we do not claim that 500K is the exact minimum. It is a sufficient tested budget for the large-margin subset, while the condition is already non-vacuous for the broader set of 57 stable features.

We will add this empirical evaluation alongside the sample-complexity analysis in App. E.8. We will also clarify that L12→L13 feature matching is governed by Thm. 2, whereas Thm. 3 concerns semantic-supernode recovery.