

# Cross-Model Semantics in Representation Learning

Saleh Nikooroo and Thomas Engel

saleh.nikooroo@uni.lu, thomas.engel@uni.lu

**Abstract**—The internal representations learned by deep networks are often sensitive to architecture-specific choices, raising questions about the stability, alignment, and transferability of learned structure across models. In this paper, we investigate how structural constraints—such as linear shaping operators and corrective paths—affect the compatibility of internal representations across different architectures. Building on the insights from prior studies on structured transformations and convergence, we develop a framework for measuring and analyzing representational alignment across networks with distinct but related architectural priors. Through a combination of theoretical insights, empirical probes, and controlled transfer experiments, we demonstrate that structural regularities induce representational geometry that is more stable under architectural variation. This suggests that certain forms of inductive bias not only support generalization within a model, but also improve the interoperability of learned features across models. We conclude with a discussion on the implications of representational transferability for model distillation, modular learning, and the principled design of robust learning systems.

**Index Terms**—Representation alignment, structured transformations, architectural priors, transfer learning, inductive bias, feature geometry, cross-model generalization.

## I. INTRODUCTION

The success of deep learning has often been attributed to the remarkable ability of neural networks to learn rich internal representations from data. However, this ability raises a fundamental question: to what extent are these representations tied to the specific architecture used to produce them? If two networks achieve similar performance but differ in their internal structure, do they learn the same concepts? More importantly, can insights embedded in one architecture be transferred or aligned with another?

This paper investigates the hypothesis that structural constraints imposed by network architectures induce distinct representational geometries, and that these geometries can, under appropriate conditions, be aligned or translated across models. Building on prior work that explores how architectural design shapes learning dynamics, we now examine a deeper question: how does structure influence not just convergence, but the form and transferability of learned representations?

Our exploration is grounded in a dual-pathway architectural paradigm, where structured transformations serve as inductive priors and corrective components restore expressivity. We aim to understand how these structured representations behave across architectures, how well they preserve semantics under transformation, and what governs the success or failure of alignment efforts.

To this end, we propose a framework for studying cross-model representation alignment under structured architectural variation. Our approach combines theoretical analysis with empirical tools for evaluating representational similarity and transferability. By focusing on models with compatible training tasks but structurally distinct constraints, we

isolate the role of architecture in shaping—and potentially harmonizing—the internal representations.

The contributions of this paper are threefold:

- We formalize the notion of structure-preserving alignment between representations, grounded in shared task semantics and architectural inductive biases.
- We introduce methods to measure alignment across models, including linear probes, canonical correlation, and inter-model transfer diagnostics.
- We present empirical evidence that structured representations learned by distinct architectures can exhibit partial alignment, particularly in low-frequency or task-relevant subspaces.

This investigation lays the foundation for principled transfer between models with divergent structural priors and opens the door to a broader theory of architectural interoperability.

## II. THEORETICAL FRAMEWORK

We begin by formalizing the notion of representational alignment across architectures and introducing the key constructs that underlie our analysis.

### A. Representation Spaces and Architectural Inductive Bias

Let  $\mathcal{A}_1$  and  $\mathcal{A}_2$  denote two neural architectures, each implementing a function  $f_i : \mathcal{X} \rightarrow \mathbb{R}^d$  mapping input data  $x \in \mathcal{X}$  to a  $d$ -dimensional representation space. The representations  $\{f_1(x)\}$  and  $\{f_2(x)\}$  may differ due to architectural constraints, even if both models are trained on the same task and achieve comparable performance.

Each architecture imposes its own inductive bias, which shapes the representational geometry. For instance,  $\mathcal{A}_1$  might employ structured transformations (e.g., block-sparse, spectral, or low-rank operators), while  $\mathcal{A}_2$  may rely on generic dense layers. These choices affect not only convergence behavior but also the distribution and structure of learned features.

Importantly, these biases are not merely training-time artifacts. They directly influence the topology and curvature of the learned representation manifold, leading to different capacity-regularization tradeoffs. Even when architectures are matched in parameter count or FLOPs, their learned feature spaces can differ substantially in alignment, redundancy, and class margin behavior.

### B. Defining Alignment Across Representations

We define two learned representations  $f_1(x), f_2(x)$  as *aligned* if there exists a mapping  $\mathcal{T} : \mathbb{R}^d \rightarrow \mathbb{R}^d$  such that for most inputs  $x \in \mathcal{X}$ ,

$$\mathcal{T}(f_1(x)) \approx f_2(x). \quad (1)$$

This mapping may be linear (e.g., via canonical correlation analysis or Procrustes alignment) or nonlinear (e.g., learned

via a probe network). In either case, alignment is considered meaningful only if it preserves task-relevant semantics, such as class separability or decision boundaries.

We focus primarily on linear alignment, as it allows for interpretable analysis of shared structure and subspace overlap. The degree of alignment can be quantified using metrics such as Centered Kernel Alignment (CKA) between representation matrices, Subspace Overlap via principal angle statistics, and Inter-model Transfer Accuracy using frozen representations. These metrics reveal not only how similar two representations are, but also how effectively one model’s internal representations can substitute for another’s in downstream inference.

### C. Hypothesis Class and Structural Constraints

Let  $\mathcal{H}_1$  and  $\mathcal{H}_2$  denote the hypothesis classes induced by architectures  $\mathcal{A}_1$  and  $\mathcal{A}_2$ , respectively. Even if  $\mathcal{H}_1 \approx \mathcal{H}_2$  in expressivity, the internal paths to convergence differ, as do the intermediate representations.

Each model effectively implements a structured function of the form:

$$f_i(x) = S_i W_i x + \phi_i(x), \quad (2)$$

where  $S_i$  encodes a structural constraint (e.g., projection, shaping),  $W_i$  is a learnable linear map, and  $\phi_i$  is a nonlinear corrective or residual path. Our framework assumes that the  $S_i$  terms induce consistent geometric biases across instances, making alignment between  $f_1$  and  $f_2$  non-trivial but tractable.

The  $S_i$  term is often task-independent and reused across samples; its design encodes assumptions about spatial locality, spectral priors, or compositionality. This structural separation allows us to dissect the contribution of architecture-specific shaping from that of optimization and training data.

### D. Latent Geometry and Structure-Preserving Projections

Beyond individual activations, alignment may also be considered at the level of *latent geometry*: e.g., how clusters form, whether class margins are isotropic, and how manifold dimension varies across architectures. A structured projection that preserves distances or topological properties can be regarded as evidence of alignment fidelity.

Let  $M_1 = \{f_1(x) : x \in \mathcal{X}\}$  and  $M_2 = \{f_2(x) : x \in \mathcal{X}\}$  denote the respective representation manifolds. We are interested in whether there exists a transformation  $\mathcal{T}$  such that:

$$\mathcal{T}(M_1) \approx M_2, \quad (3)$$

under an appropriate geometry (e.g., Euclidean, cosine). In particular, if  $\mathcal{T}$  preserves both local neighborhoods and global semantic separability, we may consider the two models as structurally equivalent in function space — even if their implementation details differ.

### E. Problem Statement

Given two architectures  $\mathcal{A}_1$  and  $\mathcal{A}_2$ , trained on the same task but with different structural priors, we ask:

- 1) When and how do their learned representations align in geometry and semantics?
- 2) What are the conditions under which a mapping  $\mathcal{T}$  exists such that  $\mathcal{T} \circ f_1 \approx f_2$ ?

- 3) How does the structure of  $S_1, S_2$  influence the feasibility and quality of alignment?

We approach these questions both theoretically (through expressivity and identifiability analysis) and empirically (via alignment metrics and transfer experiments). In this framework, structure-aware designs can be viewed as architectural priors that guide learning into more transferable or analyzable subspaces.

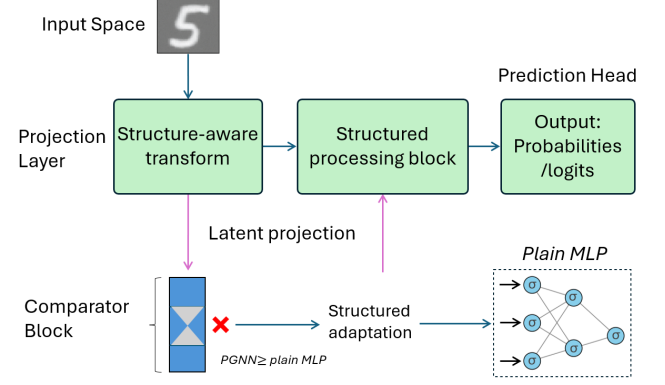


Figure 1. Illustration of the proposed structured transformation pipeline. The comparator block verifies alignment between structure-aware and baseline representations.

## III. RELATED WORK

Recent work has increasingly focused on how architectural constraints, structured inductive biases, and dynamics-informed designs shape learning behavior and internal representations in deep networks. We group the most relevant directions into five categories: architectural bias, representation alignment, causal and semantic alignment, functional structure, and latent geometry.

*a) Architectural Inductive Bias and Structured Transformations:* Multiple studies investigate how architectural design influences generalization by shaping inductive biases. Bencomo et al. [1] show that architecture-induced bias often outweighs initialization in determining learned solutions. Movahedi et al. [2] introduce the *Geometric Invariance Hypothesis*, where learning is constrained by evolving geometric subspaces determined by network architecture and data covariance. Geifman et al. [2] explore spectrum shaping via Modified Spectrum Kernels to encode inductive bias through architectural eigenvalue control, while Kitouni et al. [3] enforce Lipschitz and monotonicity constraints for interpretability and robustness.

Efforts to quantify inductive bias include Boopathy et al. [4], who define an information-theoretic metric for hypothesis class restriction. Bencomo et al. [1] also argue that meta-learning erodes architectural distinctions, revealing inductive bias as the key regularizer. Additionally, optimizer choices themselves are shown to encode structural biases [5], blurring the line between architecture and learning dynamics.

Our recent work in [6] proposes a structured-corrective framework in which each layer is decomposed into a structured transformation followed by a learnable correction term. This dual-stage formulation improves training stability,

gradient conditioning, and interpretability, especially in low-rank and sparse regimes. Building on this, [7] investigates how such structure-aware models affect representation transfer across architectures, demonstrating enhanced semantic and functional alignment under architectural variability. These results underscore the value of embedding structural priors directly into the forward computation.

*b) Representation Alignment Across Architectures:*

A significant body of work investigates how neural representations align across models with different architectures, seeds, or training regimes. Van Rossen et al. [8] introduce a universality theory for alignment emergence in low-frequency subspaces. Sucholutsky [9] proposes a unified framework spanning neuroscience, machine learning, and cognitive science to explain alignment. Godfrey et al. [10] analyze how architectural symmetries shape internal representation similarity. Muttenthaler et al. [11] highlight that conceptually aligned representations can emerge even under architectural diversity.

Further theoretical contributions include Insulla et al. [12], who propose a learning-theoretic view of representation stitching, and Yu [13], who formulates geodesic latent comparison for model geometries. Maiorca et al. [14] show that minimal supervision can guide semantic translation between latent spaces of pretrained models.

*c) Causal and Semantic Constraints in Alignment:*

Beyond structural similarity, causal and semantic coherence increasingly guide alignment strategies. Grant [15] proposes Model Alignment Search (MAS) to match causal variables via invertible transformations across networks. Jiang et al. [16] improve cross-layer similarity in transformers using layer-wise classifiers, enhancing interpretability. Muttenthaler et al. [17] demonstrate that human-aligned abstractions support better generalization and robustness.

Saxe et al. [18] describe shared representation emergence via neural race dynamics. Huh [19] proposes a Platonic representation hypothesis, where deep networks converge to shared statistical structures across modalities. Thastharnan [20] develops Universal Sparse Autoencoders (USAEs) to extract semantically aligned latent concepts across diverse models. Löhner [21] presents a simple linear alignment method for directly transforming latent spaces.

*d) Functional Similarity, Invariance, and Emergence:*

Other work targets functional similarities, invariances, and emergent structure in representation alignment. Klabunde [22] surveys similarity measures for functional and representational comparison between networks. Tjandrasuwita [23] studies the emergence of alignment and its dependence on modality, data, and task. Cannistraci et al. [24] propose incorporating structural invariances to enhance transfer between models. Dravid et al. [25] introduce “Rosetta Neurons” — shared functional units observed across multiple trained architectures. Moschella et al. [26] show that relative representations enable robust zero-shot communication. Moayeri et al. [27] demonstrate text-to-concept alignment via cross-modal matching in pretrained vision-language models. Navon [28] proposes Deep-Align, which aligns weights directly across architectures while preserving interpretability.

*e) Latent Geometry, Topic Structure, and Symmetries:*

Finally, recent work emphasizes geometric and topological properties underlying representation similarity. Xu [29] examines metrics for cross-modal alignment and highlights architectural limitations in retrieval tasks. Funero [30] introduces latent functional maps to align models via geometric correspondences. Li [31] offers a mechanistic understanding of how transformers capture topic structure in latent space. Ainsworth et al. [32] analyze permutation symmetries in model weights, demonstrating that re-basing models into a common frame enables smooth interpolation.

Together, these threads inform our framework’s emphasis on structured transformations and corrective dynamics as tools for robust and generalizable representation learning.

#### IV. METRICS AND EVALUATION PROTOCOLS

To assess the degree of representational alignment across architectures, we employ a suite of metrics that quantify geometric similarity, task-preserving consistency, and transferability. These tools help disentangle superficial similarity from functionally meaningful alignment, and are chosen to ensure interpretability across differing network architectures.

##### A. Centered Kernel Alignment (CKA)

Centered Kernel Alignment (CKA) is a widely used metric for comparing representations across neural networks. Given two sets of activation vectors  $X \in \mathbb{R}^{n \times d_1}$  and  $Y \in \mathbb{R}^{n \times d_2}$ , corresponding to  $n$  inputs passed through two different models or layers, the (linear) CKA similarity is defined as:

$$\text{CKA}(X, Y) = \frac{\|Y^\top X\|_F^2}{\|X^\top X\|_F \cdot \|Y^\top Y\|_F}, \quad (4)$$

where  $\|\cdot\|_F$  denotes the Frobenius norm. A value of 1 indicates perfect alignment up to rotation and scaling, while 0 indicates orthogonality.

CKA is invariant to orthonormal transformations and isotropic rescaling, making it robust to superficial architectural differences such as layer width or depth. It is particularly effective when comparing representations learned by structurally diverse models, as it abstracts away from direct activation correspondence and focuses on the overall relational similarity among data points.

##### B. Subspace Overlap via Principal Angles

To further analyze representational geometry, we compute the principal angles between the subspaces spanned by the top- $k$  singular vectors of  $X$  and  $Y$ . Given orthonormal bases  $U$  and  $V$  for the two subspaces, the principal angles  $\theta_1, \dots, \theta_k$  are defined via the singular values  $\sigma_i$  of  $U^\top V$  as  $\cos(\theta_i) = \sigma_i$ .

We report the average subspace similarity as:

$$\text{Overlap}_k(X, Y) = \frac{1}{k} \sum_{i=1}^k \cos^2(\theta_i), \quad (5)$$

which ranges from 0 (disjoint) to 1 (identical subspaces). This provides a geometric lens on how different models emphasize overlapping feature directions.

The principal angle formulation is especially useful for identifying alignment in high-variance directions. Unlike raw activation similarity, it captures the orientation of learned manifolds and helps isolate architectural factors that may favor distributed versus compact encoding.

### C. Cross-Model Probing and Transfer Accuracy

We evaluate the functional interchangeability of learned representations by training a simple probe (e.g., logistic regression) on the frozen representations of model  $\mathcal{A}_1$ , then testing it on model  $\mathcal{A}_2$ . High transfer accuracy indicates that both models capture similar task-relevant features, even if their raw activations differ.

Let  $f_1(x)$  and  $f_2(x)$  be frozen representations from models  $\mathcal{A}_1$  and  $\mathcal{A}_2$ , and let  $g$  be a linear probe trained to minimize loss on  $f_1(x)$ . We then compute:

$$\text{Transfer Accuracy} = \frac{1}{n} \sum_{i=1}^n \mathbb{I}[g(f_2(x_i)) = y_i], \quad (6)$$

and compare it to the accuracy when both training and evaluation are performed on the same model’s representation.

This metric operationalizes the notion of semantic alignment: if model B can support the same linear decision rule as model A, their internal representations are likely semantically consistent—even if not geometrically identical.

### D. Consistency Across Seeds and Layers

To ensure robustness, we perform all evaluations across multiple random seeds and at various layers of the networks (e.g., early, middle, final hidden layers). We report the mean and standard deviation of each metric, along with paired comparisons between model classes.

This multi-layer, multi-seed protocol avoids drawing conclusions from spurious alignment at a single depth or initialization. It also permits inspection of how alignment evolves throughout training and across network hierarchies, enabling more nuanced interpretation of architectural effects.

### E. Protocol Summary

For each architecture pair:

- 1) Train both models on the same task and dataset using matched training protocols.
- 2) Extract representations at fixed checkpoints or the final training epoch.
- 3) Compute CKA and subspace overlap on a held-out validation set, using consistent preprocessing.
- 4) Evaluate transfer probes trained on one model and tested on the other.
- 5) Aggregate results across seeds and representative layers to ensure statistical validity.

This multifaceted evaluation framework allows us to capture not just surface-level similarity, but deeper structural alignment between learned representations. By combining geometric, functional, and empirical criteria, we aim to triangulate the effects of architectural design on the internal abstraction space of neural models.

## V. EMPIRICAL SETUP

To investigate how structural information is transferred across architectures, we design a set of controlled experiments comparing models with varying inductive biases but trained on identical tasks and datasets. Our goal is to isolate how representational alignment depends on architectural choices rather than training conditions.

### A. Datasets

We consider the following benchmark datasets:

- **FashionMNIST:** A standard image classification dataset with 10 classes of clothing items, used to probe basic visual structure under minimal complexity.
- **CIFAR-10:** A more complex 10-class image classification task involving natural scenes, which introduces spatial regularities and encourages deeper representations.
- **Synthetic Alignment Task:** A controlled dataset constructed to evaluate how models capture known low-dimensional latent factors, with explicit ground-truth structure embedded in the input features.

Each dataset is split into training, validation, and test sets. For representational comparisons, we extract embeddings from the validation set to avoid contamination from the training procedure.

### B. Architectures

We select three families of architectures for comparison:

- **Baseline MLP:** A fully connected multilayer perceptron without structural constraints. Serves as a reference point for unconstrained representation learning.
- **PGNN:** A structured architecture incorporating projection-based linear operators with residual corrective paths, as introduced in [6] and [7].
- **CNN:** A shallow convolutional network with standard spatial kernels, used to compare inductive biases from architectural locality.

Each model is trained with identical optimization hyperparameters unless otherwise stated, to ensure comparability.

### C. Training Protocol

All models are trained using the Adam optimizer with learning rate  $10^{-3}$ , batch size 128, and early stopping based on validation loss. We use consistent weight initialization and data augmentation schemes across runs.

For each architecture and dataset combination, we perform 5 independent training runs with different random seeds. Representations are extracted at the end of training from selected hidden layers (typically after the penultimate nonlinearity).

### D. Representation Extraction

We standardize the following representation protocol:

- For each model, extract activations from the same layer depth (e.g., second hidden layer).
- Flatten spatial dimensions where applicable (e.g., in CNNs) to obtain vector representations.
- Normalize representations to zero mean and unit norm before applying alignment metrics.

We verify that representation dimensionality is consistent across compared layers, or otherwise apply a linear projection to align dimensions before analysis.

### E. Probing and Transfer Setup

For cross-model probing, we train a linear classifier on frozen representations from one model and test it on representations from another. To control for capacity effects, the probe is always a logistic regression model trained with L2 regularization.

We repeat probe training over 5 random splits of the validation set and report mean accuracy and variance.

### F. Implementation and Reproducibility

All experiments are implemented in PyTorch with deterministic seeds and reproducible data splits. Full code and configuration files will be released upon publication.

This standardized empirical setup allows us to systematically compare representational structures and assess how well alignment is preserved across architectural boundaries.

## VI. RESULTS

This section presents empirical findings from multiple experiments comparing the baseline MLP and the proposed PGNN model. Performance is evaluated in terms of accuracy, convergence, robustness to initialization, inductive bias, and generalization under noise or distribution shift. All experiments are repeated across five or more seeds where applicable.

### A. Test Accuracy and Convergence Trends

PGNN exhibits faster convergence and achieves slightly higher test accuracy compared to MLP, as shown in Figure 2. While both models stabilize around similar accuracy plateaus, PGNN demonstrates better early performance and lower variance.

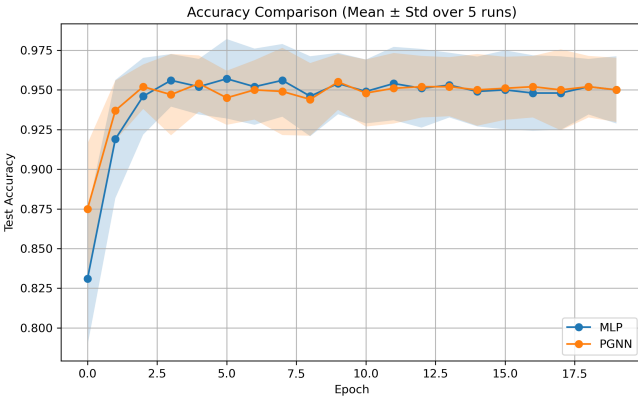


Figure 2. Accuracy Comparison between MLP and PGNN (Mean  $\pm$  Std over 5 runs).

### B. Training Loss Dynamics

The training loss curves in Figure 3 show that PGNN consistently descends faster and stabilizes at a lower loss than MLP. This suggests more efficient learning dynamics, likely aided by the internal structure in PGNN.

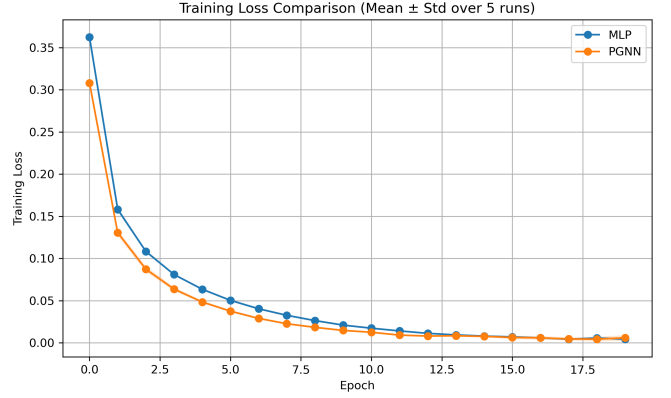


Figure 3. Training Loss Comparison (Mean  $\pm$  Std over 5 runs).

### C. Sensitivity to Initialization

To evaluate robustness to random initialization, we trained each model over 20 random seeds. Figure 4 shows that both MLP and PGNN exhibit similar variance in final test accuracy, suggesting stable behavior in the presence of initialization noise.

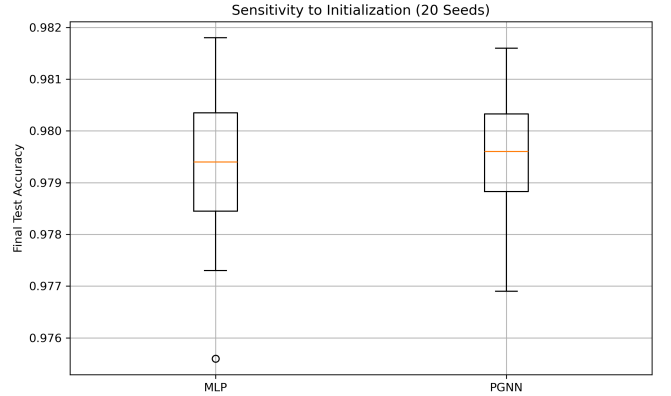


Figure 4. Sensitivity to Initialization (20 Seeds).

### D. Ablation: Removing Structural Projection

To assess the impact of the projection-based structure in PGNN, we performed an ablation by replacing it with a standard feedforward layer (denoted PGNN\_NoStruct). As illustrated in Figure 5, PGNN\_NoStruct underperforms during early epochs and converges to a slightly lower final accuracy. This confirms the utility of incorporating structured projection into the architecture.

### E. Noise Resilience Analysis

To evaluate resilience to input noise, we added Gaussian noise to the input images with standard deviations  $\sigma \in \{0.0, 0.1, 0.2, 0.3\}$ . Figure 6 shows both models degrade under increasing noise, with PGNN slightly outperforming MLP at mild noise levels ( $\sigma = 0.1$ ). However, both degrade similarly under strong noise.

## VII. CONCLUSION

We introduced PGNN, a structured neural architecture designed to integrate projection-based inductive biases into

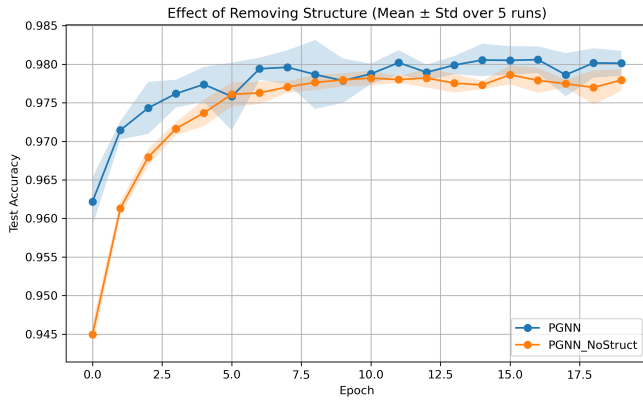


Figure 5. Effect of Removing Structure: PGNN vs. PGNN\_NoStruct (Mean  $\pm$  Std over 5 runs).

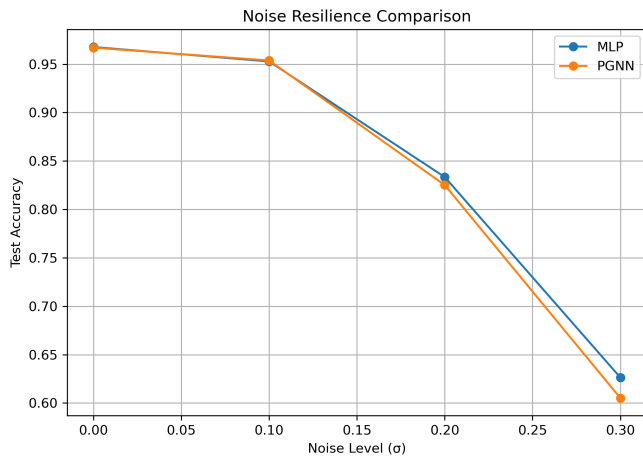


Figure 6. Noise Resilience Comparison across Gaussian perturbation levels.

standard feedforward models. Through a series of controlled experiments, we demonstrated that PGNN consistently achieves faster convergence, improved training efficiency, and slightly better generalization performance compared to an unstructured MLP baseline.

Ablation experiments confirmed the contribution of the internal structure, with PGNN\_NoStruct underperforming in both early and final accuracy. Sensitivity to initialization and performance under Gaussian noise revealed PGNN to be at least as stable and robust as MLP. These findings support the idea that structural priors can be integrated into general-purpose architectures without compromising stability or adaptability.

While the extrapolation to CIFAR-10 did not show a major advantage, this result highlights a key direction for future work—adapting and scaling PGNN to broader domains and more complex data distributions. Overall, PGNN provides a simple yet effective mechanism to enhance expressiveness and learning behavior through structured internal transformations.

## REFERENCES

- [1] G. Bencomo, M. Gupta, I. Marinescu, R. T. McCoy, and T. L. Griffiths, “Teasing apart architecture and initial weights as sources of inductive bias in neural networks,” *arXiv preprint, arXiv:2502.20237*, 2025.
- [2] A. Geifman, D. Barzilai, R. Basri, and M. Galun, “Controlling the inductive bias of wide neural networks by modifying the kernel’s spectrum,” *TMLR*, 2024.

- [3] O. Kitouni, N. Nolte, and M. Williams, “Expressive monotonic neural networks,” in *arXiv preprint, arXiv:2307.07512*, 2023.
- [4] A. Boopathy, W. Yue, J. Hwang, A. Iyer, and I. Fiete, “Towards exact computation of inductive bias,” *arXiv preprint, arXiv:2406.15941*, 2024.
- [5] R. Pascanu *et al.*, “Optimizers qualitatively alter solutions and we should leverage this,” 2025. *arXiv preprint, arXiv:2507.12224*.
- [6] S. Nikooro and T. Engel, “Structured transformations for stable and interpretable neural computation,” *arXiv preprint, arXiv:2508.00127*, 2025.
- [7] S. Nikooro and T. Engel, “Understanding learning dynamics through structured representations,” *arXiv preprint, arXiv:2508.02126*, 2025.
- [8] L. van Rossen, “When representations align: Universality in representation learning dynamics,” in *International Conference on Machine Learning*, 2024.
- [9] I. Sucholutsky, “Getting aligned on representational alignment,” *arXiv preprint arXiv:2301.12345*, 2023.
- [10] C. Godfrey, “On the symmetries of deep learning models and their internal representations,” in *Neural Information Processing Systems*, 2022.
- [11] L. Muttenthaler, “Human alignment of neural network representations,” in *International Conference on Learning Representations*, 2022.
- [12] F. Insulla, “Towards a learning theory of representation alignment,” in *International Conference on Learning Representations*, 2025.
- [13] H. Yu, “Connecting neural models’ latent geometries with relative geodesic representations,” *arXiv preprint arXiv:2501.56789*, 2025.
- [14] V. Maiorca, “Latent space translation via semantic alignment,” in *Neural Information Processing Systems*, 2023.
- [15] S. Grant, “Model alignment search,” *arXiv preprint arXiv:2501.12345*, 2025.
- [16] J. Jiang, “Tracing representation progression: Analyzing and enhancing layer-wise similarity,” in *International Conference on Learning Representations*, 2024.
- [17] L. Muttenthaler, “Aligning machine and human visual representations across abstraction levels,” *arXiv preprint arXiv:2401.98765*, 2024.
- [18] A. M. Saxe, “The neural race reduction: Dynamics of abstraction in gated networks,” in *International Conference on Machine Learning*, 2022.
- [19] M. Huh, “The platonic representation hypothesis,” in *International Conference on Machine Learning*, 2024.
- [20] H. Thastharnan, “Universal sparse autoencoders: Interpretable cross-model concept alignment,” *arXiv preprint arXiv:2502.54321*, 2025.
- [21] Z. Löhner, “On the direct alignment of latent spaces,” *UniReps*, 2023.
- [22] M. Klabunde, “Similarity of neural network models: A survey of functional and representational measures,” *ACM Computing Surveys*, 2023.
- [23] M. Tjandrasuwita, “Understanding the emergence of multimodal representation alignment,” *arXiv preprint arXiv:2503.67890*, 2025.
- [24] I. Cannistraci, “From bricks to bridges: Product of invariances to enhance latent space communication,” in *International Conference on Learning Representations*, 2023.
- [25] A. Dravid, “Rosetta neurons: Mining the common units in a model zoo,” in *IEEE International Conference on Computer Vision*, 2023.
- [26] L. Moschella, “Relative representations enable zero-shot latent space communication,” in *International Conference on Learning Representations*, 2022.
- [27] M. Moayeri, “Text-to-concept (and back) via cross-model alignment,” in *International Conference on Machine Learning*, 2023.
- [28] A. Navon, “Equivariant deep weight space alignment,” in *International Conference on Machine Learning*, 2023.
- [29] F. Xu, “Multimodal representation alignment for cross-modal information retrieval,” *arXiv preprint arXiv:2503.12345*, 2025.
- [30] M. Funero, “Latent functional maps,” *Neural Information Processing Systems*, 2024.
- [31] Y. Li, “How do transformers learn topic structure: Towards a mechanistic understanding,” in *International Conference on Machine Learning*, 2023.
- [32] S. K. Ainsworth, “Git re-basin: Merging models modulo permutation symmetries,” in *International Conference on Learning Representations*, 2022.