PREDICTING MULTI-VIEW RASHOMON REPRESENTATION: CAN WE LEARN WHERE MODELS DISAGREE?

Anonymous authors
Paper under double-blind review

ABSTRACT

Foundation models are increasingly adopted across a wide range of applications, often serving as core blocks within AI systems. Yet different foundation models may encode the same input from multiple different views, leading to substantial representation disagreement, which we term ***Rashomon Representation***. Such disagreement often signals inputs that a given model encodes in a way inconsistent with other models, offering a valuable yet underexplored signal for input reliability estimation. While prior work has largely focused on ***measuring disagreement*** across multiple models with a representation set, we instead focus on ***predicting disagreement*** from a single representation. We hypothesize that this disagreement follows some consistent, input-dependent patterns rather than occurring at random. To test this, we quantify disagreement by comparing each sample's nearest neighbors across different models' representation spaces, then train a lightweight predictor that estimates disagreement from a single model's representation. At inference time, given a new input, the predictor uses that input's representation to tell whether it aligns with or diverges from those of other models. Extensive experiments across diverse foundation models and datasets show that representational disagreement is indeed input-dependent, predictable, and generalizable, enabling efficient reliability estimation of foundation models.

1 INTRODUCTION

"The same event, told by different witnesses, becomes different events entirely."

— *Rashomon*, Akira Kurosawa, 1950

The rapid development of foundation models has enabled powerful and generalizable capabilities across a wide range of domains and applications (Awais et al., 2025). Models such as CLIP (Radford et al., 2021), DINOv2 (Oquab et al., 2024) have demonstrated remarkable transferability, establishing a "pretrain once, adapt everywhere" paradigm. Despite their shared versatility, however, these models can learn fundamentally different representations. Differences in training objectives, architectures, and data distributions can lead them to encode the same input in substantially different ways (Ciernik et al., 2025; Klabunde et al., 2025). For example, given an image, one model may focus more on high-level semantic attributes, while another may emphasize structural patterns (Barsellotti et al., 2025; Zhang & Tan, 2025). As shown in Fig. 1, different foundation models can produce distinct representations of the same input despite capturing similarly meaningful information. We refer to this phenomenon as the *Rashomon Representation*. Such discrepancies reflect representation-level uncertainty that can affect downstream reliability (Park et al., 2024). Understanding when they arise is therefore important for building robust and reliable AI systems.

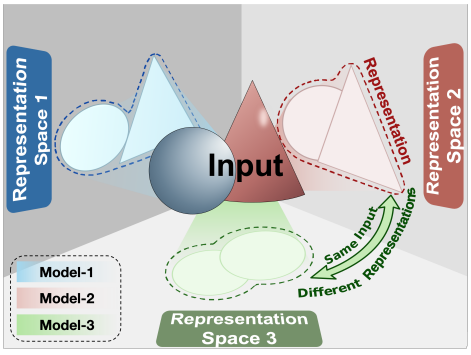

Figure 1: Illustration of *Rashomon Representation*: different foundation models can produce distinct representations of the same input, reflecting complementary views of its underlying information.

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

Prior studies have largely focused on **measuring representation differences** through direct comparisons between representation spaces, at either the dataset or sample level. At the dataset level, methods such as RSA, SVCCA, and CKA measure global similarity between representation spaces (Kriegeskorte et al., 2008; Raghu et al., 2017; Kornblith et al., 2019). Related studies further examine whether representations learned by different models converge toward similar structures (Li et al., 2015; Huh et al., 2024). Such analyses provide an aggregate view of representation similarity, but do not directly capture how cross-model disagreement varies across individual inputs (Kolling et al., 2025). Recent sample-level approaches address this limitation by quantifying representational agreement for individual inputs. For example, PNKA compares an input's similarity profiles across two representation spaces, while neighborhood consistency measures the overlap of its nearest-neighbor sets across multiple pretrained models (Kolling et al., 2025; Park et al., 2024). In essence, prior methods ask a **measurement** question: given representations, how much do models disagree on a particular input? We instead ask a **prediction** question: can such *Rashomon Representation*, i.e., cross-model representational disagreement, be predicted from a representation?

We introduce a supervised framework for sample-level prediction of *rashomon representation* across foundation models. Specifically, we introduce a neighborhood-consistency measure to quantify inter-model representational disagreement. The resulting agreement scores then serve as supervision for training a lightweight predictor that predicts *rashomon representation* directly from the representation produced by a single foundation model. Then, we systematically evaluate the predictability of *rashomon representation* across diverse experimental settings and find that it is consistently predictable from individual model representations. The observed predictability reveals three properties of *rashomon representation*. First, the degree of disagreement varies substantially across different inputs. Second, this variation is not random, but follows systematic and learnable patterns. Third, these patterns can be identified from the representation of a single foundation model, indicating that cross-model disagreement is partially encoded in individual representations. Together, these findings provide a practical mechanism for anticipating representation-level uncertainty at inference time. Our contributions are as follows:

- We identify and formalize *Rashomon Representation*, the phenomenon that different foundation models may encode the same input in substantially different ways, and formulate a new task of Rashomon Representation prediction to anticipate such cross-model disagreement from the representation of a single model.
- We develop a practical framework for this task. Specifically, we quantify sample-level disagreement using cross-model nearest-neighbor consistency over a reference dataset and train a lightweight predictor to estimate this disagreement from representations.
- Through extensive experiments, we show that representational disagreement is predictable, and generalizable, and further demonstrate its practical utility for efficiently identifying samples that are potentially vulnerable to unreliable representations across foundation models.

2 METHODS

We investigate whether Rashomon Representation, referring to sample-level representational disagreement across foundation models, can be predicted from the representation of a single foundation model. To this end, we formulate its prediction as a supervised learning problem. We first quantify sample-level cross-model representational disagreement by comparing local neighborhoods across the representation spaces of a pool of foundation models, yielding a disagreement score for each sample. These scores are then used as supervision to train a lightweight predictor that predicts agreement scores from representations produced by a designated anchor model. Once trained, the predictor can estimate the expected disagreement of a new sample using only its anchor-model representation at inference time.

2.1 QUANTIFYING REPRESENTATIONAL DISAGREEMENT

Setup and motivation. Given an input sample x and a pool of M pretrained foundation models $\mathcal{H} = \{h_0, h_1, \ldots, h_{M-1}\}$, our goal is to predict, using only the representation $h_0(x)$ produced by an *anchor model* h_0, how consistently x is represented across the model pool. We denote the remaining models by $\mathcal{H}_{-0} = \mathcal{H} \setminus \{h_0\}$ and use them to characterize cross-model representational

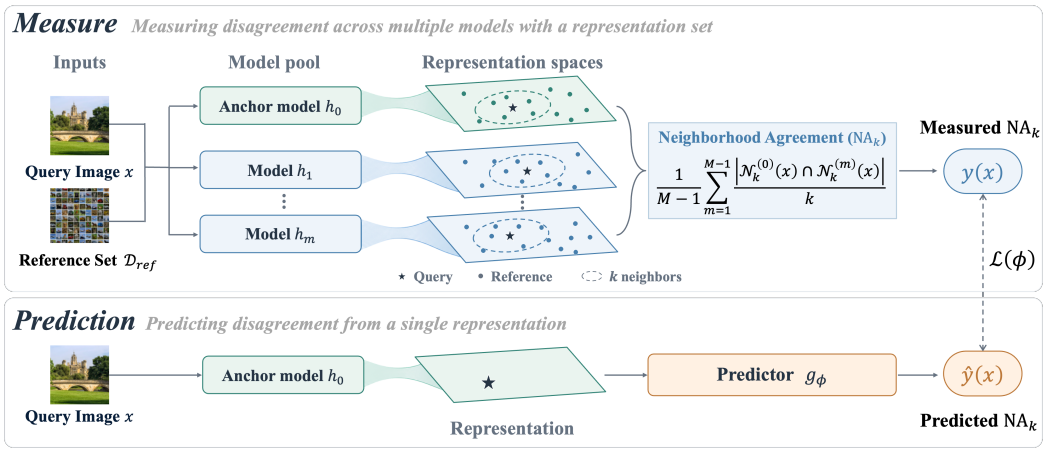

Figure 2: Training framework for predicting inter-model representational disagreement. We measure neighborhood agreement between an anchor model and comparison models over a shared reference set to obtain the supervision signal $y(x)$. A lightweight predictor g_ϕ learns to approximate $y(x)$ using only the anchor-model representation by minimizing $\mathcal{L}(\phi)$. All foundation models remain frozen, and only the predictor parameters are optimized.

agreement. Directly comparing $h_0(x)$ with $\{h_m(x)\}_{m=1}^{M-1}$ is generally not meaningful, since independently trained models encode samples in model-specific representation spaces that are not directly comparable. Rather than aligning these spaces explicitly, we compare each sample's local neighborhoods across models over the same reference samples, providing a parameter-free comparison that requires neither additional training nor assumptions about cross-model representation alignment. Specifically, for each model, we retrieve the local neighborhood of x from the same reference set and measure cross-model agreement by the overlap between these neighborhoods. This alignment-free consistency score is then used as supervision for our predictor, with lower scores indicating stronger representational disagreement.

Model-specific neighborhood construction. Let $\mathcal{D} = \{x_1, \ldots, x_N\}$ denote a reference dataset shared across all models. For each model h_m, we characterize how the sample x is locally positioned in the representation space by identifying its k nearest neighbors in $\mathcal{D}$. Our construction only requires a model-specific similarity measure to rank the reference samples and does not depend on a particular choice of metric. In our implementation, we use cosine similarity, a scale-invariant measure commonly used for comparing learned representations. Equivalently, after ℓ_2 normalization, the similarity between x and a reference sample $x_j \in \mathcal{D}$ is given by

$$s_m(x, x_j) = \frac{h_m(x)^\top h_m(x_j)}{\|h_m(x)\|_2 \, \|h_m(x_j)\|_2}.$$

We then define $\mathcal{N}_k^{(m)}(x)$ as the index set of the k nearest neighbors with the largest similarity scores:

$$\mathcal{N}_k^{(m)}(x) = \mathrm{TopK}_{j\in\{1,\ldots,N\}} \; s_m(x, x_j).$$

Although the resulting neighborhoods are constructed independently in model-specific representation spaces, they are all expressed over the same set of reference indices. Thus, the neighborhoods can be compared directly across models without explicitly aligning their representation spaces.

Cross-model neighborhood agreement. Based on model-specific neighborhoods, we quantify how consistently the anchor model h_0 represents x relative to the remaining models in $\mathcal{H}_{-0}$. We define

$$\mathrm{NA}_k(x; h_0, \mathcal{H}) = \frac{1}{M-1} \sum_{m=1}^{M-1} \frac{|\mathcal{N}_k^{(0)}(x) \cap \mathcal{N}_k^{(m)}(x)|}{k}. \tag{1}$$

where $|\mathcal{N}_k^{(0)}(x) \cap \mathcal{N}_k^{(m)}(x)|$ denotes the number of k-nearest neighbors shared by the anchor model h_0 and comparison model h_m. NA_k measures the neighborhood agreement of x at neighborhood

size k, quantifying how consistently the local neighborhood identified by the anchor model is preserved across the comparison models. For each comparison model h_m, the corresponding term measures the fraction of reference neighbors selected by the anchor model that are also selected by h_m. Averaging these pairwise overlaps over $\mathcal{H}_{-0}$ yields a sample-level measure of representational consistency relative to h_0. The score ranges from 0 to 1, with higher values indicating greater agreement in local neighborhood structure across models and lower values indicating stronger representational disagreement.

Neighborhood agreement as supervision. Since NA_k is defined solely through neighborhood overlap, its computation requires no additional model training and does not require other assumptions. It is therefore model-agnostic and can be applied to a broad range of representation models, as long as their representations can be used to construct local neighborhoods. These properties make NA_k a simple and broadly applicable measure, and we use NA_k as the sample-level supervision signal for predicting cross-model representational consistency from $h_0(x)$ alone.

2.2 From Measurement to Prediction

Predicting cross-model agreement. The neighborhood agreement score introduced above quantifies, for each sample, the degree of *Rashomon Representation*, i.e., how consistently the same input is represented across different foundation models. However, explicitly computing this score requires access to multiple models and a reference dataset at inference time, limiting its practical applicability. This raises a more fundamental question: can the *Rashomon Representation* be inferred from its representation in a single model? To test this, we use the measured consistency scores as prediction targets and train a lightweight predictor whose only input is the representation produced by an *anchor model*. Successful prediction would indicate that rashomon representation is already reflected, at least partially, in the representation space of an individual model, rather than being a property that becomes observable only after comparing multiple models.

Prediction formulation. For an input x, let $\tilde{z}_0(x) = h_0(x)/\|h_0(x)\|_2 \in \mathbb{R}^{d_0}$ denote the ℓ_2-normalized representation produced by the anchor model, where d_0 is its output dimensionality. We introduce a predictor $g_\phi : \mathbb{R}^{d_0} \to (0, 1)$ that produces $\hat{y}(x) = g_\phi(\tilde{z}_0(x))$ to approximate the target $y(x) = \mathrm{NA}_k(x; h_0, \mathcal{H})$. Higher predicted values indicate stronger expected agreement between the anchor and comparison models, while lower values indicate greater representational disagreement. Crucially, the predictor takes only $\tilde{z}_0(x)$ as input, without access to the comparison-model representations or the cross-model neighborhood comparisons used to define the target score. It therefore learns to predict cross-model consistency solely from the representation. If a lightweight predictor can do so reliably, this indicates that cross-model agreement is systematically associated with information already present in the representation.

Predictor construction. We instantiate g_ϕ as a lightweight residual feed-forward network that maps the normalized anchor representation to a scalar consistency estimate. The input is first projected into a hidden space and then processed by residual blocks composed of LayerNorm, GELU activations, dropout, and identity skip connections before the final prediction layer. Following standard residual feed-forward designs (Vaswani et al., 2017; Tolstikhin et al., 2021), we keep the predictor deliberately small and capacity-controlled so that its performance reflects how readily cross-model consistency can be predicted from a single representation, rather than the fitting power of a high-capacity auxiliary model. We additionally evaluate alternative predictor architectures in the experiments to examine whether the predictability depends on a particular architectural choice.

Predictor training. We use the predefined, disjoint training, validation, and test splits, denoted by $\mathcal{D}_{\mathrm{tr}}$, $\mathcal{D}_{\mathrm{val}}$, and $\mathcal{D}_{\mathrm{te}}$, respectively, and set $\mathcal{D}_{\mathrm{ref}} = \mathcal{D}_{\mathrm{tr}}$. For each sample x, we precompute the target $y(x) = \mathrm{NA}_k(x; h_0, \mathcal{H})$ using Eq. 1, excluding the query itself for samples in $\mathcal{D}_{\mathrm{tr}}$. The predictor is trained on $\mathcal{D}_{\mathrm{tr}}$ by minimizing

$$\mathcal{L}(\phi) = \frac{1}{|\mathcal{D}_{\mathrm{tr}}|} \sum_{x_i \in \mathcal{D}_{\mathrm{tr}}} \ell(g_\phi(\tilde{z}_0(x_i)), y(x_i)) \,,$$

where ℓ denotes the regression loss. The checkpoint with the lowest validation loss on $\mathcal{D}_{\mathrm{val}}$ is selected and finally evaluated on $\mathcal{D}_{\mathrm{te}}$. All foundation models remain fixed, and only the predictor parameters ϕ are optimized.

3 Related Work

Representational Similarity. A central question in representation analysis is whether different models organize the same inputs in similar ways. Existing methods typically compare the similarity structures induced by different models on a shared dataset. Representational similarity analysis (RSA) measures the correspondence between these structures (Kriegeskorte et al., 2008), while SVCCA and CKA provide more systematic tools for comparing representations across layers, architectures, and training procedures (Raghu et al., 2017; Kornblith et al., 2019). Related studies further investigate the convergence of independently trained representations and the effects of architectural and pretraining choices (Li et al., 2015; Nguyen et al., 2021; Raghu et al., 2021; Neyshabur et al., 2020; Xie et al., 2023). These approaches primarily produce retrospective summaries at the model, layer, or dataset level. More recent work extends representation comparison to individual inputs. For example, PNKA compares an input's similarity profiles across representation spaces, while neighborhood-based methods measure agreement through the overlap of nearest-neighbor sets (Park et al., 2024; Kolling et al., 2025). Such methods capture input-dependent variation more directly, but remain measurement-oriented: they require representations from multiple models to quantify the disagreement of a given input. In contrast, our work reformulates this problem as prediction. We ask whether cross-model representational disagreement, which we term *Rashomon Representation*, can be inferred from the representation of a single model.

Vision Foundation Models are commonly developed with different pretraining objectives and architectures. Image-language contrastive learning, as used in CLIP, aligns visual representations with natural language (Radford et al., 2021). Self-supervised methods adopt contrastive, momentum-based, bootstrapping, or self-distillation objectives, including SimCLR, MoCo, BYOL, DINO, and DINOv2 (Chen et al., 2020; He et al., 2020; Grill et al., 2020; Caron et al., 2021; Oquab et al., 2024). Masked-image modeling methods, such as BEiT and MAE, learn representations by recovering masked visual content (Bao et al., 2022; He et al., 2022). In terms of architecture, Vision Transformers provide a scalable alternative to convolutional visual encoders (Dosovitskiy et al., 2021). In our experiments, we consider a broader range of vision foundation models. Taking vision foundation models as a representative domain, we conduct extensive experiments to investigate the predictability and generality of cross-model representational disagreement.

4 Experiments

Our experiments are designed to test the central hypothesis underlying Rashomon Representation: inter-model representational disagreement is not random, but follows input-dependent patterns that can be predicted from a single model representation. We organize our investigation around two aspects. First, we establish predictability from a single representation and examine whether it persists across model configurations, distribution shifts, and predictor architectures, as well as when the anchor model is excluded from the disagreement target. Second, we assess the practical value of predicted disagreement by examining its association with downstream classification risk and the reductions in computation and reference data required to estimate disagreement.

Experimental setup. Unless otherwise specified, we use disjoint ImageNet-1K (Deng et al., 2009; Russakovsky et al., 2015) splits for predictor training, validation, and testing, with the full training split as the neighborhood reference set. Our default configuration uses CLIP ViT-B/32 (Radford et al., 2021) as the anchor model, with DINOv2 ViT-B/14 (Oquab et al., 2024) and a supervised ViT-B/16 (IN-21K) (Dosovitskiy et al., 2021) as comparison models. Below, we refer to these models as CLIP, DINOv2, and ViT-21K, respectively. We use a lightweight Residual FFN predictor built around residual feed-forward blocks and set the neighborhood size to $k = 50$ when computing the neighborhood agreement target NA_k. In this default setting, the predictor estimates neighborhood agreement using only the anchor representation, with agreement serving as an operational proxy for inter-model representational disagreement. To address different experimental questions, we vary model combinations, predictor architectures, data distributions, reference-set sizes, and other relevant settings. We report the key settings for each experiment in the main text and provide additional implementation details in the appendix.

4.1 DISAGREEMENT IS PREDICTABLE

Disagreement is predictable from a single representation. We first ask whether inter-model representational disagreement can be predicted from the representation of a single model. To test this, we take each of CLIP, DINOv2, and ViT-21K in turn as the anchor model and use the other two as comparison models. For each anchor, we train a separate predictor using only the anchor's representation as input to estimate the measured NA_k. As shown in Fig. 3, the predicted and measured NA_k values are positively correlated for all three anchor models, with Pearson correlation coefficients ranging from 0.708 to 0.746. These results support the hypothesis that inter-model representational disagreement leaves a detectable sample-level signature within each model's representation space.

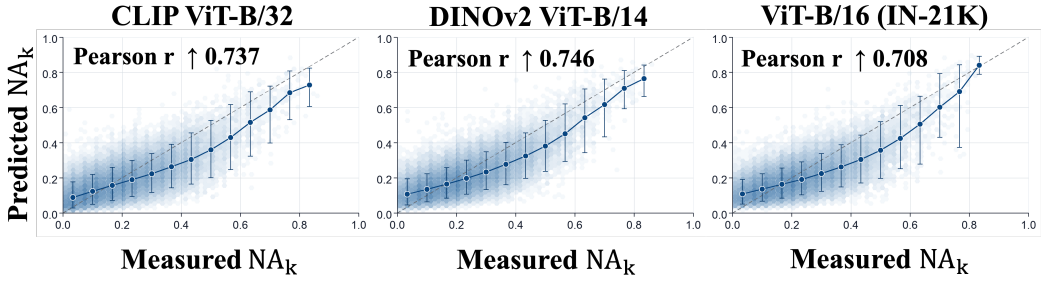

Figure 3: Measured and predicted NA_k across three encoders. Hexagons indicate sample density, while markers and error bars show binned means and 90% intervals. The strong agreement between measured and predicted values demonstrates that cross-model representational disagreement can be reliably inferred from single-model representations.

Predictability across model configurations. We next examine whether disagreement predictability depends on the anchor model or the set of comparison models. Building on the preceding experiment, we further introduce Swin-B pretrained on ImageNet-22K (Liu et al., 2021), SigLIP 2 with a ViT-B/16 image encoder at 224×224 resolution (Tschannen et al., 2025), and ConvNeXt-B pretrained on ImageNet-22K (Liu et al., 2022). Across 11 configurations, we train a separate predictor using only the designated anchor representation. The dataset setup remains unchanged from the preceding experiment, while we vary the composition and size of the comparison set, as well as the choice of anchor model.

(i) Anchor model choice. To assess how the choice of anchor model affects disagreement predictability, we evaluate six different models. We use Swin-B, SigLIP 2, and ConvNeXt-B as the comparison model set when evaluating CLIP, DINOv2, and ViT-21K, and use CLIP, DINOv2, and ViT-21K as the comparison model set when evaluating the other three models. As shown in Table 1, predicted and measured NA_k remain positively correlated across all six configurations, with Pearson r ranging from 0.701 to 0.770. These results show that inter-model representational disagreement can be learned regardless of which model supplies the representation to the predictor and serves as the anchor model in the agreement computation.

Table 1: Disagreement prediction across anchor-model choices.

Comparison models	Anchor	Pearson r ↑
Swin-B SigLIP 2 ConvNeXt-B	CLIP	0.734
	DINOv2	0.751
	ViT-21K	0.701
CLIP DINOv2 ViT-21K	Swin-B	0.770
	SigLIP 2	0.750
	ConvNeXt-B	0.767

(ii) Composition of the comparison model set. We first fix DINOv2 as the anchor model and the number of models in the comparison model set at three, while varying which model types constitute the set. As shown in Table 2a, the three comparison model sets yield Pearson correlations of 0.751, 0.754, and 0.762, respectively. The consistently high positive correlations indicate that inter-model representational disagreement remains predictable across different compositions of the comparison model set and is not limited to one particular model combination.

(iii) Comparison model set size. We next examine whether inter-model representational disagreement remains predictable when the number of models in the comparison model set changes. To test this, we fix CLIP as the anchor model and use nested comparison model sets containing two

to five models. As shown in Table 2b, performance remains comparable across all four settings, with Pearson r ranging from 0.727 to 0.750. These results show that inter-model representational disagreement can be learned across the tested comparison model set sizes and maintains strong predictive performance when more models are included in the comparison model set.

Table 2: Disagreement prediction across comparison-model compositions and comparison model set sizes.

(a) Composition (DINOv2 anchor).

Comparison models	**Pearson r ↑**
Swin-B, SigLIP 2, ConvNeXt-B	0.751
Swin-B, SigLIP 2, ViT-21K	0.754
CLIP, SigLIP 2, ConvNeXt-B	0.762

(b) Set size (CLIP anchor).

Comparison models	**Pearson r ↑**
Swin-B, SigLIP 2	0.727
Swin-B, SigLIP 2, ConvNeXt-B	0.734
Swin-B, SigLIP 2, ConvNeXt-B, DINOv2	0.744
Swin-B, SigLIP 2, ConvNeXt-B, DINOv2, ViT-21K	0.750

Taken together, these experiments show that the predictability of inter-model representational disagreement persists across comparison panel compositions, anchor model choices, and panel sizes. These variations modify complementary components of the protocol, including the models used to construct the agreement target, the predictor input, and the breadth of the comparison panel. Nevertheless, the positive association between predicted and measured NA_k remains consistent, indicating that the observed predictability is not confined to any single model combination.

Disagreement remains predictable under distribution shift. We next ask whether disagreement remains predictable when the query distribution changes. To test this, we evaluate the ImageNet-1K-trained predictor on nine external datasets, four natural ImageNet shifts, and five Gaussian-noise levels without retraining or test-time adaptation. For each query, predicted NA_k is produced by applying this fixed predictor to the query's CLIP representation, while measured NA_k is computed by comparing the neighborhoods retrieved by CLIP, DINOv2, and ViT-21K from the same fixed ImageNet-1K training reference set. As shown in Tables 3 and 4, predicted and measured NA_k remain positively correlated across all settings, with Pearson r ranging from 0.405 to 0.661 on the external datasets, from 0.710 to 0.720 on ImageNet-V2, and reaching 0.512 on ImageNet-R. These results show that disagreement predictability transfers beyond the training distribution, although its strength varies across domains. This setting reflects practical use: the data available during training are finite, while future queries may come from a much broader range of distributions.

Table 3: Cross-dataset disagreement prediction. Pearson correlations between predicted and measured NA_k on nine external datasets using the fixed ImageNet-1K-trained predictor. Dataset sources: Aircraft (Maji et al., 2013), Caltech101 (Fei-Fei et al., 2004), Cars (Krause et al., 2013), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2019), Food101 (Bossard et al., 2014), Pets (Parkhi et al., 2012), SUN397 (Xiao et al., 2010), and UCF101 (Soomro et al., 2012).

	Aircraft	Caltech101	Cars	DTD	EuroSAT	Food101	Pets	SUN397	UCF101
Pearson r ↑	0.661	0.656	0.425	0.587	0.405	0.585	0.499	0.597	0.637

Table 4: Disagreement prediction under ImageNet distribution shifts. Pearson correlations between predicted and measured NA_k on ImageNet-V2 (Recht et al., 2019), ImageNet-R (Hendrycks et al., 2021), and Gaussian-corrupted ImageNet-C (Hendrycks & Dietterich, 2019). The labels S1, S2, S3, S4, and S5 denote corruption severity levels 1, 2, 3, 4, and 5, respectively.

	ImageNet-V2			ImageNet-R	ImageNet-C: Gaussian				
	Matched	Threshold 0.7	Top Images		S1	S2	S3	S4	S5
Pearson r ↑	0.710	0.720	0.717	0.512	0.726	0.732	0.724	0.650	0.394

Under Gaussian noise, Pearson r decreases from 0.726 at severity 1 to 0.394 at severity 5. In the three examples shown in Figure 4, absolute errors increase from 0.013 to 0.031 and 0.037 as the neighborhoods retrieved by CLIP, DINOv2, and ViT-21K become increasingly divergent. These observations suggest that severe corruption pushes query representations beyond the regime learned

from ImageNet-1K, causing the learned mapping from anchor representations to cross-model agreement to become unreliable and the predictions to drift.



Figure 4: Prediction drift under Gaussian-noise corruption. From left to right, corruption severity increases; each example shows a query and its three nearest neighbors retrieved by CLIP, DINOv2, and ViT-21K.

Predictability beyond the anchor. In the preceding experiments, the anchor model provides the predictor input, and its neighborhood also contributes to the supervision target. We further ask whether a single model's representation can predict representational disagreement within a specified set of other models. To test this, we train a predictor using only CLIP representations, with the target defined as the average pairwise top-50 neighborhood agreement among DINOv2, ViT-21K, and Swin-B, excluding CLIP. As Figure 5 shows, the predicted and measured agreement remain correlated, with Pearson $r = 0.578$. For this model set, the result shows that a CLIP representation provides information for predicting disagreement among models that do not include CLIP.

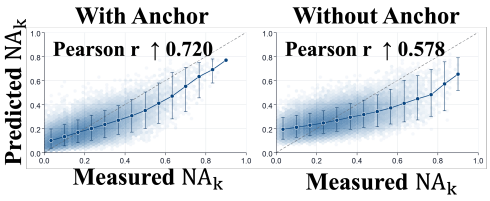

Figure 5: Leave-anchor-out diagnosis on ImageNet-1K. Both panels use only CLIP as input; the target includes CLIP on the left and excludes it on the right.

Predictability holds across predictor architectures. We assess architectural sensitivity by comparing a 2-layer MLP, Residual FFN, and Transformer Encoder across three anchor models, holding the predictor input and measured NA_k target fixed across architectures for each anchor. As shown in Table 5, all three architectures yield positive correlations for every anchor; the 2-layer MLP achieves Pearson r values ranging from 0.490 to 0.544, while the Residual FFN reaches from 0.708 to 0.746. These results show that different architectures recover the disagreement signal, indicating that its predictability is not an artifact of a particular design, although architecture affects prediction accuracy.

Table 5: Predictor architecture comparison measured by Pearson correlation. All predictors use the same single-representation input and measured consistency supervision.

Model	2-layer MLP	Transformer Encoder	Residual FFN
CLIP	0.544	0.712	**0.737**
DINOv2	0.523	0.724	**0.746**
ViT-21K	0.490	0.696	**0.708**

4.2 Predicted disagreement is practical

Prediction substantially reduces measurement cost. Computing measured NA_k requires neighborhood search across multiple representation spaces for every query. We compare its PyTorch and FAISS implementations with predicted NA_k, obtained through a single forward pass over the CLIP representation, on 100,000 ImageNet-1K test samples. As shown in Table 6, predicted NA_k takes only 5.34 ms per sample, making it $3.85\times$ faster than FAISS and $9.28\times$ faster than PyTorch, while reducing GPU memory and asset storage by over three orders of magnitude. These results show that prediction provides an efficient alternative to repeatedly measuring inter-model representational disagreement.

Table 6: Post-embedding cost on 100K ImageNet-1K samples. Lower is better.

Method	Time ↓ (ms/sample)	Peak memory ↓ (MiB)	Storage ↓ (MiB)
Measured (PyTorch)	49.54	20,643	10,009
Measured (FAISS)	20.54	29,729	10,009
Predicted	**5.34**	**9**	**5**

Predicted disagreement tracks downstream classification risk. Inter-model representational disagreement reflects differences in how foundation models organize the same input, which may lead to different behavior on downstream tasks. We therefore ask whether predicted disagreement retains the association between measured disagreement and downstream classification risk. Across nine benchmarks, we compute Kendall τ_b correlations of measured and predicted disagreement, each defined as $1 - \mathrm{NA}_k$, with the per-sample multiclass Brier risk of fixed, dataset-specific CLIP linear probes. As shown in Table 7, predicted disagreement is positively associated with risk on every dataset, and the average τ_b increases from 0.18 for measured disagreement to 0.21 for predicted disagreement. One possible explanation is that training filters some noise in the measured NA_k targets, allowing predicted disagreement to align more closely with downstream risk. These results indicate that predicted disagreement retains an association with downstream classification risk.

Table 7: Predicted representational disagreement tracks downstream classification risk. We report sample-wise Kendall τ_b correlations between measured or predicted disagreement and multiclass Brier risk across nine benchmarks. Red values indicate increases over directly measured disagreement; green values indicate decreases.

	Aircraft	Caltech101	Cars	DTD	EuroSAT	Food101	Pets	SUN397	UCF-101	Average
Measured	0.24	0.20	0.09	0.16	0.09	0.32	0.16	0.20	0.18	**0.18**
Predicted	0.26 +0.02	0.23 +0.03	0.10 +0.01	0.17 +0.01	0.08 -0.01	0.40 +0.08	0.17 +0.01	0.23 +0.03	0.23 +0.05	**0.21 +0.03**

Prediction reduces data requirements. Obtaining measured NA_k typically requires a large reference set. In contrast, estimating inter-model disagreement through prediction can leverage learned generalization to reduce dependence on reference-set size. We construct nested reference subsets ranging from 5K to 500K samples and use the measured NA_k from each subset to train a predictor on the same fixed 50,000 training samples. We then compute the Pearson correlation between predicted NA_k and measured NA_k obtained offline using the full reference set. To quantify the retained ability to approximate full-reference measured NA_k, we divide this correlation by that achieved by a predictor trained with full-reference supervision. As shown in Figure 6, even 100K references retain 82.3% of the full-reference predictive correlation, while 500K references, fewer than half of the full reference set, retain 90.6% and approach full-reference performance with substantially less data.

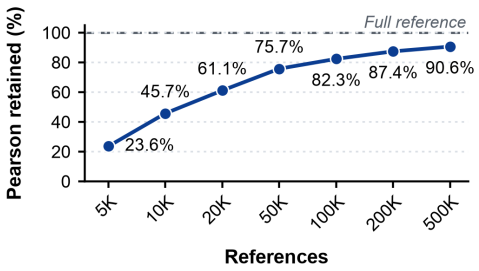

Figure 6: Reference set scaling for disagreement prediction. Pearson correlation retained relative to using the full ImageNet-1K training set as the reference set; all predictors use the same 50,000 training samples and are evaluated against the same full-reference NA_k target.

5 Conclusion

We presented a framework for predicting *Rashomon Representation*, the sample-dependent disagreement among foundation models, from a single model's representation. Cross-model neighborhood agreement operationalizes this disagreement and provides supervision for a lightweight predictor. Across diverse vision foundation models and evaluation settings, predicted agreement consistently tracks measured agreement, indicating that inter-model representational disagreement is structured, input-dependent, and partially encoded in individual representation spaces. This enables efficient inference-time estimation without computing representations and neighborhoods for every comparison model. Future work will investigate the mechanisms underlying this predictability and develop self-supervised approaches that learn representational disagreement without explicit cross-model measurements.

AI USE STATEMENT

In this work, we used generative AI tools solely to improve the grammar, wording, and readability of author-written text. The tools were not used to generate research ideas, formulate hypotheses or scientific claims, design the methodology or experiments, implement the proposed method, analyze or interpret experimental results, generate data or figures, or create citations. All AI-assisted edits were reviewed and revised by the authors for technical accuracy and consistency with the underlying evidence. We take full responsibility for the final content of this work.

REFERENCES

Muhammad Awais, Muzammal Naseer, Salman Khan, Rao Muhammad Anwer, Hisham Cholakkal, Mubarak Shah, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Foundation models defining a new era in vision: A survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(4):2245–2264, 2025.

Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT pre-training of image transformers. In *International Conference on Learning Representations*, 2022.

Luca Barsellotti, Lorenzo Bianchi, Nicola Messina, Fabio Carrara, Marcella Cornia, Lorenzo Baraldi, Fabrizio Falchi, and Rita Cucchiara. Talking to DINO: Bridging self-supervised vision backbones with language for open-vocabulary segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22025–22035, 2025.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *Computer Vision – ECCV 2014*, volume 8694 of *Lecture Notes in Computer Science*, pp. 446–461. Springer International Publishing, 2014.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607. PMLR, 2020.

Laure Ciernik, Lorenz Linhardt, Marco Morik, Jonas Dippel, Simon Kornblith, and Lukas Muttenthaler. Objective drives the consistency of representational similarity across datasets. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 10920–10948, 2025.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2004, Washington, DC, USA, June 27 - July 2, 2004*, pp. 178. IEEE Computer Society, 2004.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - a

new approach to self-supervised learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 21271–21284, 2020.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16000–16009, June 2022.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, July 2019.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8340–8349, October 2021.

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. Position: The platonic representation hypothesis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 20617–20642. PMLR, 2024.

Max Klabunde, Tassilo Wald, Tobias Schumacher, Klaus Maier-Hein, Markus Strohmaier, and Florian Lemmerich. Resi: A comprehensive benchmark for representational similarity measures. In *The Thirteenth International Conference on Learning Representations*, 2025.

Camila Kolling, Till Speicher, Vedant Nanda, Mariya Toneva, and Krishna P. Gummadi. Investigating the effects of fairness interventions using pointwise representational similarity. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3519–3529. PMLR, 2019.

Jonathan Krause, Michael Stark, Jia Deng, and Fei-Fei Li. 3d object representations for fine-grained categorization. In *2013 IEEE International Conference on Computer Vision Workshops*, pp. 554–561. IEEE, December 2013. 4th IEEE Workshop on 3D Representation and Recognition.

Nikolaus Kriegeskorte, Marieke Mur, and Peter A. Bandettini. Representational similarity analysis - connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, 2:4, 2008.

Yixuan Li, Jason Yosinski, Jeff Clune, Hod Lipson, and John Hopcroft. Convergent learning: Do different neural networks learn the same representations? In *Proceedings of the 1st International Workshop on Feature Extraction: Modern Questions and Challenges at NIPS 2015*, volume 44 of *Proceedings of Machine Learning Research*, pp. 196–212. PMLR, 2015.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11976–11986, 2022.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft, June 2013.

Behnam Neyshabur, Hanie Sedghi, and Chiyuan Zhang. What is being transferred in transfer learning? In *Advances in Neural Information Processing Systems*, volume 33, pp. 512–523, 2020.

Thao Nguyen, Maithra Raghu, and Simon Kornblith. Do wide and deep networks learn the same things? uncovering how neural network representations vary with width and depth. In *International Conference on Learning Representations*, 2021.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

Young-Jin Park, Hao Wang, Shervin Ardeshir, and Navid Azizan. Quantifying representation reliability in self-supervised learning models. In *Proceedings of the Fortieth Conference on Uncertainty in Artificial Intelligence*, volume 244 of *Proceedings of Machine Learning Research*, pp. 2835–2860. PMLR, 2024.

Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3498–3505. IEEE, June 2012.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. SVCCA: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? In *Advances in Neural Information Processing Systems*, volume 34, pp. 12116–12128, 2021.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do ImageNet classifiers generalize to ImageNet? In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5389–5400. PMLR, 2019.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Fei-Fei Li. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. Technical Report CRCV-TR-12-01, Center for Research in Computer Vision, University of Central Florida, November 2012. Also available as arXiv:1212.0402.

Ilya O. Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. MLP-Mixer: An all-MLP architecture for vision. In *Advances in Neural Information Processing Systems*, volume 34, pp. 24261–24272, 2021.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pp. 5998–6008, 2017.

Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3485–3492. IEEE, June 2010.

Zhenda Xie, Zigang Geng, Jingcheng Hu, Zheng Zhang, Han Hu, and Yue Cao. Revealing the dark secrets of masked image modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14475–14485, 2023.

Xin Zhang and Robby T. Tan. Mamba as a bridge: Where vision foundation models meet vision language models for domain-generalized semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14527–14537, 2025.

A Common Experimental Setup

A.1 Data Splits

We use predefined ImageNet-1K splits for predictor training, model selection, and final evaluation. The training, validation, and test splits, denoted by $\mathcal{D}_{\mathrm{tr}}$, $\mathcal{D}_{\mathrm{val}}$, and $\mathcal{D}_{\mathrm{te}}$, contain 1,281,167, 50,000, and 100,000 samples, respectively.

Except in the reference-set scaling experiment, the full training split serves as the shared neighborhood reference set:

$$\mathcal{D}_{\mathrm{ref}} = \mathcal{D}_{\mathrm{tr}}. \tag{2}$$

Neighborhoods for training, validation, and test queries are constructed with respect to this reference set. Representations and neighborhood sets are aligned by image identity across models so that cross-model comparisons refer to the same reference samples.

A.2 Foundation Models and Representations

We use six frozen vision encoders. Each model follows its corresponding image preprocessing pipeline, and the extracted representations are used for neighborhood construction and predictor training. Table 8 summarizes the model configurations.

Table 8: Foundation models and extracted representations.

Model	Model source	Representation	Dimension
CLIP ViT-B/32	`openai/clip-vit-base-patch32`	Projected image features	512
DINOv2 ViT-B/14	`facebook/dinov2-base`	Pooled output; CLS if unavailable	768
ViT-B/16	`google/vit-base-patch16-224-in21k`	Pooled output; CLS if unavailable	768
Swin-B	`microsoft/swin-base-patch4-window7-224-in22k`	Pooled output	1024
SigLIP 2 ViT-B/16	`google/siglip2-base-patch16-224`	Projected image features	768
ConvNeXt-B	`timm/convnext_base.fb_in22k`	Pooled pre-logit features	1024

The normalized representation of sample x under model h_m is

$$\tilde{z}_m(x) = \frac{h_m(x)}{\|h_m(x)\|_2}. \tag{3}$$

All foundation models remain frozen throughout the experiments. Cross-model neighborhood comparisons operate on reference image identities and do not require equal representation dimensionality.

A.3 Neighborhood Construction and Supervision

Following the main text, let

$$\mathcal{H} = \{h_0, h_1, \ldots, h_{M-1}\} \tag{4}$$

denote the foundation-model pool, where h_0 is the anchor and the remaining models form the comparison set. The similarity between query x and reference sample x_j under model h_m is

$$s_m(x, x_j) = \tilde{z}_m(x)^\top \tilde{z}_m(x_j). \tag{5}$$

We perform exact nearest-neighbor search using FAISS FlatIP to obtain $\mathcal{N}_k^{(m)}(x)$, with $k = 50$ by default. Pairwise neighborhood agreement is

$$a_k^{(u,v)}(x) = \frac{\left|\mathcal{N}_k^{(u)}(x) \cap \mathcal{N}_k^{(v)}(x)\right|}{k}. \tag{6}$$

The default supervision target is

$$y(x) = \mathrm{NA}_k(x; h_0, \mathcal{H}) = \frac{1}{M-1} \sum_{m=1}^{M-1} a_k^{(0,m)}(x). \tag{7}$$

This target lies in $[0, 1]$. Higher values indicate greater agreement between the local neighborhood structures induced by the anchor and comparison models, corresponding to weaker inter-model representational disagreement.

A.4 Predictor Training

The predictor uses only the anchor representation:

$$\hat{y}(x) = g_\phi(\tilde{z}_0(x)) \,. \tag{8}$$

Comparison-model representations and neighborhoods are used to construct offline supervision and do not enter the predictor's forward pass. Except in the architecture comparison, we use the Residual FFN described in Appendix I.

The predictor minimizes

$$\mathcal{L}(\phi) = \frac{1}{|\mathcal{D}_{\mathrm{tr}}|} \sum_{x_i \in \mathcal{D}_{\mathrm{tr}}} \ell(g_\phi(\tilde{z}_0(x_i)), y(x_i)) \,. \tag{9}$$

We use a mean squared error (MSE) objective with a fixed scaling factor of $1/2$. The per-sample loss is

$$\ell(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2. \tag{10}$$

Unless otherwise specified, we use AdamW with a learning rate and weight decay of 10^{-4} and train for 10 epochs. Checkpoints are selected by validation MAE. The selected parameters, denoted by $\phi^\star$, are evaluated on the test split. Only predictor parameters are optimized.

B Common Evaluation and Visualization Procedures

B.1 Pearson Correlation

For two nonconstant sample sequences $a = (a_i)_{i=1}^n$ and $b = (b_i)_{i=1}^n$, Pearson correlation is

$$\mathrm{Corr}_{\mathrm{P}}(a, b) = \frac{\sum_{i=1}^n (a_i - \bar{a})(b_i - \bar{b})}{\sqrt{\sum_{i=1}^n (a_i - \bar{a})^2} \sqrt{\sum_{i=1}^n (b_i - \bar{b})^2}}, \tag{11}$$

where

$$\bar{a} = \frac{1}{n} \sum_{i=1}^n a_i, \qquad \bar{b} = \frac{1}{n} \sum_{i=1}^n b_i. \tag{12}$$

In the disagreement prediction experiments, let

$$y_i = y(x_i), \qquad \hat{y}_i = g_{\phi^\star}(\tilde{z}_0(x_i)). \tag{13}$$

The reported correlation is

$$r_{\mathrm{pred}} = \mathrm{Corr}_{\mathrm{P}}(\hat{y}, y), \tag{14}$$

which measures the linear association between predicted and measured neighborhood agreement.

B.2 Binned Means and Distribution Intervals

Sample-level figures display the two-dimensional density of $(y_i, \hat{y}_i)$ and group samples into bins of measured agreement. For a nonempty bin I_b, define

$$\mathcal{I}_b = \{i : y_i \in I_b\}. \tag{15}$$

The mean measured and predicted agreement within the bin are

$$\bar{y}_b = \frac{1}{|\mathcal{I}_b|} \sum_{i \in \mathcal{I}_b} y_i, \qquad \bar{\hat{y}}_b = \frac{1}{|\mathcal{I}_b|} \sum_{i \in \mathcal{I}_b} \hat{y}_i. \tag{16}$$

The binned means summarize prediction trends across measured agreement levels. Vertical intervals extend from the 5th to the 95th percentile of predictions within each bin, representing the central 90% of the prediction distribution.

C Qualitative Visualization of Inter-Model Disagreement

Figure 7 illustrates *Rashomon Representation* through high- and low-agreement queries. For the envelope and hornbill examples, CLIP, DINOv2, and ViT-21K retrieve similar neighbors, and their linear probes predict the correct class. For the flying-bird and black-swan examples, the neighborhoods diverge and the probes assign different classes. The Residual FFN, using only the CLIP representation, preserves this high–low ordering in its predicted NA_{50}.

Across ImageNet-1K and ImageNet-V2 variants, we use the ImageNet-1K training set as the shared reference set. Measured NA_{50} averages CLIP–DINOv2 and CLIP–ViT-21K Top-50 overlap; the figure shows only Top-5 neighbors. In the 10,000-query case-study cohort, all three probes are correct for 24.1% of queries with $\mathrm{NA}_{50} < 0.10$, versus 68.3% with $\mathrm{NA}_{50} \geq 0.40$, indicating an association between neighborhood organization and classification outcomes.

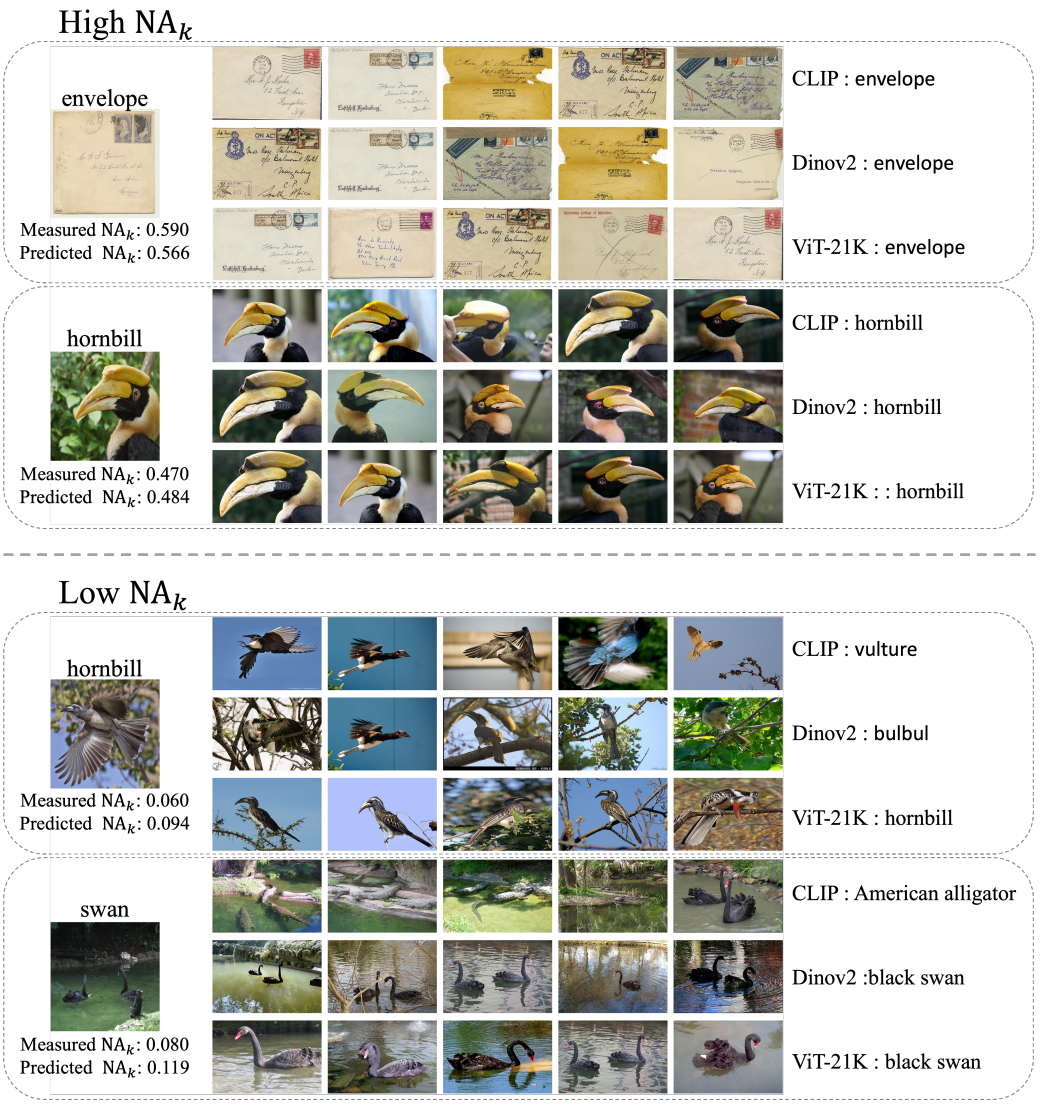

Figure 7: High (top) and low (bottom) cross-model neighborhood agreement. Rows correspond to CLIP, DINOv2, and ViT-21K.

D Predicting Disagreement from a Single Representation

This experiment tests whether inter-model disagreement can be predicted from a single model representation. We separately use CLIP, DINOv2, and ViT as anchors, with the other two models defining the supervision target.

Table 9: Configurations for prediction from a single representation.

Anchor model	Comparison models	Input dimension
CLIP	DINOv2, ViT-21K	512
DINOv2	CLIP, ViT-21K	768
ViT-21K	CLIP, DINOv2	768

A corresponding Residual FFN is trained for each configuration using only the designated anchor representation. All three configurations share the data splits, reference set, neighborhood size, and training protocol. We evaluate the association between predicted and measured agreement on the ImageNet-1K test split.

E Effect of Comparison-Model Composition

This experiment fixes DINOv2 as the anchor and uses three comparison models while varying their composition.

Table 10: Comparison-model compositions with DINOv2 fixed as the anchor.

Configuration	Comparison models
1	Swin, SigLIP 2, ConvNeXt
2	Swin, SigLIP 2, ViT-21K
3	CLIP, SigLIP 2, ConvNeXt

All configurations use the 768-dimensional DINOv2 representation as predictor input. We construct a separate supervision target for each comparison set and train a corresponding Residual FFN. The data splits, reference set, $k = 50$, and training protocol remain fixed.

F Effect of Anchor-Model Choice

This experiment examines disagreement predictability across different anchor representations. Each configuration uses three comparison models.

Table 11: Configurations for evaluating anchor-model choice.

Anchor model	Comparison models
CLIP	Swin, SigLIP 2, ConvNeXt
DINOv2	Swin, SigLIP 2, ConvNeXt
ViT-21K	Swin, SigLIP 2, ConvNeXt
Swin	DINOv2, ViT-21K, CLIP
SigLIP 2	DINOv2, ViT-21K, CLIP
ConvNeXt	DINOv2, ViT-21K, CLIP

Each predictor receives only its corresponding anchor representation. Input dimensionality varies with the anchor, while the hidden dimension remains 512. A Residual FFN is trained for each configuration using the same data splits, reference set, neighborhood size, and training protocol.

G Effect of Comparison-Set Size

This experiment fixes CLIP as the anchor and uses nested comparison sets containing two to five models.

Table 12: Nested comparison sets with CLIP fixed as the anchor.

Number of models	Comparison set
2	Swin, SigLIP 2
3	Swin, SigLIP 2, ConvNeXt
4	Swin, SigLIP 2, ConvNeXt, DINOv2
5	Swin, SigLIP 2, ConvNeXt, DINOv2, ViT-21K

For each set, the target averages pairwise neighborhood agreement between CLIP and each comparison model. All settings use the same CLIP input representations, reference set, and $k = 50$, with a separate Residual FFN trained for each target. This evaluates prediction performance as the comparison set is progressively expanded.

H Generalization under Distribution Shift

This experiment reuses the CLIP-anchor Residual FFN trained on ImageNet-1K without retraining or test-time adaptation. Measured agreement is computed using CLIP, DINOv2, and ViT with respect to the full ImageNet-1K training reference set.

The evaluation datasets match the tables in the main text:

- **External datasets:** Aircraft, Caltech101, Cars, DTD, EuroSAT, Food101, Oxford-IIIT Pets, SUN397, and UCF101.
- **Natural distribution shifts:** ImageNet-R and the ImageNet-V2 Matched Frequency, Threshold 0.7, and Top Images variants.
- **Image corruption:** ImageNet-C Gaussian Noise at severity levels 1 through 5.

The nine external datasets use their corresponding validation splits. Pearson correlation between predicted and measured agreement is computed separately for each dataset and corruption severity. The predictor and reference set remain fixed across evaluation settings.

I Predictor Architecture Comparison

I.1 Experimental Setup

We evaluate the 2-layer MLP, Residual FFN, and Transformer Encoder under the three anchor/comparison configurations in Appendix D. For a given anchor, all architectures use the same input representations, supervision targets, data splits, and training protocol. Each architecture produces a scalar agreement prediction through a sigmoid output.

Below, $\mathrm{Linear}_{a\to b}$ denotes a learnable affine map from a to b dimensions, LN denotes LayerNorm, and $\mathrm{Drop}_{0.1}$ denotes dropout with probability 0.1. Maps and normalization layers at different locations have independent parameters. Dropout is enabled only during training.

I.2 2-layer MLP

The 2-layer MLP maps the anchor representation to a 512-dimensional hidden vector and then to a scalar:

$$u = \mathrm{Drop}_{0.1}\left(\mathrm{GELU}\left(\mathrm{Linear}_{d_0\to 512}(\tilde{z}_0(x))\right)\right), \tag{17}$$

$$\hat{y}(x) = \sigma(\mathrm{Linear}_{512\to 1}(u)), \tag{18}$$

where σ denotes the sigmoid function.

I.3 RESIDUAL FFN

The input projection is

$$u^{(0)} = \mathrm{Drop}_{0.1}\left(\mathrm{GELU}\left(\mathrm{LN}\left(\mathrm{Linear}_{d_0\to 512}(\tilde{z}_0(x))\right)\right)\right). \tag{19}$$

Two residual feed-forward blocks are then applied:

$$u^{(\ell)} = u^{(\ell-1)} + F_\ell\left(\mathrm{LN}_\ell(u^{(\ell-1)})\right), \qquad \ell = 1, 2. \tag{20}$$

Each residual branch is

$$F_\ell(v) = \mathrm{Drop}_{0.1}\Big(\mathrm{Linear}_{512\to 512}\big(\mathrm{Drop}_{0.1}\big(\mathrm{GELU}(\mathrm{Linear}_{512\to 512}(v))\big)\big)\Big). \tag{21}$$

The prediction is

$$\hat{y}(x) = \sigma\left(\mathrm{Linear}_{512\to 1}\left(\mathrm{LN}(u^{(2)})\right)\right). \tag{22}$$

I.4 TRANSFORMER ENCODER

Input tokens. The anchor representation is first projected to a 512-dimensional token:

$$t(x) = \mathrm{LN}\left(\mathrm{Drop}_{0.1}\left(\mathrm{GELU}\left(\mathrm{Linear}_{d_0\to 512}(\tilde{z}_0(x))\right)\right)\right). \tag{23}$$

Let $c_{\mathrm{CLS}} \in \mathbb{R}^{512}$ be a learnable CLS token and $P \in \mathbb{R}^{2\times 512}$ be learnable positional embeddings. The initial sequence is

$$T^{(0)} = \begin{bmatrix} c_{\mathrm{CLS}}^\top \\ t(x)^\top \end{bmatrix} + P. \tag{24}$$

The encoder therefore receives one CLS token and one representation token.

Multi-head self-attention. Each layer uses eight attention heads, each with dimension $512/8 = 64$. For an input sequence S, head j computes

$$\begin{aligned} Q_j &= \mathrm{Linear}^{Q_j}_{512\to 64}(S), \\ K_j &= \mathrm{Linear}^{K_j}_{512\to 64}(S), \\ V_j &= \mathrm{Linear}^{V_j}_{512\to 64}(S). \end{aligned} \tag{25}$$

Its output is

$$A_j(S) = \mathrm{Drop}_{0.1}\left[\mathrm{softmax}\left(\frac{Q_j K_j^\top}{\sqrt{64}}\right)\right] V_j, \tag{26}$$

where softmax is applied over the key-token dimension. Head outputs are concatenated and projected:

$$\mathrm{MHSA}(S) = \mathrm{Linear}_{512\to 512}\left(\mathrm{Concat}\left(A_1(S), \ldots, A_8(S)\right)\right). \tag{27}$$

Encoder layers. Both layers use a pre-norm configuration. For $\ell = 1, 2$,

$$U^{(\ell)} = T^{(\ell-1)} + \mathrm{Drop}_{0.1}\left(\mathrm{MHSA}_\ell\left(\mathrm{LN}^{\mathrm{attn}}_\ell(T^{(\ell-1)})\right)\right), \tag{28}$$

$$T^{(\ell)} = U^{(\ell)} + \mathrm{FFN}_\ell\left(\mathrm{LN}^{\mathrm{ffn}}_\ell(U^{(\ell)})\right). \tag{29}$$

The feed-forward branch is

$$\mathrm{FFN}_\ell(v) = \mathrm{Drop}_{0.1}\Big(\mathrm{Linear}_{2048\to 512}\big(\mathrm{Drop}_{0.1}\big(\mathrm{GELU}(\mathrm{Linear}_{512\to 2048}(v))\big)\big)\Big). \tag{30}$$

Output head. The CLS output of the second encoder layer, $T^{(2)}_{\mathrm{CLS}}$, is mapped to

$$\hat{y}(x) = \sigma\left(\mathrm{Linear}_{512\to 1}\left(\mathrm{LN}(T^{(2)}_{\mathrm{CLS}})\right)\right). \tag{31}$$

J LEAVE-ANCHOR-OUT DIAGNOSTIC

This experiment fixes CLIP representations as predictor inputs and compares supervision targets that include or exclude CLIP neighborhoods. Let h_0, h_1, h_2, h_3 denote CLIP, DINOv2, ViT, and Swin, respectively. The matched control target is

$$y_{\text{anchor}}(x) = \frac{1}{3}\left[a_k^{(0,1)}(x) + a_k^{(0,2)}(x) + a_k^{(0,3)}(x)\right], \tag{32}$$

and the leave-anchor-out target is

$$y_{\text{LAO}}(x) = \frac{1}{3}\left[a_k^{(1,2)}(x) + a_k^{(1,3)}(x) + a_k^{(2,3)}(x)\right]. \tag{33}$$

Both settings use 512-dimensional CLIP inputs, a Residual FFN, the full training reference set, and $k = 50$, with the same training protocol. They differ only in the supervision target, and a corresponding predictor is trained for each.

Both targets average three pairwise agreement terms, but y_{LAO} contains no CLIP neighborhoods. We compute Pearson correlation between predictions and the corresponding measured target to assess predictability when the anchor does not participate in target construction.

K COMPUTATIONAL COST COMPARISON

This experiment compares FAISS GPU FlatIP, PyTorch matrix multiplication followed by Top-k retrieval, and a forward pass through the Residual FFN. All methods process the same 100,000 ImageNet-1K test queries individually. Direct measurement uses the full training reference set and computes neighborhood agreement across CLIP, DINOv2, and ViT with $k = 50$.

Timing starts from precomputed representations. FAISS indexes are constructed and loaded before the timed loop, and the PyTorch implementation keeps reference representations on the GPU. After warm-up, CUDA synchronization is performed before and after processing, and total elapsed time is recorded.

For n queries processed in T seconds, average per-sample time is

$$t_{\text{sample}} = \frac{1000T}{n} \quad \text{ms/sample}. \tag{34}$$

The speedup over direct measurement is

$$\text{Speedup} = \frac{T_{\text{measured}}}{T_{\text{predicted}}}. \tag{35}$$

Space costs are reported as peak GPU memory and disk storage for required assets, both in MiB. Direct measurement includes reference representations or indexes, whereas prediction storage is measured using the predictor checkpoint.

L ASSOCIATION WITH DOWNSTREAM CLASSIFICATION RISK

L.1 LINEAR CLASSIFICATION PROBES

For Aircraft, Caltech101, Cars, DTD, EuroSAT, Food101, Oxford-IIIT Pets, SUN397, and UCF101, we train separate linear probes on frozen CLIP representations using the corresponding training splits.

For a dataset with C classes, the probe outputs

$$p(x) = \text{softmax}\left(W_{\text{cls}}\tilde{z}_0(x) + b_{\text{cls}}\right), \tag{36}$$

where $W_{\text{cls}} \in \mathbb{R}^{C \times d_0}$. Ground-truth classes are denoted by c_i, distinguishing them from agreement targets y_i. The probe training objective is

$$\mathcal{L}_{\text{cls}} = -\frac{1}{n_{\text{tr}}}\sum_{i=1}^{n_{\text{tr}}} \log p_{c_i}(x_i). \tag{37}$$

The disagreement predictor is the CLIP-anchor Residual FFN trained on ImageNet-1K and remains fixed during downstream evaluation.

L.2 Multiclass Brier Score

The sample-wise multiclass Brier score is

$$B_i = \sum_{c=1}^{C} \left(p_c(x_i) - \mathbf{1}[c = c_i]\right)^2, \tag{38}$$

where $\mathbf{1}[\cdot]$ is the indicator function. Equivalently,

$$B_i = \sum_{c=1}^{C} p_c(x_i)^2 - 2p_{c_i}(x_i) + 1. \tag{39}$$

We sum over classes without dividing by the number of classes, giving $B_i \in [0, 2]$. Lower scores indicate that class probabilities are closer to the one-hot ground-truth label.

L.3 Kendall Rank Correlation between Disagreement and Brier Risk

For downstream evaluation only, agreement scores are expressed in the direction of disagreement:

$$d_i^{\text{measured}} = 1 - y_i, \qquad d_i^{\text{predicted}} = 1 - \hat{y}_i. \tag{40}$$

Predictor training continues to use the original agreement target.

We quantify the association between disagreement and per-sample Brier risk using Kendall's τ_b, which accounts for tied values. For paired sequences d and B, let C and D denote the numbers of concordant and discordant sample pairs. Let T_d and T_B denote the numbers of pairs tied only in d and only in B, respectively. Then

$$\tau_b(d, B) = \frac{C - D}{\sqrt{(C + D + T_d)(C + D + T_B)}}. \tag{41}$$

For each dataset, the two correlations are

$$\tau_{\text{measured},B} = \tau_b(d^{\text{measured}}, B), \qquad \tau_{\text{predicted},B} = \tau_b(d^{\text{predicted}}, B). \tag{42}$$

A positive value indicates that samples with higher disagreement scores tend to have higher Brier risk.

Correlations are computed independently within each of the nine datasets. For $m \in \{\text{measured}, \text{predicted}\}$, the equal-weight average is

$$\bar{\tau}_{m,B} = \frac{1}{9} \sum_{j=1}^{9} \tau_{m,B}^{(j)}. \tag{43}$$

The unrounded difference on dataset j is

$$\Delta\tau_j = \tau_{\text{predicted},B}^{(j)} - \tau_{\text{measured},B}^{(j)}. \tag{44}$$

Table 7 displays correlations to two decimal places. Its colored differences are calculated from those displayed values; the Average column is calculated from unrounded dataset-level correlations before being displayed to two decimal places.

M Effect of Reference-Set Size

This experiment fixes CLIP as the anchor, DINOv2 and ViT as comparison models, a Residual FFN predictor, and $k = 50$. Predictor training queries are fixed to 50,000 samples, denoted by $\mathcal{Q}_{\text{tr}}$.

Let $\mathcal{R}_L \subseteq \mathcal{D}_{\text{tr}}$ be a reference subset containing L samples, where

$$L \in \{5{,}000, 10{,}000, 20{,}000, 50{,}000, 100{,}000, 200{,}000, 500{,}000\}. \tag{45}$$

Reference subsets are nested and satisfy

$$\mathcal{Q}_{\mathrm{tr}} \cap \mathcal{R}_L = \varnothing. \tag{46}$$

The full-reference control uses all 1,281,167 training samples.

Let $y^{(L)}(x)$ denote supervision constructed using $\mathcal{R}_L$, and let g_{ϕ_L} denote the corresponding predictor. All predictors use the same training queries and are selected on validation data and evaluated on test data against the full-reference target $y^{(\mathrm{full})}(x)$. Defining

$$\hat{y}_i^{(L)} = g_{\phi_L}(\tilde{z}_0(x_i)), \tag{47}$$

we compute

$$r_L = \mathrm{Corr}_{\mathrm{P}}\left(\hat{y}^{(L)}, y^{(\mathrm{full})}\right). \tag{48}$$

The full-reference control also uses 50,000 training queries, and its correlation is denoted by r_{full}. Pearson retention is

$$\mathrm{Retention}(L) = \frac{r_L}{r_{\mathrm{full}}} \times 100\%. \tag{49}$$

When multiple reference subsets are evaluated at the same size, their correlations are averaged before computing the reported retention.