UNIVERSAL IN-CONTEXT LEARNING OVER FOUNDATION MODEL EMBEDDINGS

Anonymous authors
Paper under double-blind review

ABSTRACT

Foundation-model embeddings are widely used across many domains, yet using them for prediction usually requires training a separate classifier for each dataset. This adds engineering work, makes results sensitive to hyperparameter choices, and increases the risk of overfitting when labeled data are scarce. It also means that embedding benchmarks measure classifier optimization as well as representation quality. Meanwhile, tabular foundation models have shown that transformers can classify arbitrary tabular data in context, in a single forward pass and without task-level training. We adapt a tabular foundation model into a single in-context predictor that works across embedding spaces. Instead of training a separate head for each task, we meta-train one shared model across tasks and encoders. At every training step, we apply a freshly sampled random orthogonal projection that changes the coordinate basis of the embeddings while preserving their dimension and geometry, which prevents the model from relying on any fixed coordinate orientation. At inference, the model averages its predictions over multiple random orthogonal projections of the same inputs and requires no parameter updates. We curate a benchmark spanning 7 domains, 29 encoders, 75 tasks, 308 task–encoder pairs, and 169,554 samples, and evaluate it under in-distribution, novel-task, novel-encoder, and joint task–encoder transfer. Averaged over all 308 pairs, our model reaches 0.8886 ROC-AUC, improving on the strongest baseline by 0.0198 (95% CI $[0.0130, 0.0274]$) without a single gradient step at inference. We release all code, benchmark data, and trained model checkpoints.

1 INTRODUCTION

Foundation-model embeddings distill complex inputs into vector representations learned through large-scale pretraining (Reimers & Gurevych, 2019; Radford et al., 2021; Lin et al., 2023). By making learned semantic and structural information reusable, they allow a single pretrained encoder to support many downstream applications without end-to-end retraining (Oquab et al., 2024; Zhai et al., 2023; Elnaggar et al., 2023). Embeddings now support retrieval, similarity search, clustering, and prediction across language, vision, audio, molecules, proteins, graphs, and time series (Baevski et al., 2020; Ross et al., 2022; Ansari et al., 2024).

Embeddings, however, are representations rather than predictive models. Using them for classification generally requires fitting a separate downstream classifier for every dataset–encoder combination, commonly a regularized linear probe, shallow MLP, tree ensemble, or nearest-neighbor model (Alain & Ollivier, 2016; Kornblith et al., 2019; Radford et al., 2021; Oquab et al., 2024). At scale, this requires many separate training and tuning pipelines. It also makes measured representation quality depend on the downstream model, optimizer, and hyperparameters. These choices can change encoder rankings and increase overfitting when data are scarce (Hewitt & Liang, 2019; Voita & Titov, 2020; Belinkov, 2022; Kumar et al., 2022).

These limitations raise a central question: can one predictor operate across heterogeneous foundation-model embedding spaces and learn a new classification task directly from labeled examples, without parameter updates at inference?

Tabular foundation models provide a natural starting point for such a predictor. Models such as TabPFN (Hollmann et al., 2022; 2025) and TabICL (Qu et al., 2025) treat labeled datasets as context and predict query labels in a single forward pass. These models already accommodate datasets that

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.

vary in both the number and meaning of their features. Embeddings create a different challenge. A tabular feature often has a clear meaning on its own, but an individual embedding coordinate usually does not. Its meaning comes from the full vector and the geometry of the embedding space. A universal predictor must therefore use relationships among complete vectors while handling spaces with different dimensions and coordinate systems.

We address this challenge by adapting a tabular foundation model across heterogeneous embedding spaces. The predictor accepts the encoders' different embedding dimensions directly. Within each episode, we apply the same random orthogonal basis change to support and query embeddings, retaining their full dimensionality and pairwise geometry. Resampling this transformation throughout training discourages reliance on any fixed coordinate orientation, while averaging predictions across independent transformations improves stability at inference. The resulting model uses labeled examples as context to classify embeddings from new tasks and encoders without task-specific gradient updates.

Our contributions are threefold. First, we formulate prediction over foundation-model embeddings as a universal in-context learning problem and train one shared predictor that replaces task-specific heads; it supports embeddings of different dimensions and transfers across unrelated coordinate systems through randomly resampled orthogonal basis changes, which remove coordinate identity without discarding information. Second, we construct a benchmark of 75 tasks, 29 encoders, and 308 task–encoder pairs across 7 domains, with controlled evaluation of unseen-task, unseen-encoder, and joint transfer. Third, our model outperforms standard downstream heads in every transfer regime without gradient updates at inference, and ablations show that per-step basis resampling drives transfer to unseen encoders. We release code, checkpoints, and the full embedding benchmark.

2 Related Work

Foundation-model embeddings. Large-scale pretraining produces reusable vector representations that capture different forms of structure across data domains. In language, contextual and contrastive encoders organize syntactic and semantic information in dense representation spaces (Reimers & Gurevych, 2019; Radford et al., 2021; Xiao et al., 2023); in vision, self-distilled and vision–language models encode visual semantics and scene structure (Caron et al., 2021; Oquab et al., 2024; Zhai et al., 2023). Masked and contrastive objectives similarly yield representations of acoustic content in audio and speech (Baevski et al., 2020; Chen et al., 2022; Elizalde et al., 2023), chemical structure in molecular sequences (Ross et al., 2022; Ahmad et al., 2022), and evolutionary and biophysical structure in proteins (Rives et al., 2021; Lin et al., 2023; Elnaggar et al., 2023). Beyond these domains, message-passing networks summarize graph topology and node interactions (Kipf & Welling, 2017; Xu et al., 2019; Veličković et al., 2018), while convolutional transforms and pretrained sequence models capture temporal patterns in time series (Dempster et al., 2020; 2021; Ansari et al., 2024). Although these methods differ in architecture, training objective, dimension, and domain, they produce the same type of output: a vector representation that must be converted into a task-specific prediction.

Tabular in-context learning. Prior-data fitted networks (PFNs) learn from many synthetic datasets and perform Bayesian inference in a single transformer forward pass (Müller et al., 2022). TabPFN (Hollmann et al., 2022; 2025) and TabICL (Qu et al., 2025) extend this approach to real-world tabular classification. They process the labeled examples and queries together without performing gradient updates for each new task. These models already handle datasets whose feature counts and meanings vary across tasks. Embeddings create a different challenge: their coordinates change across encoders, and an individual coordinate usually has no independent meaning because information is distributed across the full vector. Recent work applies frozen tabular foundation models to perceptual representations (Zhang et al., 2026), but does not train one predictor specifically to transfer across different embedding spaces. Our approach uses parameter-efficient adaptation across seven domains. It directly accepts the number of features produced by each encoder and learns across changing orthogonal coordinate bases.

Adapting foundation representations. Standard probing methods train a separate prediction head for each task, most commonly an ℓ_2-regularized linear probe (Alain & Ollivier, 2016; Conneau & Kiela, 2018; Radford et al., 2021; Muennighoff et al., 2023), a shallow multi-layer perceptron (MLP)

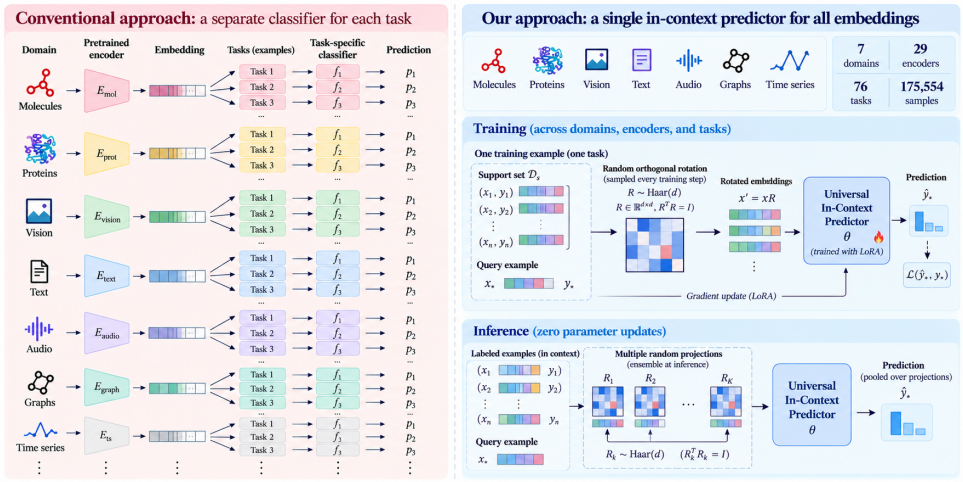


Figure 1: **Universal in-context prediction over foundation-model embeddings.** Left: Standard methods train separate classifiers per task–encoder pair. Right: We train a single predictor across 7 domains, 29 encoders, and 75 tasks. By applying random orthogonal basis changes and updating only shared LoRA parameters during training, our model classifies novel embeddings at inference (averaged over orthogonal views) without task-specific tuning.

(Wu et al., 2018; Rao et al., 2019; Yang et al., 2021; Xu et al., 2022), or a nearest-neighbor classifier (Oquab et al., 2024). Their results can depend strongly on the head architecture, optimizer, and hyperparameters, which makes comparisons of representation quality less reliable (Hewitt & Liang, 2019; Voita & Titov, 2020; Belinkov, 2022). This dependence is especially problematic when labeled data are limited, because task-specific training can overfit (Altae-Tran et al., 2017; Huang et al., 2021). Gradient updates can also change the representation geometry under distribution shift (Kumar et al., 2022). Encoder-specific adapters reduce the number of trained parameters but still require a new fitting procedure for each encoder (Tajjar et al., 2026). We instead adapt one in-context learner with shared LoRA parameters (Hu et al., 2022), allowing prediction on new tasks and encoders without additional fitting.

3 Methodology

3.1 Problem Formulation

We consider a collection of classification tasks $\tau \in \mathcal{T}$ and embedding encoders $e \in \mathcal{E}$. For a task–encoder pair (τ, e), encoder e represents each input in $\mathbb{R}^{d_e}$. An episode consists of a labeled support set (X_s, y_s) and a query set (X_q, y_q), where $X_s \in \mathbb{R}^{n_s \times d_e}$, $X_q \in \mathbb{R}^{n_q \times d_e}$, and the labels belong to the task-specific class set $\mathcal{Y}_\tau$. The support and query examples share an encoder within an episode, but the embedding dimension and coordinate system may change between episodes.

Our objective is to learn a single conditional predictor

$$p_\theta(y_q \mid X_s, y_s, X_q) \tag{1}$$

that transfers across tasks and encoders. At inference, τ, e, or both may be absent from meta-training. In every case, the model must infer the task from the labeled support set without updating θ or fitting parameters specific to the task or encoder.

This setting differs from conventional tabular in-context learning in one important respect. Individual embedding coordinates do not carry stable, independent semantics: information is distributed across the full vector, and different encoders need not share either a dimension or a coordinate basis. A universal predictor should therefore avoid relying on fixed coordinate identities while retaining the within-space structure that relates support and query examples.

3.2 Randomized Orthogonal Basis Changes

For an encoder producing d_e-dimensional embeddings, we draw a square orthogonal matrix by applying QR decomposition to a Gaussian matrix. Writing this matrix as $R \in \mathbb{R}^{d_e \times d_e}$, we compute

$$Z_s = X_s R, \qquad Z_q = X_q R. \tag{2}$$

The same R is applied to support and query examples within an episode. Since $R^\top R = I$, this basis change retains the full representation and preserves inner products and pairwise Euclidean distances. The tabular predictor accepts different embedding widths directly; no padding, dimensionality reduction, or encoder-specific adapter is required.

At every optimization step, we sample a new matrix R. The embedding information is preserved, but its coordinate system changes. This encourages the predictor to use relationships between support and query embeddings instead of depending on a fixed orientation. Section 5.4 compares fixed, pooled, lossy, non-orthogonal, and resampled orthogonal transformations to isolate the source of the gain.

3.3 Cross-Embedding In-Context Adaptation

We instantiate p_θ with the pretrained TabICL architecture (Qu et al., 2025). Given (Z_s, y_s, Z_q), TabICL processes the labeled support set as context and returns query-class logits in a single forward pass. We freeze the pretrained backbone and optimize one shared set of LoRA parameters (Hu et al., 2022) across all meta-training episodes. No parameters are associated with an individual task, encoder, or domain.

At each optimization step, we sample a training pair (τ, e), construct its support and query sets, draw a fresh orthogonal basis change R, and minimize query cross-entropy:

$$\min_\theta \; \mathbb{E}_{(\tau,e),R} \left[\mathrm{CE}(\mathcal{M}_\theta(X_s R, y_s, X_q R), y_q) \right]. \tag{3}$$

Here, θ denotes only the shared adapter parameters and $\mathcal{M}_\theta$ returns logits over the classes present in the episode. The held-out test set of every task is excluded from both optimization and checkpoint selection and is used only for final evaluation. The model is therefore trained once across many embedding spaces instead of being retrained for every task–encoder pair.

3.4 Inference by Orthogonal-View Ensembling

For a new task–encoder pair, the learned parameters remain fixed. We draw K independent orthogonal transformations, evaluate the same support and query sets under each resulting view, and average the predicted class probabilities:

$$\bar{p}_K(y \mid x_q, X_s, y_s) = \frac{1}{K} \sum_{k=1}^{K} \left[\mathrm{softmax}(\mathcal{M}_\theta(X_s R_k, y_s, x_q R_k)) \right]_y. \tag{4}$$

We use $K = 16$ unless stated otherwise. Averaging predictions across these views reduces variation caused by the randomly selected coordinate basis. Conditional on a fixed model, support set, and query, the variance of the sample mean decreases as $1/K$ for independent transformation draws; this argument concerns variance rather than model bias (Appendix B). Inference requires only forward passes and introduces no gradient updates, checkpoint selection, or learned transformation for the new pair. Algorithms 1–2 provide complete pseudocode.

3.5 Implementation Summary

LoRA adapters are applied to the feed-forward layers of TabICL's column embedder, row interactor, and in-context predictor. We use rank $r = 8$, scale $\alpha = 32$, and 467,992 trainable parameters; all backbone parameters remain frozen. Each episode uses the complete training set as context. Training uses gradient checkpointing and bf16 autocast on a single 47 GB A40 GPU. Appendix C reports the remaining implementation details.

4 BENCHMARK SUITE: CROSS-DOMAIN TASKS AND EMBEDDING ENCODERS

We construct a cross-domain benchmark for both meta-training and evaluation. It contains 75 classification tasks, 29 embedding encoders ($d \in [256, 1280]$; Table 1), 308 task–encoder pairs, and 169,554 samples across 7 domains. The datasets come from established public sources, and publicly available encoders convert the raw inputs into fixed vector representations. Appendix D documents the complete benchmark inventory.

4.1 DATA MODALITIES, REPRESENTATIONS, AND TASK FORMULATION

Table 1 summarizes the task collections and encoder families in each domain. Appendix D lists every task, including its sample count, number of classes, and class balance. It also lists every encoder, including its checkpoint, architecture, and embedding dimension.

Table 1: Domains, task collections, and representation encoders in the benchmark. Appendix D provides the complete task- and encoder-level specifications.

Domain	Task collections	Representation encoders
Molecules	10 binary ADME and toxicity assays from TDC, represented as canonical SMILES (Huang et al., 2021).	Five SMILES Transformer encoders from the ChemBERTa family and SMILES-BERT (Wang et al., 2019; Ahmad et al., 2022).
Proteins	15 tasks covering localization, bioactivity, expression, stability, binding, solubility, structural class, and antibody developability (Lhoest et al., 2021; Huang et al., 2021).	Four protein-language-model instances from ESM-2, ProtBERT, and Ankh (Elnaggar et al., 2021; Lin et al., 2023; Elnaggar et al., 2023).
Vision	10 natural- and medical-image tasks, including CIFAR-10, SVHN, Beans, and MedMNIST v2 (Yang et al., 2023).	DINOv2, SigLIP, MetaCLIP, ViT, and BEiT (Dosovitskiy et al., 2021; Oquab et al., 2024; Zhai et al., 2023).
Text	11 topic, sentiment, emotion, hate-speech, irony, and offensive-language tasks (Lhoest et al., 2021).	BGE, BGE-M3, Arctic-Embed, and Nomic Embed (Xiao et al., 2023).
Audio	7 tasks covering speech emotion, environmental sound, music genre, and language identification (Lhoest et al., 2021).	WavLM, Whisper, CLAP, and AST (Chen et al., 2022; Radford et al., 2023; Elizalde et al., 2023).
Graphs	10 molecular and macromolecular classification tasks from TUDataset (Morris et al., 2020; Fey & Lenssen, 2019).	GIN, GCN, GAT, and GraphSAGE with graph-level pooling (Xu et al., 2019; Kipf & Welling, 2017; Veličković et al., 2018; Hamilton et al., 2017).
Time series	12 ECG, motion, sensor, telemetry, shape, and spectrography tasks from UCR/UEA (Dau et al., 2019; Middlehurst et al., 2024).	MiniRocket, Rocket, and Chronos-T5 (Dempster et al., 2020; 2021; Ansari et al., 2024).

4.2 EVALUATION PROTOCOLS, METRICS, AND GENERALIZATION SPECTRUM

Data partitioning. Our evaluation uses two distinct levels of data partitioning. At the meta-learning level, we divide task–encoder pairs into meta-training, meta-validation, and meta-test sets for each run. Meta-training pairs provide episodes for gradient-based adaptation, meta-validation pairs are used for checkpoint selection, and meta-test pairs are reserved for evaluating transfer. Separately, the examples within each task are divided into training, validation, and test sets. We retain official dataset splits when available and otherwise, we apply the standard splitting protocol for its domain. The training examples provide the support set, while validation examples from meta-training pairs provide query labels for the meta-training objective. Test examples are excluded from both gradient updates and checkpoint selection. We distinguish generalization to new examples from generalization to new tasks and encoders. For each evaluated pair, its regime is determined by whether its task and encoder occur in the meta-training set: In-Distribution includes both a previously observed task and encoder; Novel Task includes a held-out task and an observed encoder; Novel Encoder includes

Table 2: Mean ROC-AUC by task–encoder generalization regime. Δ is the paired difference between our model and the strongest baseline, with pointwise 95% bootstrap confidence intervals. Here n counts run-level task–encoder evaluations across the three runs.

Regime	**Ours**	Δ	**95% CI**	n
In-Distribution	0.8891	$+0.0207$	$[+0.0138, +0.0286]$	581
Novel Task	0.8862	$+0.0167$	$[+0.0065, +0.0289]$	130
Novel Encoder	0.8893	$+0.0203$	$[+0.0121, +0.0294]$	174
Joint Generalization	0.8872	$+0.0140$	$[+0.0064, +0.0225]$	39

an observed task and a held-out encoder; and Joint Generalization includes both a held-out task and encoder. These labels describe exposure during gradient-based meta-training. We repeat the experiment across three independently generated partitions. Consequently, a pair's generalization regime may differ across runs. Table 9 provides the partition assignments and seeds, and reports each pair's result and regime separately for every run. We report ROC-AUC for each task–encoder pair and average pair-level scores within each regime; for multi-class tasks, ROC-AUC is macro-averaged using a one-vs-rest formulation.

Statistical uncertainty. Tasks and encoders vary in domain, sample size, class balance, and difficulty. Therefore, the spread of raw AUC scores mainly reflects differences among evaluations rather than uncertainty in the comparison between methods. We instead compute the paired AUC difference for each task–encoder evaluation before aggregation. To construct 95% confidence intervals, we use a paired two-way bootstrap over the task and encoder dimensions. Analyses by regime are stratified by domain, and the strongest baseline is selected again in each bootstrap sample. The resulting intervals measure uncertainty in the mean paired improvement across both tasks and encoders.

Baselines. We compare against five established approaches spanning standard choices for prediction from frozen representations. Linear probing measures linearly accessible information (Alain & Ollivier, 2016; Kornblith et al., 2019); k-NN measures local representation geometry (Caron et al., 2021; Oquab et al., 2024); a shallow MLP tests a task-specific nonlinear head (Wu et al., 2018; Rao et al., 2019); and XGBoost provides a strong tree-ensemble baseline for fixed vector features (Chen & Guestrin, 2016). We also evaluate pretrained TabICL without adaptation (Qu et al., 2025). Appendix C gives full fitting procedures.

5 Empirical Results & Analysis

We organize the empirical analysis around overall performance, generalization regimes and domains, variation across tasks and encoders, basis-randomization and inference ablations, and support-set scaling. Appendix E reports the complete pair-level results.

5.1 Overall Performance

Across all 308 task–encoder pairs, our model achieves a mean ROC-AUC of 0.8886, compared with 0.8689 for MLP, the strongest baseline overall. The paired improvement is $+0.0198$ AUC with a 95% confidence interval of $[+0.0130, +0.0274]$. The ordering is consistent across the remaining baselines: zero-shot TabICL reaches 0.8615, XGBoost 0.8607, the linear probe 0.8556, and k-NN 0.8325. Thus the aggregate advantage is not tied to a single weak comparator.

5.2 Generalization Regimes and Domains

Table 2 breaks performance down by task–encoder generalization regime. Mean ROC-AUC varies only from 0.8862 to 0.8893 across the four regimes. The paired advantage over the strongest baseline is positive in every regime, ranging from $+0.0140$ for Joint Generalization to $+0.0207$ for In-Distribution, and every 95% confidence interval excludes zero. Joint Generalization therefore remains competitive even when both the task and encoder are absent from meta-training.

Table 3: Mean ROC-AUC by domain. Δ is the paired difference between our model and the strongest baseline, with pointwise 95% confidence intervals. n counts task–encoder evaluations.

Domain	**Ours**	Δ	**95% CI**	n
Molecules	0.8626	$+0.0156$	$[+0.0104, +0.0222]$	150
Proteins	0.8932	$+0.0070$	$[+0.0035, +0.0109]$	180
Vision	0.9038	$+0.0094$	$[+0.0024, +0.0178]$	150
Text	0.8947	$+0.0047$	$[+0.0017, +0.0076]$	132
Audio	0.9199	$+0.0036$	$[-0.0021, +0.0087]$	84
Graphs	0.7999	$+0.0383$	$[+0.0206, +0.0525]$	120
Time Series	0.9631	$+0.0031$	$[-0.0026, +0.0110]$	108

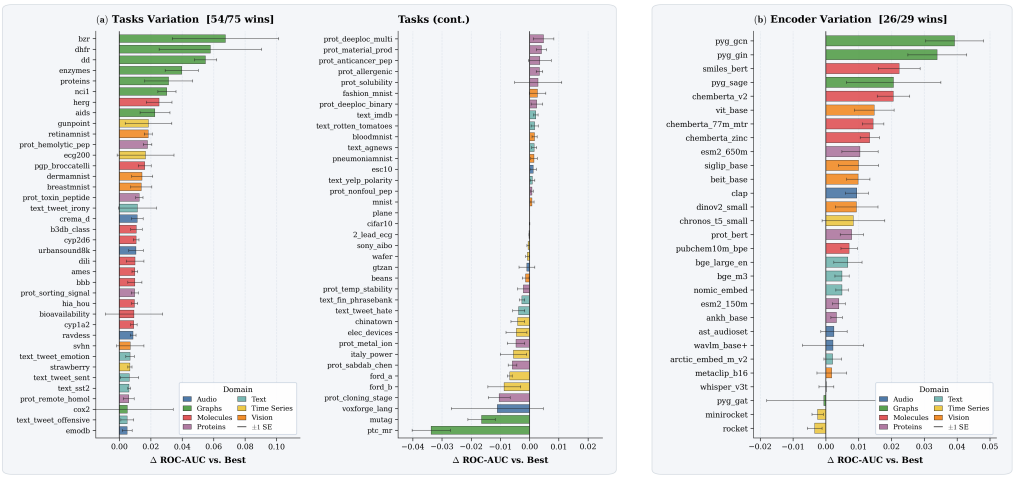


Figure 2: **Variation across tasks and encoders.** Bars show the mean paired ROC-AUC difference between our model and the strongest baseline for each task (a, left) and encoder (b, right), sorted by effect size and colored by domain. Error bars show one standard error. Our model has a positive mean difference on 54 of 75 tasks and 26 of 29 encoders.

Table 3 reports the corresponding domain breakdown. Our model has a positive mean advantage in all seven domains. The interval excludes zero for graphs, molecules, proteins, text, and vision; audio and time series remain positive by point estimate but inconclusive. These results show broad but non-uniform gains across representation families.

5.3 Variation Across Tasks and Encoders

Aggregating by task and encoder reveals where the overall improvement is concentrated (Figure 2). The model wins on 54 of 75 tasks; 33 task-level confidence intervals exclude zero in the positive direction, while seven exclude zero in the negative direction. At the encoder level, 26 of 29 point estimates are positive and 17 intervals exclude zero; none of the negative encoder-level point estimates has an interval excluding zero. The aggregate result is therefore broad across encoders but heterogeneous at task level.

5.4 Basis Randomization and Inference Ablations

Our method combines three ingredients: LoRA meta-training, a random orthogonal basis change resampled at every training step, and averaging over random orthogonal views at inference. The ablations ask which of these is responsible for transfer to unseen tasks and encoders. We report ROC-AUC separately on *seen* pairs (In-Distribution) and *unseen* pairs (Novel Task, Novel Encoder, and Joint Generalization, pooled), and summarize memorization by the *seen–unseen gap*; a model that transfers well has a small gap. Full protocols and results are given in Appendix C.4.

Resampled bases improve transfer relative to raw-input training. Training on raw embeddings performs similarly to our method on seen pairs (0.8900 vs. 0.8891), but achieves lower ROC-AUC on unseen pairs (0.8690 vs. 0.8879). The corresponding seen–unseen gaps are 0.0210 and 0.0012. These results are consistent with basis randomization reducing dependence on encoder-specific coordinate structure, although the performance gap alone does not establish the mechanism responsible for the difference.

Rotating once is not enough; the basis must change. Applying a single fixed orthogonal transformation to all encoders behaves like raw-input training (unseen 0.8685, gap 0.0212). A fixed rotation is merely another fixed coordinate system, which the model can memorize just as easily. Giving each encoder its own fixed transformation helps (unseen 0.8745), but the gap remains large (0.0150): each encoder is still always presented in the same basis, so the model can learn encoder-specific coordinates.

More bases yield better transfer. Sampling from a fixed pool of 4, 16, or 64 random orthogonal bases raises unseen ROC-AUC from 0.8730 to 0.8785 and 0.8830, while the gap shrinks from 0.0165 to 0.0108 and 0.0062. Resampling at every step is the limit of this trend and performs best. The per-encoder condition falls between the 4- and 16-basis pools despite exposing more bases, indicating that what matters is that the basis cannot be predicted from encoder identity.

Orthogonality matters among resampled transformations. Permutations with sign flips and a deterministic PCA basis leave coordinate structure largely intact and transfer poorly. A lossy Johnson–Lindenstrauss projection achieves a small gap but lowers ROC-AUC on all pairs (0.8714 overall), showing that invariance alone is insufficient when information is discarded. A full-dimensional Gaussian transformation is resampled and full rank, but distorts distances and reaches 0.8800 on unseen pairs, below 0.8879. Random orthogonal transformations are the only tested option that is resampled, full rank, and exactly preserves inner products and Euclidean distances, and they perform best.

Basis robustness is learned, and orthogonal views exploit it. Table 5 shows that a single random orthogonal transformation at test time reduces ROC-AUC by 0.0190 for the raw-trained model and 0.0090 for the per-encoder model, but by only 0.0026 for ours. The best inference protocol also depends on training. Our model gains from orthogonal views (0.8886 with 16 views vs. 0.8770 with 16 permutations). The raw-trained model shows the reverse (0.8680 vs. 0.8822), because rotations destroy the coordinates it memorized.

Orthogonal views alone are not sufficient. Applied to frozen TabICL, orthogonal-view averaging raises ROC-AUC only from 0.8615 to 0.8659, still below the MLP baseline (0.8689). Meta-training with resampled bases raises it to 0.8886 overall and from 0.8668 to 0.8879 on unseen pairs.

Training and inference contribute jointly. Table 5 shows that our strongest result depends on the combination of basis-randomized training and orthogonal-view ensembling. For our trained model, increasing from one random orthogonal view to 16 views raises overall ROC-AUC from 0.8718 to 0.8886. Under native-input inference, the same model scores 0.8744, compared with 0.8790 for raw-input training. Conversely, applying 16 orthogonal views to frozen TabICL achieves only 0.8659. We therefore interpret the results as evidence for the combined procedure: meta-training improves the model's ability to use changing bases, and inference-time averaging converts predictions across those bases into a stronger final predictor.

5.5 Scaling with Support-Set Size

Figure 3 evaluates how predictive performance changes as the maximum labeled support-set size increases from 32 to 4,096 examples. This analysis is restricted to unseen tasks. For a requested support cap (m), a task with (n_{train}) available training examples contributes $(\min(m, n_{\text{train}}))$ support examples. Tasks with fewer than (m) available examples therefore use their complete training set. The horizontal axis represents a maximum support size rather than an identical number of examples for every task. Support subsets are sampled using random sample, and all methods receive the same selected support examples. Our model outperforms the strongest comparator at every evaluated cap. The mean paired advantage is largest at small support caps, reaching +0.032 ROC-AUC at 32 examples and +0.027 at 64 examples, and decreases to +0.007 at the 4,096-example cap. These results demonstrate the benefit of the method when the labeled context available at inference is small.

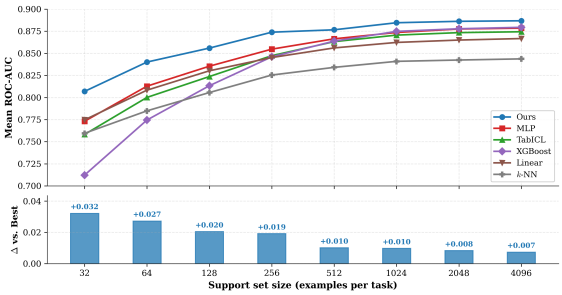


Figure 3: **Performance as a function of maximum support-set size on unseen tasks.** Each task uses the smaller of the requested support cap and its available training-set size. The upper panel reports mean ROC-AUC, and the lower panel reports our mean paired advantage over the strongest baseline.

Because the evaluated tasks are absent from gradient-based meta-training, these gains cannot be attributed to training the shared predictor on additional labels from the same target task

5.6 Performance–Efficiency Trade-off

Increasing orthogonal views and support-set size improves predictive performance but incurs higher inference costs. For instance, averaging 16 views increases our model's ROC-AUC from 0.8718 (single view) to 0.8886 (Table 4). Timing experiments on a single NVIDIA A40 GPU (bf16 autocast) show runtime scales approximately linearly with the number of views. With full support sets, mean evaluation time per task–encoder pair is 1.46 seconds for one view, 5.90 seconds for four, and 23.0 seconds for 16. Support-set size similarly impacts efficiency. On a 54-pair subset, increasing the support cap from 32 to 1,024 examples raises mean evaluation time from 1.14 to 3.90 seconds (four views) and from 4.72 to 14.92 seconds (16 views), inclusive of all processing steps. These results highlight a practical trade-off. Users can dynamically balance predictive performance and computational cost at inference time by adjusting support size and views, requiring no task-specific parameter updates.

6 Conclusion and Limitations

We introduced a universal in-context predictor for different foundation-model embedding spaces. Instead of fitting a new classifier for every task–encoder pair, the model is trained once across many embedding spaces and uses labeled examples as context at inference. The tabular predictor accepts different embedding dimensions directly, while resampled orthogonal basis changes discourage dependence on any single coordinate orientation without discarding information. Once adapted, the model applies to a new task or encoder without task-specific gradient updates.

Across seven domains, the model reaches 0.8886 mean ROC-AUC over 308 task–encoder pairs and improves on the strongest baseline by 0.0198 AUC. The advantage has a positive confidence interval in all four task–encoder generalization regimes and in five of seven domain comparisons. Performance is positive relative to the strongest baseline on 54 of 75 tasks and 26 of 29 encoders, demonstrating broad transfer while also exposing meaningful task-level heterogeneity. The ablations show that repeatedly resampling a full-rank, distance-preserving basis during training closes the seen–unseen gap; the matched-compute inference comparison further shows that orthogonal-view ensembling is most effective when the model was trained for basis robustness. The support-size study shows the largest gains in low-data settings.

This study has several limitations. It considers classification from fixed embeddings but does not examine regression, structured prediction, generation, or joint encoder fine-tuning. All experiments adapt one tabular foundation-model backbone, so transfer to other in-context architectures remains unknown. Improvements are aggregate, and individual task–encoder pairs can favor conventional baselines.

The method trades inference cost for stability: averaging views scales approximately linearly with ensemble size, while full support sets can be expensive. Future work should study efficient basis averaging, other geometry-preserving transformations and in-context backbones, and extensions beyond classification.

Reproducibility Statement

To support reproducibility of the method and empirical benchmark, we provide: (1) Complete architectural, optimization, baseline, and hardware details in Appendix C; (2) Comprehensive specifications, preprocessing steps, and dimensions for all 75 tasks and 29 embedding encoders across 7 domains in Section 4 and Appendix D; (3) All source code, trained model checkpoints and data in the accompanying anonymized repository: https://anonymous.4open.science/r/univarsal_embeding_icl-EE32/README.md.

Ethics Statement

Our benchmark includes biomedical, chemical, and scientific datasets, including bioassays, antibody-development data, protein datasets, and medical-image classification tasks. Although automated classification may support drug discovery and medical research, models used in biological or clinical settings require careful experimental validation and expert oversight.

AI Use Statement

Generative AI tools assisted with substantive drafting, mathematical checking, literature-search support, and analysis-code review. All reported experiments were executed from the project code, and the authors reviewed the resulting claims and take responsibility for the manuscript.

References

Walid Ahmad, Elana Simon, Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. ChemBERTa-2: Towards chemical foundation models. *arXiv preprint arXiv:2209.01712*, 2022.

Guillaume Alain and Yann Ollivier. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Han Altae-Tran, Bharath Ramsundar, Aneesh S Pappu, and Vijay Pande. Low data drug discovery with one-shot learning. *ACS Central Science*, 3(4):283–293, 2017.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Processing Systems*, volume 33, pp. 12449–12460, 2020.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

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.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, et al. WavLM: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6): 1505–1518, 2022.

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016. doi: 10.1145/2939672.2939785.

Alexis Conneau and Douwe Kiela. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

Hoang Anh Dau, Anthony Bagnall, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, and Eamonn Keogh. The UCR time series archive. *IEEE/CAA Journal of Automatica Sinica*, 6(6):1293–1305, 2019.

Angus Dempster, François Petitjean, and Geoffrey I. Webb. ROCKET: exceptionally fast and accurate time series classification using random convolutional kernels. *Data Mining and Knowledge Discovery*, 34(5):1454–1495, 2020.

Angus Dempster, Daniel F. Schmidt, and Geoffrey I. Webb. MiniRocket: A very fast (almost) deterministic transform for time series classification. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 248–257, 2021.

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.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. CLAP: Learning audio concepts from natural language supervision. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023.

Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, et al. Prottrans: Towards understanding the language of life through self-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):7112–7127, 2021.

Ahmed Elnaggar, Hazem Essam, Wafaa Salah-Eldin, Walid Moustafa, Mohamed Elkerdawy, Charlotte Rochereau, and Burkhard Rost. Ankh: Optimized protein language model unlocks general-purpose modelling. *arXiv preprint arXiv:2301.06568*, 2023.

Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with PyTorch Geometric. *arXiv preprint arXiv:1903.02428*, 2019.

Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, 2017.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2733–2743, 2019.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. TabPFN: A transformer that solves small tabular problems in a second. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 28858–28871, 2022.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637:319–326, 2025.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor W Coley, Jimeng Sun, and Marinka Zitnik. Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks*, 2021.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better ImageNet models transfer better? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2661–2671, 2019.

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations*, 2022.

Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, et al. Datasets: A community library for natural language processing. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 175–184, 2021.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

Matthew Middlehurst, Ali Ismail-Fawaz, Antoine Guillaume, Christopher Holder, David Guijo-Rubio, Guzal Bulatova, Leonidas Tsaprounis, Lukasz Mentel, Martin Walter, Patrick Schäfer, and Anthony Bagnall. aeon: a python toolkit for learning from time series. *Journal of Machine Learning Research*, 25(289):1–10, 2024.

Christopher Morris, Nils M. Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. TUDataset: A collection of benchmark datasets for learning with graphs. *ICML Workshop on Graph Representation Learning and Beyond (GRL+)*, 2020.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. MTEB: Massive text embedding benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 2014–2037, 2023.

Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers can do bayesian inference. In *International Conference on Learning Representations (ICLR)*, 2022.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research (TMLR)*, 2024.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICL: A tabular foundation model for in-context learning on large data. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 50817–50847, 2025. URL https://proceedings.mlr.press/v267/qu25d.html.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pp. 8748–8763. PMLR, 2021.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning (ICML)*, pp. 28492–28518. PMLR, 2023.

Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Xi Chen, John Canny, Pieter Abbeel, and Yun S Song. Evaluating protein transfer learning with tape. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pp. 9689–9701, 2019.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pp. 3982–3992, 2019.

Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C Lawrence Zitnick, Jerry Ma, et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, 2021.

Jerret Ross, Brian Belgodere, Vijil Chenthamarakshan, Inkit Padhi, Youssef Sharifi, and Payel Das. Large-scale chemical language representations capture molecular structure and properties. *Nature Machine Intelligence*, 4(12):1256–1264, 2022.

Mustafa Tajjar, Alexander Pfefferle, Lennart Purucker, and Frank Hutter. Towards pretraining text encoders for TabPFN. *arXiv preprint arXiv:2606.04876*, 2026. URL https://arxiv.org/abs/2606.04876.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

Elena Voita and Ivan Titov. Information-theoretic probing with minimum description length. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 183–196, 2020.

Sheng Wang, Yuzhi Guo, Yuhong Wang, Hongmao Sun, and Junzhou Huang. SMILES-BERT: large scale unsupervised pre-training for molecular property prediction. *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pp. 429–436, 2019.

Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Pahari, Karl Leswing, Irina Zhang, Viola Zhou, Chenguang Huang, Antonis Tsovantas, et al. MoleculeNet: a benchmark for molecular machine learning. *Chemical Science*, 9(2):513–530, 2018.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. C-pack: Packaged resources to advance general chinese embedding. *arXiv preprint arXiv:2309.07597*, 2023.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019.

Minghao Xu, Zuobai Zhang, Jiarui Jue, Menghao Chen, Yan Lu, Ming Qu, and Jian Tang. Peer: A comprehensive and multi-task benchmark for protein sequence understanding. In *Advances in Neural Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks*, volume 35, pp. 35156–35173, 2022.

Jiancheng Yang, Rui Shi, Donglai Wei, Ziyue Liu, Linxiang Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. MedMNIST v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, 2023.

Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y Lin, Andy T Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, et al. SUPERB: Speech processing universal performance benchmark. In *Interspeech 2021*, pp. 1194–1198, 2021.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11975–11986, 2023.

Jingxiang Zhang, Lujia Zhong, Zijie Zhu, Shuo Huang, and Yuang Xu. TabPFN beyond tabular data: Calibration and accuracy on multimodal embeddings. *arXiv preprint arXiv:2607.11007*, 2026. URL https://arxiv.org/abs/2607.11007.

A Extended Experimental Results and Algorithms

A.1 Training and Inference Procedures

Algorithm 1 Dynamic Orthogonal-Basis Meta-Training

Require: Multi-domain task store $\mathcal{T}$, training pairs $\mathcal{S}$, maximum meta-steps $C = 8000$
1: Initialize TabICL with LoRA adapters ($r = 8, \alpha = 32$) on FFN layers across all modules
2: **for** $c = 1$ **to** C **do**
3: Sample a task–encoder pair (τ, Φ) with embedding dimension d_Φ from $\mathcal{S}$
4: Context = the complete training set for (τ, Φ)
5: Sample a fresh square orthogonal $R_c \in \mathbb{R}^{d_\Phi \times d_\Phi}$ (QR decomposition of a Gaussian matrix)
6: $\hat{p} \leftarrow \mathcal{M}_\theta(X_s R_c, y_s, X_q R_c)$
7: Update LoRA parameters by cross-entropy on query labels;
8: **end for**

Algorithm 2 Test-Time Orthogonal-View Ensembling

Require: Support set (X_s, y_s) and query features X_q with embedding dimension d, ensemble size $K = 16$, base seed $s_0 = 42$
1: **for** $k = 0$ **to** $K - 1$ **do**
2: Sample a square orthogonal $R_k \in \mathbb{R}^{d \times d}$ using explicit seed $s_0 + k \cdot 10007$
3: $\hat{p}_k \leftarrow \mathrm{softmax}(\mathcal{M}_\theta(X_s R_k, y_s, X_q R_k))$
4: **end for**
5: **return** $\bar{p}_K = K^{-1} \sum_{k=0}^{K-1} \hat{p}_k$

B Variance Reduction from Orthogonal-View Ensembling

Fix the trained model, support set, query, and any one class, and let p_{R_k} be the predicted probability of that class after applying the orthogonal transformation R_k. Define $\mu = \mathbb{E}_R[p_R]$ and $\sigma_R^2 = \mathrm{Var}_R(p_R)$. Because the R_k are sampled independently, the p_{R_k} are conditionally independent and identically distributed. Therefore

$$\mathbb{E}_R[\bar{p}_K] = \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}_R[p_{R_k}] = \mu, \qquad \mathrm{Var}_R(\bar{p}_K) = \frac{1}{K^2} \sum_{k=1}^{K} \mathrm{Var}_R(p_{R_k}) = \frac{\sigma_R^2}{K}. \tag{5}$$

Let p^* denote the target conditional probability and $b = \mu - p^*$. Expanding around μ gives

$$\mathbb{E}_R[(\bar{p}_K - p^*)^2] = \mathrm{Var}_R(\bar{p}_K) + b^2 = \frac{\sigma_R^2}{K} + b^2, \tag{6}$$

because the cross term has expectation zero. Thus averaging over independent orthogonal views reduces conditional Monte Carlo variance at rate $1/K$ while retaining the bias of the view-averaged predictor. ■

C Implementation Details & Hyperparameters

C.1 Meta-Training Specifications

All fine-tuning runs were executed using the TabICL tabular transformer backbone.

- **LoRA Configuration:** Production runs use: rank $r = 8$, scale $\alpha = 32$. Adapters target feed-forward modules named `linear1` and `linear2` in the column embedder, row interactor, and in-context predictor; the remaining weights are frozen. No dropout is applied to the adapters.
- **Random Orthogonal Basis:** For an input matrix $X \in \mathbb{R}^{N \times d}$, a fresh square orthogonal matrix $R \in \mathbb{R}^{d \times d}$ is obtained by QR decomposition of a Gaussian matrix and sampled from the global PyTorch random-number generator at every training step. The transformation changes the coordinate basis without reducing the embedding dimension.

- **Optimization:** AdamW with learning rate 3×10^{-4}, cosine decay to 5% of peak, weight decay 1×10^{-4}, gradient clipping at 1.0, and up to 8,000 meta-steps.
- **Meta-training objective:** Each episode uses the training examples of a meta-training task–encoder pair as the labeled support set. Query examples are sampled from that task's within-task validation split, and their labels provide the cross-entropy training loss.
- **Checkpoint selection:** We evaluate checkpoints on the meta-validation pairs every 1000 optimization steps and select the checkpoint with the highest mean ROC-AUC. Meta-validation pairs do not provide gradient updates. The selected checkpoint is then frozen for all final evaluations. Test examples are used only for final reporting and do not influence checkpoint selection.

C.2 Baseline Configurations

Every method uses the same training, validation, and test sets and is evaluated on the same held-out test examples. Hyperparameters are selected by randomized search using only the training and validation data. The selected pipeline is refitted on the complete training and validation data before test-set evaluation. Feature scaling, when considered, is fitted using only the data available during model selection, preventing leakage from the test set.

- **Linear probe:** A `StandardScaler` followed by ℓ_2-regularized scikit-learn `LogisticRegression` (`lbfgs`, `max_iter=2000`). The search samples C log-uniformly from $[10^{-4}, 10^{2}]$ and chooses `class_weight` from `{None, balanced}`. Multi-class tasks use multinomial logistic regression.
- **k-NN:** Scikit-learn `KNeighborsClassifier`; the search selects standardized or raw features, $k \in \{1, 3, 5, 7, 9, 11, 15, 21, 31, 41, 51, 75, 101\}$, uniform or distance-weighted voting, and cosine, Euclidean, or Manhattan distance.
- **MLP:** A one-hidden-layer network, Linear$(d, h) \rightarrow$ BatchNorm1d$(h) \rightarrow$ ReLU $\rightarrow$ Dropout$(p) \rightarrow$ Linear(h, C). Randomized search considers hidden width $h \in \{64, 128, 256\}$, dropout $p \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5\}$, learning rate in $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$, and weight decay in $\{10^{-4}, 10^{-3}\}$.
- **XGBoost:** `XGBClassifier`. The search samples the number of trees from $[50, 1000]$, maximum depth from $[2, 10]$, learning rate log-uniformly from $[10^{-3}, 0.3]$, row subsampling from $[0.5, 1]$, column subsampling from $[0.3, 1]$, minimum child weight log-uniformly from $[1, 20]$, and split penalty γ, ℓ_1, and ℓ_2 regularization log-uniformly from $[10^{-8}, 5]$, $[10^{-8}, 10]$, and $[10^{-8}, 10]$, respectively. Binary tasks use the `binary:logistic` objective and multi-class tasks use `multi:softprob`.
- **Zero-shot TabICL:** The pretrained `jingang/TabICL` classifier checkpoint (`tabicl-classifier-v2-20260212.ckpt`) is evaluated without parameter updates or hyperparameter search. The complete training set is used as context.

The four classical baselines run on CPU-only jobs with 16 CPU cores and 32 GB of system memory. Zero-shot TabICL runs separately on one NVIDIA A40 GPU with eight CPU cores and 40 GB of system memory.

C.3 Computational Resources

All adaptation runs used one NVIDIA A40 GPU, eight CPU cores, and a 40 GB system-memory allocation. The 40 GB allocation refers to host memory rather than GPU memory;

The saved checkpoint is 41,536,508 bytes (39.6 MiB) and contains 467,992 trainable adapter parameters together with the associated checkpoint state.

C.4 Ablation Configurations and Full Results

Table 4 compares complete training-and-inference configurations. "Seen" contains the In-Distribution evaluations. "Unseen" pools evaluations for which at least one element was withheld from meta-training. Gap is Seen minus Unseen. Gap vs. frozen subtracts the A0 gap, controlling for the different composition and difficulty of the Seen and Unseen pools.

The end-to-end configurations are:

- **A0:** frozen TabICL without LoRA, evaluated with 16 random orthogonal views.
- **A1:** LoRA training on raw embeddings, without an orthogonal transformation during training; inference averages 16 feature-permutation views in the raw basis.
- **A2:** one fixed global random orthogonal transformation, with permutation views in that basis at inference.
- **A3:** one fixed random orthogonal transformation for each encoder; unseen encoders receive a new fixed transformation.
- **A4:** a fixed pool of $M \in \{4, 16, 64\}$ random orthogonal transformations, sampled during training and inference.
- **A5:** per-step random coordinate permutations with sign flips.
- **A6:** a per-step lossy Johnson–Lindenstrauss projection to $k = 128$ dimensions.
- **A7:** a deterministic, unwhitened PCA basis for each encoder, with permutation views at inference.
- **A8:** a per-step full-dimensional Gaussian transformation, which is full rank but does not preserve geometry.
- **A9:** our method, which samples a new random orthogonal transformation at every training step and uses random orthogonal views at inference.

Table 4: End-to-end transformation ablations. Seen is In-Distribution; Unseen pools Novel Task, Novel Encoder, and Joint Generalization. Rows differ in both training and inference. Overall is computed from unrounded run-level values; the displayed rounded Seen and Unseen values may reproduce it only approximately.

ID	Training condition	Inference (16 views)	Seen	Unseen	Gap	Gap vs. frozen	Overall
A0	Frozen TabICL (no LoRA)	Random orthogonal	.8654	.8668	-.0014	.0000	.8659
A1	LoRA, raw input	Permutations, raw basis	.8900	.8690	+.0210	+.0224	.8822
A2	Fixed global orthogonal basis	Permutations, fixed basis	.8897	.8685	+.0212	+.0226	.8818
A3	Fixed basis per encoder	Permutations, encoder basis	.8895	.8745	+.0150	+.0164	.8839
A4	Orthogonal pool, $M = 4$	Pool draws	.8895	.8730	+.0165	+.0179	.8834
A4	Orthogonal pool, $M = 16$	Pool draws	.8893	.8785	+.0108	+.0122	.8853
A4	Orthogonal pool, $M = 64$	Pool draws	.8892	.8830	+.0062	+.0076	.8869
A5	Permutation and sign flips	Permutation/sign draws	.8893	.8720	+.0173	+.0187	.8829
A6	JL projection, $k = 128$	JL draws	.8720	.8705	+.0015	+.0029	.8714
A7	Deterministic PCA	Permutations, PCA basis	.8880	.8740	+.0140	+.0154	.8828
A8	Per-step full-dimensional Gaussian	Gaussian draws	.8850	.8800	+.0050	+.0064	.8831
A9	Per-step random orthogonal (ours)	Random orthogonal	.8891	.8879	+.0012	+.0026	.8886

Table 5: Training condition by inference protocol, reported as Overall ROC-AUC. Every cell within a column uses the identical inference protocol, so rows can be compared directly. Table 4: permutations for A1 and A3, and random orthogonal views for A0 and A9.

ID	Training condition	Native	One random orthogonal	16 permutations	16 random orthogonal	Sensitivity
A0	Frozen TabICL	.8620	.8580	.8640	.8659	-.0040
A1	Raw input	.8790	.8600	.8822	.8680	-.0190
A3	Fixed basis per encoder	.8810	.8720	.8839	.8790	-.0090
A9	Per-step random orthogonal (ours)	.8744	.8718	.8770	.8886	-.0026

D Complete Dataset and Task Manifest

D.1 Representation Extraction, Pooling, and Preprocessing Protocols

To ensure rigorous reproducibility and eliminate data leakage, representations are extracted using frozen, off-the-shelf foundation models prior to downstream adaptation.

- **Vision:** Images are passed through each model's native `AutoProcessor`. For contrastive dual-encoder models exposing a dedicated image-projection head (`siglip_base`, `metaclip_b16`), the projected image-feature vector is used directly. For other models, the pooled output (`pooler_output` where present, else the `[CLS]` token, i.e. position 0 of the last hidden state) is extracted; if the resulting representation still has a token dimension, it is mean-pooled over tokens.
- **Molecules:** Canonical SMILES are tokenized up to 512 tokens per the checkpoint's own tokenizer. All five molecular encoders are bidirectional Transformer encoders (ChemBERTa family, SMILES-BERT); representations are obtained by mean-pooling the last hidden state over non-padding tokens (attention-mask-weighted average), yielding $d \in \{384, 768\}$.
- **Proteins:** Sequences are upper-cased and stripped of whitespace, then tokenized up to 1,024 residues. Representations are the attention-mask-weighted mean of the last hidden state. For the one encoder-decoder checkpoint (`ankh_base`, a T5 encoder-decoder), only the encoder stack is run — `model.encoder(...)` rather than the full seq2seq `forward()` — so no decoder input is required.
- **Text:** Sentences are tokenized up to the checkpoint's native context length and mean-pooled over the attention-masked last hidden state, following each model's own embedding convention (all four working text encoders are contrastively-trained sentence/passage embedders).
- **Audio:** Raw waveforms are resampled to each model's native sample rate (typically 16 kHz) via the model's own `AutoFeatureExtractor`. whisper_large_v3_turbo runs only its encoder stack (mean-pooled over the encoder's last hidden state); contrastive models exposing `get_audio_features` (`clap_general`) use that projection directly; remaining models mean-pool the last hidden state over time.
- **Graphs:** Each source graph (adjacency and node labels) is passed through one of four 3-layer graph neural networks — GIN (global sum pooling), GCN, GAT, and GraphSAGE (all three: global mean pooling) — producing a fixed $d = 256$ graph-level embedding per architecture. Each encoder uses a single self-supervised checkpoint, trained without labels and shared across all graph tasks.
- **Time Series:** Univariate sequences are zero-padded to the batch's maximum length. `minirocket`/`rocket` apply the corresponding `aeon` random-convolutional-kernel transform, while `chronos_t5_small` runs the Chronos time-series encoder.
- **Data Split Protocol:** Every task is divided into disjoint training, validation, and test sets. We retain the dataset's official split when one is available; otherwise, we use the standard splitting protocol for that domain. The same partitions are used for all methods. Classes are remapped to a dense $0..C-1$ range at task-creation time; no task is converted to a binary decomposition (see the native multi-class discussion in Section 4.1).

D.2 Complete Benchmark Suite and Dataset Inventory (75 Tasks)

Table 6 provides a comprehensive technical inventory of all 75 benchmark tasks across the 7 domains, detailing per-task sample counts, class counts, minority-class fractions, sources, and descriptions ($N_{\text{total}} = 169{,}554$).

Table 6: Complete inventory of all 75 benchmark tasks and 169,554 samples across seven data modalities. N is the number of labeled examples used for that task; C is the number of classes; $\hat{p}_{\min}$ is the minority-class fraction. Every method uses the same training, validation, and test partition for each task.

Task ID	N	C	$\hat{p}_{\min}$	Source	Description
Molecules (10 tasks, 28,939 samples)					
`tdc_ames`	5,727	2	0.4741	Therapeutics Data Commons (TOX)	Ames bacterial reverse mutagenicity
`tdc_b3db_classification`	5,616	2	0.3504	Therapeutics Data Commons (ADME)	B3DB Blood-Brain Barrier distribution
`tdc_bbb_martins`	2,030	2	0.236	Therapeutics Data Commons (ADME)	Blood-brain barrier penetration (Martins)

continued on next page

Table 6 continued from previous page

Task ID	N	C	$\hat{p}_{\min}$	Source	Description
tdc_bioavailability_ma	640	2	0.2313	Therapeutics Data Commons (ADME)	Oral drug bioavailability (Ma)
tdc_cyp1a2_veith	6,000	2	0.464	Therapeutics Data Commons (ADME)	CYP1A2 enzyme inhibition (Veith)
tdc_cyp2d6_veith	6,000	2	0.1928	Therapeutics Data Commons (ADME)	CYP2D6 enzyme inhibition (Veith)
tdc_dili	475	2	0.4968	Therapeutics Data Commons (TOX)	Drug-induced liver injury toxicity
tdc_herg	655	2	0.3115	Therapeutics Data Commons (TOX)	hERG potassium cardiac channel block
tdc_hia_hou	578	2	0.1349	Therapeutics Data Commons (ADME)	Human intestinal absorption (Hou)
tdc_pgp_broccatelli	1,218	2	0.4663	Therapeutics Data Commons (ADME)	P-glycoprotein transporter inhibition
Proteins (15 tasks, 65,627 samples)					
prot_allergenic	5,680	2	0.5	HuggingFace (LrZhang/Protein_allergenic)	Allergenic protein sequence classification
prot_anticancer_peptide	949	4	0.019	HuggingFace (gelabgaboo/Anticancer-peptide)	Anticancer peptide activity classification (4 classes: active/inactive levels)
prot_cloning_stage	8,000	2	0.3944	HuggingFace (proteinglm/cloning_clf)	Protein cloning-stage success classification (PredPPCrys structural genomics pipeline)
prot_deeploc_binary	1,500	2	0.3747	HuggingFace (SaProtHub/Dataset-Binary_Localization-DeepLoc)	DeepLoc protein membrane vs soluble localization
prot_deeploc_multi	1,500	10	0.01	HuggingFace (SaProtHub/Dataset-Subcellular_Localization-DeepLoc)	DeepLoc 10-class subcellular localization
prot_hemolytic_peptide	4,611	2	0.2065	HuggingFace (jablonkagroup/peptides_hemolytic)	Hemolytic vs non-hemolytic peptide classification
prot_material_production	8,000	2	0.2999	HuggingFace (proteinglm/material_production)	Protein material-production stage success classification (PredPPCrys structural genomics pipeline)
prot_metal_ion	1,500	2	0.4707	HuggingFace (SaProtHub/Dataset-Metal_Ion_Binding)	Protein metal ion binding prediction
prot_nonfouling_peptide	8,000	2	0.2097	HuggingFace (jablonkagroup/peptides_nonfouling)	Non-fouling vs fouling peptide classification (resistance to non-specific binding)
prot_remote_homology	3,000	7	0.015	HuggingFace (proteinea/remote_homology)	Protein fold structural classification (7 classes)
prot_sabdab_chen	2,409	2	0.2001	Therapeutics Data Commons (Develop)	SAbDab antibody sequence developability viability
prot_solubility	2,500	2	0.4132	SaProtHub (Dataset-Solubility)	Protein solubility classification (binary)
prot_sorting_signal	1,592	9	0.022	HuggingFace (AI4Protein/SortingSignal)	Protein sorting-signal type classification (SP/MT/NLS/PTS/GPI/CH/NES/TH/TM), single-type sequences only
prot_temperature_stability	8,000	2	0.4954	HuggingFace (proteinglm/temperature_stability)	Protein structural stability at 65C classification (binary)
prot_toxin_peptide	8,386	2	0.4987	HuggingFace (tanthinhdt/toxinpred3)	Toxic vs non-toxic peptide classification (ToxinPred3 dataset)
Vision (10 tasks, 11,460 samples)					
vision_beans	1,034	3	0.3298	HuggingFace (beans)	Bean leaf disease classification
vision_bloodmnist	2,000	8	0.0745	MedMNIST (bloodmnist)	Peripheral blood cell classification
vision_breastmnist	546	2	0.2692	MedMNIST (breastmnist)	Breast ultrasound tumor classification
vision_cifar10	600	10	0.0767	HuggingFace (cifar10)	CIFAR-10 real photos
vision_dermamnist	2,000	7	0.012	MedMNIST (dermamnist)	Dermatoscopy skin lesion classification
vision_fashion_mnist	600	10	0.0917	HuggingFace (fashion_mnist)	Fashion-MNIST clothing items
vision_mnist	1,000	10	0.078	HuggingFace (mnist)	Handwritten digit classification (10 classes)
vision_pneumoniamnist	2,000	2	0.267	MedMNIST (pneumoniamnist)	Pediatric chest X-ray pneumonia classification
vision_retinamnist	1,080	5	0.0611	MedMNIST (retinamnist)	Retinal fundus diabetic retinopathy classification
vision_svhn	600	10	0.0567	HuggingFace (svhn)	Street View House Numbers digits
Text (11 tasks, 23,264 samples)					
text_agnews	1,500	4	0.1507	HuggingFace (ag_news)	AG News topic classification

continued on next page

Table 6 continued from previous page

Task ID	N	C	$\hat{p}_{\min}$	Source	Description
text_financial_phrasebank	2,264	3	0.1338	HuggingFace (financial_phrasebank/sentences_allagree)	Financial sentiment classification
text_imdb	3,000	2	0.1667	HuggingFace (imdb/)	IMDb large movie reviews sentiment
text_rotten_tomatoes	3,000	2	0.4997	HuggingFace (rotten_tomatoes/)	Rotten Tomatoes movie reviews sentiment
text_sst2	1,500	2	0.45	HuggingFace (glue)	Stanford Sentiment Treebank v2
text_tweet_emotion	1,500	4	0.0967	HuggingFace (tweet_eval)	Twitter emotion recognition
text_tweet_hate	3,000	2	0.42	HuggingFace (tweet_eval/hate)	Hate speech detection in tweets
text_tweet_irony	1,500	2	0.4967	HuggingFace (tweet_eval)	Twitter irony detection
text_tweet_offensive	3,000	2	0.344	HuggingFace (tweet_eval/offensive)	Offensive language detection in tweets
text_tweet_sentiment	1,500	3	0.1573	HuggingFace (tweet_eval)	Twitter sentiment recognition
text_yelp_polarity	1,500	2	0.4727	HuggingFace (yelp_polarity)	Yelp restaurant reviews sentiment
Audio (7 tasks, 7,574 samples)					
audio_crema_d	2,000	6	0.1465	HuggingFace (confit/crema-d)	Crowd-sourced emotional speech classification (6 emotions)
audio_emodb	535	7	0.086	HuggingFace (renumics/emodb)	German emotional speech classification (7 emotions)
audio_esc10	400	10	0.1	HuggingFace (ashraq/esc50)	ESC-10 environmental sound classification (10 classes)
audio_gtzan	999	10	0.0991	HuggingFace (marsyas/gtzan)	Music genre classification across 10 genres (GTZAN)
audio_ravdess	1,440	8	0.0667	HuggingFace (xbgoose/ravdess)	RAVDESS emotional speech classification
audio_urbansound8k	2,000	10	0.042	HuggingFace (danavery/urbansound8K)	Urban environmental sound classification (10 classes)
audio_voxforge_lang	200	6	0.135	HuggingFace (DynamicSuperb/LanguageIdentification_VoxForge)	Spoken language identification (6 European languages) from VoxForge recordings
Graphs (10 tasks, 11,161 samples)					
graph_aids	2,000	2	0.2	PyG TUDataset (AIDS)	Anti-HIV activity screening
graph_bzr	405	2	0.2123	TUDataset (BZR)	BZR receptor ligand molecular graphs
graph_cox2	467	2	0.2184	TUDataset (COX2)	COX2 inhibitor molecular graphs
graph_dd	1,178	2	0.4134	PyG TUDataset (DD)	Enzymes vs non-enzymes protein structures
graph_dhfr	756	2	0.3902	PyG TUDataset (DHFR)	Dihydrofolate reductase inhibitors
graph_enzymes	600	6	0.1667	PyG TUDataset (ENZYMES)	Protein enzymes classified into 6 EC levels
graph_mutag	188	2	0.3351	TUDataset (MUTAG)	Mutagenic molecular compounds (MUTAG)
graph_nci1	4,110	2	0.4995	PyG TUDataset (NCI1)	Chemical compounds screened against lung cancer
graph_proteins	1,113	2	0.4043	TUDataset (PROTEINS)	Protein 3D structure macromolecule graphs
graph_ptc_mr	344	2	0.4419	PyG TUDataset (PTC_MR)	Carcinogenicity in male rats
Time Series (12 tasks, 21,529 samples)					
ts_chinatown	363	2	0.2865	UCR Archive (Chinatown) via AEON	Pedestrian traffic flow in Melbourne Chinatown
ts_ecg200	100	2	0.31	UCR Archive (ECG200)	Electrocardiogram heartbeat series (ECG200)
ts_electric_devices	8,926	7	0.057	UCR Archive (ElectricDevices)	Household electricity device telemetry
ts_ford_a	3,601	2	0.4874	UCR Archive (FordA)	Automotive subsystem acoustic noise telemetry
ts_ford_b	4,446	2	0.4915	UCR Archive (FordB) via AEON	Engine noise symptom classification under noisy condition
ts_gunpoint	50	2	0.48	UCR Archive (GunPoint)	Surveillance video motion tracker sensor
ts_italy_power	67	2	0.4925	UCR Archive (ItalyPowerDemand)	Italy national electrical power demand profile
ts_plane	210	7	0.1429	UCR Archive (Plane) via AEON	Airplane shape contour time series classification
ts_sony_aibo	621	2	0.438	UCR Archive (SonyAIBORobotSurface1) via AEON	Robot accelerometer surface classification
ts_strawberry	983	2	0.3571	UCR Archive (Strawberry) via AEON	Strawberry puree food spectrography authentication
ts_two_lead_ecg	1,162	2	0.5	UCR Archive (TwoLeadECG) via AEON	Two-lead ECG heartbeat classification
ts_wafer	1,000	2	0.097	UCR Archive (Wafer)	Semiconductor wafer fabrication sensor

D.3 Complete Foundation Model Encoder Inventory (29 Encoders)

Table 7 documents all 29 encoders used in this study, spanning representation dimensions $d \in [256, 1280]$ across all seven domains.

Table 7: Complete inventory of all 29 encoders across seven modalities ($d \in [256, 1280]$).

Domain	Encoder ID	Checkpoint	d	Architecture & Pretraining
Molecules	chemberta_77m_mtr	DeepChem/ChemBERTa-77M-MTR	384	RoBERTa, masked-token-replacement pretraining on SMILES
Molecules	chemberta_v2	seyonec/ChemBERTa_zinc250k_v2_40k	768	RoBERTa, masked LM, ZINC-250k
Molecules	chemberta_zinc_base	seyonec/ChemBERTa-zinc-base-v1	768	RoBERTa, masked LM on ZINC SMILES
Molecules	pubchem10m_bpe	DeepChem/ChemBERTa-10M-MTR	384	RoBERTa-small, 10M-SMILES MTR pretraining
Molecules	smiles_bert	unikei/bert-base-smiles	768	BERT-base, masked SMILES recovery
Proteins	ankh_base	ElnaggarLab/ankh-base	768	T5 encoder-decoder (encoder stack only used)
Proteins	esm2_150m	facebook/esm2_t30_150M_UR50D	640	Lightweight ESM-2 variant, UniRef50
Proteins	esm2_650m	facebook/esm2_t33_650M_UR50D	1280	Transformer encoder, UniRef50 masked-residue pretraining
Proteins	prot_bert	Rostlab/prot_bert	1024	BERT, UniRef100 masked amino-acid pretraining
Vision	beit_base_patch16_224	microsoft/beit-base-patch16-224	768	ViT-B/16, masked visual-token prediction
Vision	dinov2_small	facebook/dinov2-small	384	ViT-S/14, self-distillation on LVD-142M
Vision	metaclip_b16	facebook/metaclip-b16-400m	512	ViT-B/16, curated CLIP-style contrastive pretraining
Vision	siglip_base	google/siglip-base-patch16-224	768	ViT-B/16, sigmoid image-text contrastive
Vision	vit_base_patch16_224	google/vit-base-patch16-224-in21k	768	ViT-B/16, supervised ImageNet-21k pretraining
Text	arctic_embed_m_v2	Snowflake/snowflake-arctic-embed-m-v1.5	768	Matryoshka-trained embedding transformer
Text	bge_large_en_v1_5	BAAI/bge-large-en-v1.5	1024	BERT-large, RetroMAE + contrastive text pretraining
Text	bge_m3	BAAI/bge-m3	1024	Multi-lingual/granularity embedder, self-knowledge distillation
Text	nomic_embed_v1_5	nomic-ai/nomic-embed-text-v1.5	768	Long-context embedder, Matryoshka representation learning
Audio	ast_audioset	MIT/ast-finetuned-audioset-10-10-0.4593	768	Spectrogram transformer, AudioSet-supervised
Audio	clap_general	laion/larger_clap_general	512	HTSAT + RoBERTa, contrastive language-audio pretraining
Audio	wavlm_base_plus	microsoft/wavlm-base-plus	768	Conv+Transformer, denoising & diarization SSL on 60k hrs speech
Audio	whisper_large_v3_turbo	openai/whisper-large-v3-turbo	1280	Encoder-only pass of ASR transformer (encoder stack only used)
Graphs	pyg_gat	anon/gat-graphcl-tud	256	GAT, label-free GraphCL SSL on pooled TUDataset graphs; global mean pooling
Graphs	pyg_gcn	anon/gcn-graphcl-tud	256	GCN, label-free GraphCL SSL on pooled TUDataset graphs; global mean pooling
Graphs	pyg_gin	anon/gin-graphcl-tud	256	GIN, label-free GraphCL SSL on pooled TUDataset graphs; global sum pooling
Graphs	pyg_sage	anon/sage-graphcl-tud	256	GraphSAGE, label-free GraphCL SSL on pooled TUDataset graphs; global mean pooling
Time Series	chronos_t5_small	amazon/chronos-t5-small	512	T5-based time-series foundation model
Time Series	minirocket	MiniRocket transform, aeon	504	Random convolutional-kernel feature transform
Time Series	rocket	Rocket transform, aeon	1024	Random convolutional-kernel feature transform

D.4 Benchmark Cardinality

Across the benchmark suite, 75 tasks define 308 task–encoder pairs (5 encoders per task for molecules and vision; 4 for proteins, text, audio, and graphs; and 3 for time series, per Table 7):

- **Molecules (10 tasks, 50 pairs):** 5 encoders $\times$ 10 tasks.

- **Proteins (15 tasks, 60 pairs):** 4 encoders $\times$ 15 tasks.
- **Vision (10 tasks, 50 pairs):** 5 encoders $\times$ 10 tasks.
- **Text (11 tasks, 44 pairs):** 4 encoders $\times$ 11 tasks.
- **Audio (7 tasks, 28 pairs):** 4 encoders $\times$ 7 tasks.
- **Graphs (10 tasks, 40 pairs):** 4 encoders $\times$ 10 tasks.
- **Time Series (12 tasks, 36 pairs):** 3 encoders $\times$ 12 tasks.

Total: $50 + 60 + 50 + 44 + 28 + 40 + 36 = 308$ task–encoder pairs across 75 tasks.

E Raw Per-Pair Evaluation Results

E.1 Raw Per-Pair Table

Table 8 lists ROC-AUC for the 308 task–encoder pairs for our model and all five baselines, underlying every aggregate reported in Section 5. **Ours** is the mean $\pm$ standard deviation across three runs; because a pair's generalization regime may vary between runs, this column is a per-pair average rather than a regime-specific comparison. The regime-resolved analysis appears in Table 2.

Table 8: Raw per-(task, encoder)-pair ROC-AUC for all 308 pairs. **Ours** is the mean $\pm$ standard deviation across three runs. Baseline columns report kNN, linear probe, MLP, pretrained zero-shot (ZS) TabICL, and XGBoost on the same pair.

Task	Encoder	Ours	kNN	Lin	MLP	ZS	XGB
Molecules (50 pairs)							
tdc_ames	chemberta_77m_mtr	0.842±0.001	0.781	0.796	0.824	0.832	0.808
tdc_ames	chemberta_v2	0.800±0.002	0.767	0.751	0.789	0.791	0.766
tdc_ames	chemberta_zinc_base	0.776±0.004	0.748	0.704	0.759	0.771	0.761
tdc_ames	pubchem10m_bpe	0.850±0.002	0.808	0.796	0.834	0.841	0.821
tdc_ames	smiles_bert	0.771±0.002	0.726	0.691	0.726	0.754	0.744
tdc_b3db_classification	chemberta_77m_mtr	0.940±0.003	0.882	0.880	0.917	0.915	0.900
tdc_b3db_classification	chemberta_v2	0.936±0.000	0.915	0.876	0.895	0.926	0.913
tdc_b3db_classification	chemberta_zinc_base	0.927±0.002	0.901	0.880	0.903	0.924	0.910
tdc_b3db_classification	pubchem10m_bpe	0.937±0.002	0.902	0.886	0.906	0.934	0.918
tdc_b3db_classification	smiles_bert	0.940±0.002	0.923	0.886	0.902	0.927	0.927
tdc_bbb_martins	chemberta_77m_mtr	0.917±0.002	0.877	0.857	0.897	0.917	0.887
tdc_bbb_martins	chemberta_v2	0.901±0.002	0.840	0.878	0.879	0.893	0.868
tdc_bbb_martins	chemberta_zinc_base	0.888±0.001	0.823	0.814	0.879	0.860	0.849
tdc_bbb_martins	pubchem10m_bpe	0.912±0.003	0.870	0.867	0.901	0.909	0.863
tdc_bbb_martins	smiles_bert	0.906±0.004	0.853	0.848	0.881	0.896	0.890
tdc_bioavailability_ma	chemberta_77m_mtr	0.660±0.011	0.687	0.693	0.702	0.641	0.681
tdc_bioavailability_ma	chemberta_v2	0.694±0.019	0.670	0.613	0.615	0.640	0.670
tdc_bioavailability_ma	chemberta_zinc_base	0.682±0.015	0.599	0.685	0.662	0.655	0.649
tdc_bioavailability_ma	pubchem10m_bpe	0.702±0.008	0.695	0.627	0.695	0.682	0.635
tdc_bioavailability_ma	smiles_bert	0.676±0.023	0.687	0.632	0.693	0.601	0.619
tdc_cyp1a2_veith	chemberta_77m_mtr	0.908±0.001	0.831	0.889	0.899	0.901	0.889
tdc_cyp1a2_veith	chemberta_v2	0.909±0.000	0.865	0.878	0.894	0.902	0.900
tdc_cyp1a2_veith	chemberta_zinc_base	0.907±0.000	0.868	0.861	0.883	0.893	0.893
tdc_cyp1a2_veith	pubchem10m_bpe	0.914±0.001	0.863	0.889	0.901	0.912	0.901
tdc_cyp1a2_veith	smiles_bert	0.880±0.001	0.838	0.834	0.875	0.865	0.864
tdc_cyp2d6_veith	chemberta_77m_mtr	0.843±0.005	0.754	0.822	0.819	0.833	0.830
tdc_cyp2d6_veith	chemberta_v2	0.818±0.002	0.717	0.740	0.815	0.810	0.817
tdc_cyp2d6_veith	chemberta_zinc_base	0.811±0.001	0.713	0.745	0.791	0.796	0.784
tdc_cyp2d6_veith	pubchem10m_bpe	0.843±0.004	0.742	0.803	0.821	0.838	0.833

continued on next page

Table 8 continued from previous page

Task	Encoder	Ours	kNN	Lin	MLP	ZS	XGB
tdc_cyp2d6_veith	smiles_bert	0.763±0.007	0.706	0.721	0.767	0.747	0.749
tdc_dili	chemberta_77m_mtr	0.849±0.007	0.720	0.766	0.783	0.846	0.751
tdc_dili	chemberta_v2	0.882±0.006	0.867	0.727	0.813	0.847	0.835
tdc_dili	chemberta_zinc_base	0.871±0.006	0.857	0.785	0.835	0.863	0.842
tdc_dili	pubchem10m_bpe	0.921±0.002	0.887	0.809	0.887	0.922	0.907
tdc_dili	smiles_bert	0.862±0.002	0.729	0.726	0.798	0.856	0.772
tdc_herg	chemberta_77m_mtr	0.830±0.017	0.775	0.726	0.835	0.794	0.802
tdc_herg	chemberta_v2	0.845±0.012	0.763	0.789	0.762	0.805	0.841
tdc_herg	chemberta_zinc_base	0.843±0.009	0.726	0.781	0.778	0.837	0.795
tdc_herg	pubchem10m_bpe	0.789±0.011	0.711	0.765	0.755	0.765	0.746
tdc_herg	smiles_bert	0.743±0.007	0.650	0.768	0.688	0.698	0.738
tdc_hia_hou	chemberta_77m_mtr	0.973±0.002	0.919	0.946	0.943	0.960	0.948
tdc_hia_hou	chemberta_v2	0.973±0.001	0.953	0.954	0.955	0.957	0.937
tdc_hia_hou	chemberta_zinc_base	0.933±0.005	0.903	0.914	0.928	0.929	0.875
tdc_hia_hou	pubchem10m_bpe	0.977±0.001	0.946	0.968	0.976	0.970	0.960
tdc_hia_hou	smiles_bert	0.981±0.002	0.973	0.973	0.983	0.973	0.984
tdc_pgp_broccatelli	chemberta_77m_mtr	0.949±0.002	0.905	0.838	0.923	0.928	0.913
tdc_pgp_broccatelli	chemberta_v2	0.914±0.001	0.874	0.855	0.880	0.895	0.895
tdc_pgp_broccatelli	chemberta_zinc_base	0.902±0.008	0.883	0.875	0.886	0.878	0.888
tdc_pgp_broccatelli	pubchem10m_bpe	0.941±0.001	0.928	0.899	0.929	0.943	0.921
tdc_pgp_broccatelli	smiles_bert	0.901±0.002	0.879	0.818	0.860	0.882	0.869
Proteins (60 pairs)							
prot_allergenic	ankh_base	0.975±0.001	0.955	0.960	0.972	0.973	0.973
prot_allergenic	esm2_150m	0.980±0.000	0.959	0.958	0.971	0.977	0.974
prot_allergenic	esm2_650m	0.982±0.000	0.971	0.962	0.976	0.981	0.975
prot_allergenic	prot_bert	0.980±0.000	0.958	0.958	0.973	0.973	0.972
prot_anticancer_peptide	ankh_base	0.957±0.002	0.854	0.934	0.923	0.956	0.946
prot_anticancer_peptide	esm2_150m	0.963±0.001	0.860	0.924	0.892	0.963	0.941
prot_anticancer_peptide	esm2_650m	0.953±0.000	0.888	0.856	0.870	0.936	0.931
prot_anticancer_peptide	prot_bert	0.935±0.003	0.818	0.878	0.855	0.938	0.937
prot_cloning_stage	ankh_base	0.751±0.009	0.681	0.708	0.748	0.758	0.757
prot_cloning_stage	esm2_150m	0.751±0.005	0.680	0.723	0.767	0.730	0.726
prot_cloning_stage	esm2_650m	0.741±0.004	0.662	0.732	0.755	0.706	0.730
prot_cloning_stage	prot_bert	0.731±0.007	0.658	0.695	0.744	0.715	0.716
prot_deeploc_binary	ankh_base	0.988±0.001	0.990	0.976	0.989	0.988	0.988
prot_deeploc_binary	esm2_150m	0.993±0.001	0.970	0.984	0.991	0.991	0.989
prot_deeploc_binary	esm2_650m	0.992±0.001	0.980	0.981	0.984	0.993	0.986
prot_deeploc_binary	prot_bert	0.989±0.001	0.973	0.980	0.982	0.979	0.977
prot_deeploc_multi	ankh_base	0.968±0.004	0.923	0.957	0.970	0.958	0.946
prot_deeploc_multi	esm2_150m	0.967±0.004	0.925	0.955	0.953	0.967	0.939
prot_deeploc_multi	esm2_650m	0.969±0.003	0.923	0.963	0.971	0.974	0.971
prot_deeploc_multi	prot_bert	0.957±0.004	0.884	0.941	0.947	0.942	0.931
prot_hemolytic_peptide	ankh_base	0.727±0.010	0.643	0.645	0.696	0.713	0.681
prot_hemolytic_peptide	esm2_150m	0.732±0.006	0.650	0.641	0.707	0.719	0.655
prot_hemolytic_peptide	esm2_650m	0.721±0.010	0.627	0.642	0.708	0.695	0.684
prot_hemolytic_peptide	prot_bert	0.686±0.008	0.629	0.632	0.676	0.665	0.650
prot_material_production	ankh_base	0.825±0.003	0.780	0.738	0.819	0.811	0.811
prot_material_production	esm2_150m	0.835±0.001	0.769	0.766	0.828	0.820	0.819
prot_material_production	esm2_650m	0.838±0.001	0.778	0.761	0.840	0.804	0.827
prot_material_production	prot_bert	0.824±0.001	0.755	0.763	0.820	0.803	0.806
prot_metal_ion	ankh_base	0.892±0.002	0.857	0.801	0.884	0.894	0.893
prot_metal_ion	esm2_150m	0.882±0.002	0.869	0.853	0.885	0.891	0.869
prot_metal_ion	esm2_650m	0.882±0.002	0.848	0.839	0.871	0.878	0.876
prot_metal_ion	prot_bert	0.869±0.003	0.844	0.821	0.840	0.879	0.847
prot_nonfouling_peptide	ankh_base	0.938±0.001	0.910	0.899	0.934	0.936	0.930
prot_nonfouling_peptide	esm2_150m	0.938±0.000	0.899	0.918	0.937	0.938	0.924
prot_nonfouling_peptide	esm2_650m	0.937±0.001	0.893	0.907	0.936	0.936	0.926

continued on next page

Table 8 continued from previous page

Task	Encoder	Ours	kNN	Lin	MLP	ZS	XGB
prot_nonfouling_peptide	prot_bert	0.928±0.001	0.891	0.891	0.927	0.928	0.916
prot_remote_homology	ankh_base	0.963±0.005	0.901	0.950	0.951	0.960	0.928
prot_remote_homology	esm2_150m	0.971±0.002	0.861	0.955	0.957	0.977	0.910
prot_remote_homology	esm2_650m	0.969±0.005	0.856	0.962	0.967	0.951	0.910
prot_remote_homology	prot_bert	0.950±0.001	0.812	0.944	0.953	0.923	0.890
prot_sabdab_chen	ankh_base	0.891±0.006	0.839	0.872	0.895	0.897	0.894
prot_sabdab_chen	esm2_150m	0.890±0.003	0.828	0.860	0.879	0.892	0.894
prot_sabdab_chen	esm2_650m	0.881±0.006	0.843	0.854	0.879	0.871	0.892
prot_sabdab_chen	prot_bert	0.881±0.007	0.831	0.857	0.860	0.882	0.887
prot_solubility	ankh_base	0.734±0.004	0.644	0.646	0.697	0.724	0.675
prot_solubility	esm2_150m	0.663±0.005	0.631	0.616	0.663	0.655	0.645
prot_solubility	esm2_650m	0.683±0.009	0.624	0.673	0.652	0.665	0.657
prot_solubility	prot_bert	0.663±0.004	0.605	0.648	0.652	0.687	0.642
prot_sorting_signal	ankh_base	0.964±0.001	0.906	0.937	0.942	0.959	0.916
prot_sorting_signal	esm2_150m	0.971±0.001	0.913	0.945	0.950	0.966	0.924
prot_sorting_signal	esm2_650m	0.965±0.001	0.891	0.948	0.959	0.950	0.911
prot_sorting_signal	prot_bert	0.964±0.003	0.872	0.946	0.951	0.950	0.919
prot_temperature_stability	ankh_base	0.950±0.002	0.774	0.948	0.959	0.954	0.947
prot_temperature_stability	esm2_150m	0.983±0.000	0.955	0.972	0.981	0.978	0.977
prot_temperature_stability	esm2_650m	0.988±0.001	0.965	0.983	0.988	0.986	0.983
prot_temperature_stability	prot_bert	0.978±0.000	0.940	0.973	0.980	0.974	0.976
prot_toxin_peptide	ankh_base	0.937±0.000	0.910	0.877	0.923	0.927	0.921
prot_toxin_peptide	esm2_150m	0.936±0.002	0.916	0.896	0.922	0.929	0.925
prot_toxin_peptide	esm2_650m	0.939±0.001	0.914	0.906	0.929	0.923	0.933
prot_toxin_peptide	prot_bert	0.909±0.002	0.868	0.857	0.894	0.885	0.891
Vision (50 pairs)							
vision_beans	beit_base_patch16_224	0.990±0.001	0.974	0.990	0.989	0.986	0.977
vision_beans	dinov2_small	0.979±0.002	0.954	0.974	0.976	0.978	0.961
vision_beans	metaclip_b16	0.995±0.000	0.975	0.995	0.996	0.993	0.988
vision_beans	siglip_base	0.985±0.001	0.961	0.988	0.989	0.980	0.970
vision_beans	vit_base_patch16_224	0.981±0.001	0.954	0.986	0.984	0.975	0.956
vision_bloodmnist	beit_base_patch16_224	0.984±0.001	0.925	0.981	0.984	0.980	0.957
vision_bloodmnist	dinov2_small	0.979±0.000	0.894	0.973	0.977	0.971	0.939
vision_bloodmnist	metaclip_b16	0.939±0.000	0.824	0.931	0.939	0.929	0.890
vision_bloodmnist	siglip_base	0.963±0.001	0.897	0.955	0.958	0.954	0.920
vision_bloodmnist	vit_base_patch16_224	0.982±0.000	0.910	0.980	0.982	0.981	0.961
vision_breastmnist	beit_base_patch16_224	0.808±0.008	0.711	0.793	0.798	0.744	0.743
vision_breastmnist	dinov2_small	0.825±0.007	0.771	0.792	0.811	0.784	0.782
vision_breastmnist	metaclip_b16	0.760±0.009	0.661	0.670	0.728	0.692	0.702
vision_breastmnist	siglip_base	0.817±0.004	0.730	0.791	0.791	0.696	0.770
vision_breastmnist	vit_base_patch16_224	0.754±0.008	0.730	0.731	0.765	0.655	0.683
vision_cifar10	beit_base_patch16_224	1.000±0.000	1.000	1.000	1.000	1.000	0.998
vision_cifar10	dinov2_small	0.999±0.000	0.986	1.000	1.000	0.999	0.987
vision_cifar10	metaclip_b16	0.998±0.001	0.998	0.997	0.998	1.000	0.990
vision_cifar10	siglip_base	1.000±0.000	1.000	0.999	1.000	0.998	0.996
vision_cifar10	vit_base_patch16_224	1.000±0.000	1.000	1.000	1.000	0.999	0.995
vision_dermamnist	beit_base_patch16_224	0.864±0.005	0.679	0.843	0.846	0.857	0.822
vision_dermamnist	dinov2_small	0.870±0.004	0.678	0.814	0.811	0.843	0.796
vision_dermamnist	metaclip_b16	0.819±0.013	0.664	0.812	0.815	0.830	0.784
vision_dermamnist	siglip_base	0.876±0.007	0.680	0.825	0.852	0.859	0.818
vision_dermamnist	vit_base_patch16_224	0.851±0.004	0.693	0.805	0.791	0.820	0.805
vision_fashion_mnist	beit_base_patch16_224	0.988±0.000	0.947	0.980	0.981	0.987	0.966
vision_fashion_mnist	dinov2_small	0.985±0.001	0.964	0.971	0.972	0.979	0.974
vision_fashion_mnist	metaclip_b16	0.984±0.000	0.965	0.985	0.987	0.981	0.973

continued on next page

Table 8 continued from previous page

Task	Encoder	Ours	kNN	Lin	MLP	ZS	XGB
vision_fashion_mnist	siglip_base	0.984±0.000	0.935	0.983	0.988	0.980	0.966
vision_fashion_mnist	vit_base_patch16_224	0.974±0.000	0.952	0.970	0.974	0.973	0.956
vision_mnist	beit_base_patch16_224	0.996±0.000	0.976	0.993	0.995	0.991	0.968
vision_mnist	dinov2_small	0.993±0.000	0.943	0.988	0.989	0.991	0.966
vision_mnist	metaclip_b16	0.994±0.001	0.958	0.995	0.995	0.992	0.963
vision_mnist	siglip_base	0.999±0.000	0.991	0.999	0.999	0.996	0.995
vision_mnist	vit_base_patch16_224	0.988±0.001	0.929	0.983	0.987	0.982	0.945
vision_pneumoniamnist	beit_base_patch16_224	0.960±0.000	0.920	0.956	0.953	0.950	0.953
vision_pneumoniamnist	dinov2_small	0.980±0.001	0.956	0.976	0.977	0.975	0.971
vision_pneumoniamnist	metaclip_b16	0.954±0.002	0.923	0.951	0.946	0.958	0.953
vision_pneumoniamnist	siglip_base	0.958±0.001	0.913	0.959	0.961	0.957	0.947
vision_pneumoniamnist	vit_base_patch16_224	0.969±0.001	0.941	0.971	0.970	0.967	0.956
vision_retinamnist	beit_base_patch16_224	0.766±0.006	0.717	0.728	0.730	0.748	0.748
vision_retinamnist	dinov2_small	0.739±0.011	0.661	0.698	0.718	0.731	0.715
vision_retinamnist	metaclip_b16	0.715±0.009	0.646	0.686	0.706	0.695	0.708
vision_retinamnist	siglip_base	0.738±0.008	0.639	0.703	0.681	0.718	0.716
vision_retinamnist	vit_base_patch16_224	0.742±0.011	0.662	0.693	0.708	0.717	0.707
vision_svhn	beit_base_patch16_224	0.783±0.021	0.608	0.758	0.763	0.761	0.665
vision_svhn	dinov2_small	0.699±0.022	0.580	0.705	0.725	0.695	0.600
vision_svhn	metaclip_b16	0.733±0.014	0.636	0.758	0.764	0.717	0.663
vision_svhn	siglip_base	0.882±0.006	0.754	0.866	0.884	0.884	0.829
vision_svhn	vit_base_patch16_224	0.672±0.022	0.597	0.648	0.596	0.642	0.598
Text (44 pairs)							
text_agnews	arctic_embed_m_v2	0.975±0.001	0.955	0.956	0.972	0.973	0.953
text_agnews	bge_large_en_v1_5	0.973±0.001	0.953	0.953	0.970	0.972	0.960
text_agnews	bge_m3	0.972±0.000	0.947	0.953	0.961	0.968	0.961
text_agnews	nomic_embed_v1_5	0.977±0.000	0.953	0.957	0.974	0.978	0.959
text_financial_phrasebank	arctic_embed_m_v2	0.967±0.002	0.910	0.965	0.972	0.962	0.943
text_financial_phrasebank	bge_large_en_v1_5	0.986±0.001	0.964	0.983	0.987	0.975	0.960
text_financial_phrasebank	bge_m3	0.980±0.001	0.939	0.983	0.982	0.968	0.953
text_financial_phrasebank	nomic_embed_v1_5	0.980±0.001	0.945	0.982	0.983	0.978	0.959
text_imdb	arctic_embed_m_v2	0.989±0.000	0.954	0.980	0.987	0.988	0.978
text_imdb	bge_large_en_v1_5	0.992±0.000	0.972	0.988	0.990	0.988	0.986
text_imdb	bge_m3	0.976±0.000	0.930	0.960	0.967	0.975	0.960
text_imdb	nomic_embed_v1_5	0.983±0.000	0.950	0.972	0.973	0.979	0.960
text_rotten_tomatoes	arctic_embed_m_v2	0.929±0.001	0.877	0.891	0.931	0.927	0.911
text_rotten_tomatoes	bge_large_en_v1_5	0.942±0.001	0.916	0.927	0.941	0.943	0.933
text_rotten_tomatoes	bge_m3	0.926±0.001	0.899	0.884	0.923	0.920	0.916
text_rotten_tomatoes	nomic_embed_v1_5	0.916±0.001	0.870	0.886	0.912	0.909	0.902
text_sst2	arctic_embed_m_v2	0.966±0.001	0.917	0.916	0.959	0.947	0.938
text_sst2	bge_large_en_v1_5	0.985±0.000	0.975	0.981	0.981	0.977	0.970
text_sst2	bge_m3	0.977±0.001	0.947	0.954	0.969	0.968	0.966
text_sst2	nomic_embed_v1_5	0.969±0.001	0.923	0.943	0.963	0.961	0.945
text_tweet_emotion	arctic_embed_m_v2	0.892±0.002	0.787	0.823	0.872	0.894	0.850
text_tweet_emotion	bge_large_en_v1_5	0.926±0.001	0.845	0.872	0.916	0.918	0.914
text_tweet_emotion	bge_m3	0.899±0.003	0.810	0.870	0.897	0.885	0.879
text_tweet_emotion	nomic_embed_v1_5	0.912±0.003	0.834	0.842	0.904	0.904	0.900
text_tweet_hate	arctic_embed_m_v2	0.818±0.001	0.784	0.721	0.815	0.811	0.796
text_tweet_hate	bge_large_en_v1_5	0.827±0.001	0.790	0.767	0.831	0.813	0.808
text_tweet_hate	bge_m3	0.811±0.001	0.791	0.748	0.821	0.810	0.798
text_tweet_hate	nomic_embed_v1_5	0.818±0.001	0.804	0.756	0.822	0.811	0.808
text_tweet_irony	arctic_embed_m_v2	0.681±0.009	0.608	0.612	0.699	0.661	0.654
text_tweet_irony	bge_large_en_v1_5	0.712±0.001	0.670	0.640	0.663	0.682	0.690
text_tweet_irony	bge_m3	0.722±0.006	0.643	0.654	0.718	0.713	0.698
text_tweet_irony	nomic_embed_v1_5	0.686±0.003	0.631	0.697	0.676	0.656	0.659

continued on next page

Table 8 continued from previous page

Task	Encoder	Ours	kNN	Lin	MLP	ZS	XGB
text_tweet_offensive	arctic_embed_m_v2	0.758±0.008	0.639	0.698	0.756	0.749	0.716
text_tweet_offensive	bge_large_en_v1_5	0.781±0.004	0.722	0.714	0.781	0.779	0.763
text_tweet_offensive	bge_m3	0.799±0.002	0.708	0.741	0.780	0.794	0.763
text_tweet_offensive	nomic_embed_v1_5	0.800±0.000	0.667	0.729	0.800	0.787	0.770
text_tweet_sentiment	arctic_embed_m_v2	0.758±0.004	0.635	0.724	0.749	0.771	0.729
text_tweet_sentiment	bge_large_en_v1_5	0.818±0.002	0.663	0.748	0.808	0.800	0.768
text_tweet_sentiment	bge_m3	0.820±0.001	0.680	0.783	0.809	0.805	0.772
text_tweet_sentiment	nomic_embed_v1_5	0.796±0.003	0.667	0.747	0.778	0.791	0.761
text_yelp_polarity	arctic_embed_m_v2	0.989±0.000	0.924	0.982	0.986	0.987	0.976
text_yelp_polarity	bge_large_en_v1_5	0.993±0.000	0.950	0.978	0.993	0.990	0.987
text_yelp_polarity	bge_m3	0.990±0.001	0.964	0.981	0.991	0.985	0.987
text_yelp_polarity	nomic_embed_v1_5	0.998±0.000	0.978	0.987	0.996	0.998	0.995
Audio (28 pairs)							
audio_crema_d	ast_audioset	0.825±0.004	0.720	0.794	0.804	0.811	0.782
audio_crema_d	clap_general	0.811±0.005	0.721	0.776	0.798	0.798	0.778
audio_crema_d	wavlm_base_plus	0.849±0.006	0.703	0.828	0.838	0.825	0.793
audio_crema_d	whisper_large_v3_turbo	0.891±0.004	0.718	0.893	0.891	0.857	0.838
audio_emodb	ast_audioset	0.940±0.002	0.901	0.929	0.933	0.926	0.841
audio_emodb	clap_general	0.938±0.006	0.903	0.928	0.933	0.923	0.863
audio_emodb	wavlm_base_plus	0.930±0.010	0.821	0.920	0.918	0.921	0.880
audio_emodb	whisper_large_v3_turbo	0.975±0.004	0.762	0.971	0.980	0.962	0.879
audio_esc10	ast_audioset	1.000±0.000	1.000	1.000	1.000	1.000	0.997
audio_esc10	clap_general	1.000±0.000	1.000	1.000	1.000	1.000	0.999
audio_esc10	wavlm_base_plus	0.996±0.000	0.968	0.992	0.991	0.992	0.990
audio_esc10	whisper_large_v3_turbo	0.997±0.001	0.961	0.996	0.997	0.990	0.982
audio_gtzan	ast_audioset	0.983±0.001	0.952	0.979	0.984	0.978	0.951
audio_gtzan	clap_general	0.988±0.001	0.973	0.980	0.984	0.983	0.955
audio_gtzan	wavlm_base_plus	0.935±0.001	0.863	0.922	0.932	0.912	0.873
audio_gtzan	whisper_large_v3_turbo	0.934±0.002	0.857	0.943	0.944	0.907	0.886
audio_ravdess	ast_audioset	0.909±0.002	0.825	0.900	0.904	0.882	0.848
audio_ravdess	clap_general	0.918±0.004	0.815	0.907	0.910	0.901	0.874
audio_ravdess	wavlm_base_plus	0.916±0.004	0.752	0.897	0.901	0.882	0.837
audio_ravdess	whisper_large_v3_turbo	0.953±0.001	0.834	0.944	0.947	0.933	0.908
audio_urbansound8k	ast_audioset	0.995±0.000	0.979	0.991	0.994	0.991	0.983
audio_urbansound8k	clap_general	0.994±0.001	0.980	0.988	0.989	0.991	0.985
audio_urbansound8k	wavlm_base_plus	0.944±0.000	0.850	0.904	0.918	0.928	0.906
audio_urbansound8k	whisper_large_v3_turbo	0.961±0.001	0.892	0.946	0.951	0.933	0.935
audio_voxforge_lang	ast_audioset	0.602±0.020	0.600	0.635	0.619	0.577	0.606
audio_voxforge_lang	clap_general	0.762±0.014	0.715	0.710	0.732	0.722	0.591
audio_voxforge_lang	wavlm_base_plus	0.810±0.030	0.768	0.861	0.867	0.810	0.706
audio_voxforge_lang	whisper_large_v3_turbo	1.000±0.000	1.000	1.000	1.000	1.000	0.943
Graphs (40 pairs)							
graph_aids	pyg_gat	0.961±0.008	0.785	0.752	0.873	0.956	0.907
graph_aids	pyg_gcn	0.980±0.001	0.882	0.870	0.946	0.942	0.962
graph_aids	pyg_gin	1.000±0.000	0.987	1.000	1.000	1.000	1.000
graph_aids	pyg_sage	0.974±0.002	0.877	0.849	0.946	0.904	0.955
graph_bzr	pyg_gat	0.881±0.024	0.798	0.773	0.706	0.834	0.796
graph_bzr	pyg_gcn	0.931±0.018	0.892	0.913	0.943	0.587	0.872
graph_bzr	pyg_gin	0.920±0.021	0.814	0.858	0.851	0.803	0.832
graph_bzr	pyg_sage	0.933±0.011	0.871	0.844	0.895	0.601	0.894
graph_cox2	pyg_gat	0.769±0.017	0.657	0.814	0.763	0.855	0.607
graph_cox2	pyg_gcn	0.807±0.005	0.677	0.773	0.662	0.559	0.781
graph_cox2	pyg_gin	0.827±0.017	0.697	0.740	0.688	0.731	0.830
graph_cox2	pyg_sage	0.806±0.019	0.760	0.862	0.786	0.635	0.764
graph_dd	pyg_gat	0.817±0.005	0.658	0.599	0.605	0.812	0.768
graph_dd	pyg_gcn	0.847±0.008	0.700	0.663	0.625	0.808	0.769
graph_dd	pyg_gin	0.830±0.003	0.808	0.756	0.821	0.646	0.776
graph_dd	pyg_sage	0.833±0.004	0.641	0.531	0.611	0.759	0.794
graph_dhfr	pyg_gat	0.720±0.026	0.661	0.557	0.491	0.755	0.661
graph_dhfr	pyg_gcn	0.784±0.020	0.672	0.721	0.543	0.598	0.703
graph_dhfr	pyg_gin	0.779±0.009	0.695	0.759	0.693	0.678	0.726
graph_dhfr	pyg_sage	0.761±0.021	0.689	0.776	0.633	0.513	0.697

continued on next page

Table 8 continued from previous page

Task	Encoder	Ours	kNN	Lin	MLP	ZS	XGB
graph_enzymes	pyg_gat	0.708±0.029	0.637	0.501	0.643	0.667	0.634
graph_enzymes	pyg_gcn	0.730±0.050	0.685	0.605	0.560	0.472	0.707
graph_enzymes	pyg_gin	0.726±0.028	0.685	0.636	0.624	0.545	0.705
graph_enzymes	pyg_sage	0.714±0.041	0.658	0.548	0.588	0.517	0.673
graph_mutag	pyg_gat	0.772±0.051	0.593	0.778	0.654	0.664	0.630
graph_mutag	pyg_gcn	0.751±0.071	0.753	0.759	0.741	0.642	0.747
graph_mutag	pyg_gin	0.891±0.013	0.929	0.920	0.889	0.886	0.858
graph_mutag	pyg_sage	0.737±0.089	0.654	0.759	0.667	0.475	0.747
graph_nci1	pyg_gat	0.734±0.014	0.656	0.694	0.683	0.729	0.686
graph_nci1	pyg_gcn	0.831±0.002	0.750	0.757	0.738	0.495	0.801
graph_nci1	pyg_gin	0.853±0.001	0.809	0.794	0.808	0.577	0.827
graph_nci1	pyg_sage	0.830±0.003	0.766	0.764	0.796	0.630	0.813
graph_proteins	pyg_gat	0.795±0.003	0.807	0.742	0.764	0.816	0.799
graph_proteins	pyg_gcn	0.827±0.015	0.742	0.719	0.738	0.474	0.760
graph_proteins	pyg_gin	0.846±0.010	0.803	0.796	0.834	0.600	0.840
graph_proteins	pyg_sage	0.834±0.014	0.760	0.788	0.768	0.381	0.778
graph_ptc_mr	pyg_gat	0.537±0.029	0.575	0.503	0.481	0.612	0.611
graph_ptc_mr	pyg_gcn	0.568±0.022	0.586	0.532	0.520	0.407	0.561
graph_ptc_mr	pyg_gin	0.624±0.011	0.649	0.551	0.598	0.409	0.561
graph_ptc_mr	pyg_sage	0.530±0.006	0.582	0.480	0.487	0.433	0.631
Time Series (36 pairs)							
ts_chinatown	chronos_t5_small	0.958±0.007	0.913	0.968	0.962	0.958	0.883
ts_chinatown	minirocket	0.995±0.002	0.963	0.996	0.998	0.989	0.989
ts_chinatown	rocket	0.999±0.001	0.984	1.000	0.996	0.996	0.967
ts_ecg200	chronos_t5_small	0.650±0.000	0.712	0.575	0.575	0.675	0.500
ts_ecg200	minirocket	0.725±0.000	0.600	0.725	0.725	0.675	0.625
ts_ecg200	rocket	0.750±0.000	0.712	0.750	0.775	0.725	0.650
ts_electric_devices	chronos_t5_small	0.980±0.002	0.877	0.977	0.978	0.978	0.962
ts_electric_devices	minirocket	0.989±0.000	0.941	0.984	0.988	0.992	0.991
ts_electric_devices	rocket	0.979±0.000	0.956	0.985	0.991	0.992	0.991
ts_ford_a	chronos_t5_small	0.960±0.003	0.872	0.962	0.967	0.950	0.948
ts_ford_a	minirocket	0.974±0.000	0.940	0.978	0.982	0.991	0.987
ts_ford_a	rocket	0.979±0.000	0.927	0.981	0.984	0.984	0.985
ts_ford_b	chronos_t5_small	0.926±0.004	0.834	0.920	0.923	0.919	0.903
ts_ford_b	minirocket	0.950±0.001	0.929	0.965	0.971	0.975	0.972
ts_ford_b	rocket	0.963±0.002	0.916	0.966	0.971	0.965	0.964
ts_gunpoint	chronos_t5_small	1.000±0.000	0.611	0.889	0.889	0.889	1.000
ts_gunpoint	minirocket	1.000±0.000	1.000	1.000	1.000	1.000	1.000
ts_gunpoint	rocket	1.000±0.000	1.000	1.000	1.000	1.000	0.944
ts_italy_power	chronos_t5_small	0.983±0.024	1.000	1.000	1.000	1.000	1.000
ts_italy_power	minirocket	1.000±0.000	1.000	1.000	1.000	1.000	1.000
ts_italy_power	rocket	1.000±0.000	1.000	1.000	1.000	1.000	1.000
ts_plane	chronos_t5_small	1.000±0.000	1.000	1.000	1.000	1.000	0.932
ts_plane	minirocket	1.000±0.000	1.000	1.000	1.000	1.000	0.981
ts_plane	rocket	1.000±0.000	1.000	1.000	1.000	1.000	0.978
ts_sony_aibo	chronos_t5_small	0.977±0.002	0.946	0.979	0.978	0.980	0.937
ts_sony_aibo	minirocket	1.000±0.000	0.992	1.000	1.000	0.999	0.995
ts_sony_aibo	rocket	1.000±0.000	1.000	1.000	1.000	1.000	1.000
ts_strawberry	chronos_t5_small	0.968±0.001	0.946	0.958	0.950	0.950	0.938
ts_strawberry	minirocket	0.981±0.003	0.965	0.976	0.979	0.980	0.970
ts_strawberry	rocket	0.990±0.002	0.953	0.985	0.984	0.982	0.965
ts_two_lead_ecg	chronos_t5_small	0.999±0.000	0.994	0.996	0.998	0.999	0.999
ts_two_lead_ecg	minirocket	1.000±0.000	1.000	1.000	1.000	1.000	1.000
ts_two_lead_ecg	rocket	1.000±0.000	0.996	1.000	1.000	1.000	1.000
ts_wafer	chronos_t5_small	0.996±0.001	0.996	0.994	0.994	0.998	0.986
ts_wafer	minirocket	1.000±0.000	1.000	1.000	1.000	1.000	1.000
ts_wafer	rocket	1.000±0.000	1.000	1.000	1.000	1.000	1.000

Table 9: Per-(task, encoder)-pair generalization regime and ROC-AUC for each of the three independently generated partitions (seeds 123, 456, 789). ID = In-Distribution, NT = Novel Task, NE = Novel Encoder, JG = Joint Generalization. A pair's regime can differ across seeds since the held-out task/encoder sets are re-sampled per partition.

Task	Encoder	Seed 123		Seed 456		Seed 789	
		Reg.	AUC	Reg.	AUC	Reg.	AUC
Molecules (50 pairs)							
tdc_ames	chemberta_77m_mtr	ID	0.840	ID	0.841	ID	0.844
tdc_ames	chemberta_v2	ID	0.797	ID	0.799	NE	0.803
tdc_ames	chemberta_zinc_base	ID	0.774	ID	0.780	ID	0.772
tdc_ames	pubchem10m_bpe	NE	0.852	NE	0.852	ID	0.848

continued on next page

Table 9 continued from previous page

Task	Encoder	Seed 123		Seed 456		Seed 789	
		Reg.	AUC	Reg.	AUC	Reg.	AUC
tdc_ames	smiles_bert	ID	0.771	ID	0.769	ID	0.773
tdc_b3db_classification	chemberta_77m_mtr	ID	0.937	ID	0.938	NT	0.944
tdc_b3db_classification	chemberta_v2	ID	0.936	ID	0.936	JG	0.936
tdc_b3db_classification	chemberta_zinc_base	ID	0.930	ID	0.925	NT	0.927
tdc_b3db_classification	pubchem10m_bpe	NE	0.935	NE	0.936	NT	0.939
tdc_b3db_classification	smiles_bert	ID	0.943	ID	0.938	NT	0.940
tdc_bbb_martins	chemberta_77m_mtr	ID	0.915	ID	0.916	ID	0.919
tdc_bbb_martins	chemberta_v2	ID	0.899	ID	0.900	NE	0.903
tdc_bbb_martins	chemberta_zinc_base	ID	0.889	ID	0.887	ID	0.889
tdc_bbb_martins	pubchem10m_bpe	NE	0.915	NE	0.910	ID	0.910
tdc_bbb_martins	smiles_bert	ID	0.907	ID	0.900	ID	0.911
tdc_bioavailability_ma	chemberta_77m_mtr	ID	0.673	ID	0.662	ID	0.646
tdc_bioavailability_ma	chemberta_v2	ID	0.704	ID	0.668	NE	0.711
tdc_bioavailability_ma	chemberta_zinc_base	ID	0.699	ID	0.663	ID	0.683
tdc_bioavailability_ma	pubchem10m_bpe	NE	0.708	NE	0.691	ID	0.706
tdc_bioavailability_ma	smiles_bert	ID	0.695	ID	0.644	ID	0.689
tdc_cyp1a2_veith	chemberta_77m_mtr	ID	0.909	ID	0.906	ID	0.908
tdc_cyp1a2_veith	chemberta_v2	ID	0.909	ID	0.909	NE	0.910
tdc_cyp1a2_veith	chemberta_zinc_base	ID	0.906	ID	0.907	ID	0.907
tdc_cyp1a2_veith	pubchem10m_bpe	NE	0.914	NE	0.915	ID	0.915
tdc_cyp1a2_veith	smiles_bert	ID	0.880	ID	0.879	ID	0.881
tdc_cyp2d6_veith	chemberta_77m_mtr	ID	0.849	NT	0.836	ID	0.845
tdc_cyp2d6_veith	chemberta_v2	ID	0.816	NT	0.818	NE	0.821
tdc_cyp2d6_veith	chemberta_zinc_base	ID	0.810	NT	0.812	ID	0.810
tdc_cyp2d6_veith	pubchem10m_bpe	NE	0.839	JG	0.841	ID	0.849
tdc_cyp2d6_veith	smiles_bert	ID	0.764	NT	0.753	ID	0.771
tdc_dili	chemberta_77m_mtr	ID	0.855	NT	0.839	ID	0.852
tdc_dili	chemberta_v2	ID	0.882	NT	0.889	NE	0.875
tdc_dili	chemberta_zinc_base	ID	0.879	NT	0.865	ID	0.868
tdc_dili	pubchem10m_bpe	NE	0.923	JG	0.918	ID	0.922
tdc_dili	smiles_bert	ID	0.863	NT	0.864	ID	0.858
tdc_herg	chemberta_77m_mtr	ID	0.829	ID	0.810	ID	0.851
tdc_herg	chemberta_v2	ID	0.830	ID	0.846	NE	0.858
tdc_herg	chemberta_zinc_base	ID	0.852	ID	0.845	ID	0.831
tdc_herg	pubchem10m_bpe	NE	0.795	NE	0.773	ID	0.798
tdc_herg	smiles_bert	ID	0.733	ID	0.750	ID	0.746
tdc_hia_hou	chemberta_77m_mtr	ID	0.975	ID	0.975	ID	0.971
tdc_hia_hou	chemberta_v2	ID	0.973	ID	0.975	NE	0.972
tdc_hia_hou	chemberta_zinc_base	ID	0.937	ID	0.935	ID	0.925
tdc_hia_hou	pubchem10m_bpe	NE	0.976	NE	0.977	ID	0.979
tdc_hia_hou	smiles_bert	ID	0.984	ID	0.979	ID	0.981
tdc_pgp_broccatelli	chemberta_77m_mtr	NT	0.949	ID	0.947	NT	0.952
tdc_pgp_broccatelli	chemberta_v2	NT	0.914	ID	0.914	JG	0.912
tdc_pgp_broccatelli	chemberta_zinc_base	NT	0.900	ID	0.913	NT	0.893
tdc_pgp_broccatelli	pubchem10m_bpe	JG	0.942	NE	0.939	NT	0.942

continued on next page

Table 9 continued from previous page

Task	Encoder	Seed 123		Seed 456		Seed 789	
		Reg.	AUC	Reg.	AUC	Reg.	AUC
tdc_pgp_broccatelli	smiles_bert	NT	0.904	ID	0.900	NT	0.899
Proteins (60 pairs)							
prot_allergenic	ankh_base	NE	0.976	ID	0.974	ID	0.976
prot_allergenic	esm2_150m	ID	0.979	ID	0.980	ID	0.980
prot_allergenic	esm2_650m	ID	0.982	ID	0.981	ID	0.982
prot_allergenic	prot_bert	ID	0.979	NE	0.979	NE	0.980
prot_anticancer_peptide	ankh_base	JG	0.957	ID	0.959	ID	0.954
prot_anticancer_peptide	esm2_150m	NT	0.963	ID	0.965	ID	0.962
prot_anticancer_peptide	esm2_650m	NT	0.953	ID	0.953	ID	0.954
prot_anticancer_peptide	prot_bert	NT	0.932	NE	0.934	NE	0.939
prot_cloning_stage	ankh_base	NE	0.757	ID	0.737	ID	0.758
prot_cloning_stage	esm2_150m	ID	0.755	ID	0.744	ID	0.753
prot_cloning_stage	esm2_650m	ID	0.741	ID	0.736	ID	0.745
prot_cloning_stage	prot_bert	ID	0.738	NE	0.722	NE	0.732
prot_deeploc_binary	ankh_base	NE	0.987	ID	0.987	ID	0.989
prot_deeploc_binary	esm2_150m	ID	0.994	ID	0.992	ID	0.993
prot_deeploc_binary	esm2_650m	ID	0.994	ID	0.991	ID	0.993
prot_deeploc_binary	prot_bert	ID	0.989	NE	0.987	NE	0.989
prot_deeploc_multi	ankh_base	JG	0.972	ID	0.971	ID	0.962
prot_deeploc_multi	esm2_150m	NT	0.969	ID	0.970	ID	0.961
prot_deeploc_multi	esm2_650m	NT	0.970	ID	0.972	ID	0.964
prot_deeploc_multi	prot_bert	NT	0.955	NE	0.963	NE	0.952
prot_hemolytic_peptide	ankh_base	NE	0.730	ID	0.713	ID	0.737
prot_hemolytic_peptide	esm2_150m	ID	0.740	ID	0.724	ID	0.731
prot_hemolytic_peptide	esm2_650m	ID	0.728	ID	0.707	ID	0.728
prot_hemolytic_peptide	prot_bert	ID	0.692	NE	0.674	NE	0.690
prot_material_production	ankh_base	NE	0.829	ID	0.823	NT	0.824
prot_material_production	esm2_150m	ID	0.836	ID	0.836	NT	0.835
prot_material_production	esm2_650m	ID	0.838	ID	0.840	NT	0.838
prot_material_production	prot_bert	ID	0.822	NE	0.824	JG	0.824
prot_metal_ion	ankh_base	NE	0.894	NT	0.890	ID	0.891
prot_metal_ion	esm2_150m	ID	0.885	NT	0.880	ID	0.881
prot_metal_ion	esm2_650m	ID	0.884	NT	0.880	ID	0.881
prot_metal_ion	prot_bert	ID	0.872	JG	0.868	NE	0.865
prot_nonfouling_peptide	ankh_base	NE	0.939	ID	0.937	ID	0.939
prot_nonfouling_peptide	esm2_150m	ID	0.938	ID	0.938	ID	0.938
prot_nonfouling_peptide	esm2_650m	ID	0.936	ID	0.936	ID	0.938
prot_nonfouling_peptide	prot_bert	ID	0.928	NE	0.927	NE	0.928
prot_remote_homology	ankh_base	NE	0.959	NT	0.961	ID	0.970
prot_remote_homology	esm2_150m	ID	0.969	NT	0.970	ID	0.974
prot_remote_homology	esm2_650m	ID	0.964	NT	0.966	ID	0.977
prot_remote_homology	prot_bert	ID	0.950	JG	0.948	NE	0.951
prot_sabdab_chen	ankh_base	NE	0.895	ID	0.897	NT	0.882

continued on next page

Table 9 continued from previous page

Task	Encoder	Seed 123		Seed 456		Seed 789	
		Reg.	AUC	Reg.	AUC	Reg.	AUC
prot_sabdab_chen	esm2_150m	ID	0.893	ID	0.888	NT	0.887
prot_sabdab_chen	esm2_650m	ID	0.889	ID	0.875	NT	0.879
prot_sabdab_chen	prot_bert	ID	0.888	NE	0.883	JG	0.871
prot_solubility	ankh_base	NE	0.735	ID	0.738	NT	0.730
prot_solubility	esm2_150m	ID	0.658	ID	0.661	NT	0.670
prot_solubility	esm2_650m	ID	0.677	ID	0.676	NT	0.695
prot_solubility	prot_bert	ID	0.669	NE	0.658	JG	0.661
prot_sorting_signal	ankh_base	NE	0.966	ID	0.965	ID	0.963
prot_sorting_signal	esm2_150m	ID	0.972	ID	0.969	ID	0.971
prot_sorting_signal	esm2_650m	ID	0.965	ID	0.964	ID	0.966
prot_sorting_signal	prot_bert	ID	0.965	NE	0.959	NE	0.967
prot_temperature_stability	ankh_base	JG	0.952	NT	0.947	ID	0.952
prot_temperature_stability	esm2_150m	NT	0.983	NT	0.983	ID	0.984
prot_temperature_stability	esm2_650m	NT	0.988	NT	0.987	ID	0.989
prot_temperature_stability	prot_bert	NT	0.978	JG	0.978	NE	0.978
prot_toxin_peptide	ankh_base	NE	0.937	ID	0.937	ID	0.936
prot_toxin_peptide	esm2_150m	ID	0.938	ID	0.934	ID	0.937
prot_toxin_peptide	esm2_650m	ID	0.940	ID	0.938	ID	0.940
prot_toxin_peptide	prot_bert	ID	0.909	NE	0.906	NE	0.912
Vision (50 pairs)							
vision_beans	beit_base_patch16_224	NE	0.988	ID	0.991	ID	0.990
vision_beans	dinov2_small	ID	0.977	ID	0.981	ID	0.979
vision_beans	metaclip_b16	ID	0.994	NE	0.995	ID	0.995
vision_beans	siglip_base	ID	0.984	ID	0.986	ID	0.984
vision_beans	vit_base_patch16_224	ID	0.980	ID	0.981	NE	0.982
vision_bloodmnist	beit_base_patch16_224	NE	0.983	ID	0.984	ID	0.985
vision_bloodmnist	dinov2_small	ID	0.979	ID	0.979	ID	0.979
vision_bloodmnist	metaclip_b16	ID	0.939	NE	0.939	ID	0.939
vision_bloodmnist	siglip_base	ID	0.964	ID	0.962	ID	0.964
vision_bloodmnist	vit_base_patch16_224	ID	0.982	ID	0.982	NE	0.982
vision_breastmnist	beit_base_patch16_224	NE	0.796	ID	0.813	ID	0.814
vision_breastmnist	dinov2_small	ID	0.827	ID	0.815	ID	0.832
vision_breastmnist	metaclip_b16	ID	0.761	NE	0.748	ID	0.770
vision_breastmnist	siglip_base	ID	0.816	ID	0.812	ID	0.822
vision_breastmnist	vit_base_patch16_224	ID	0.750	ID	0.747	NE	0.765
vision_cifar10	beit_base_patch16_224	NE	1.000	NT	1.000	ID	1.000
vision_cifar10	dinov2_small	ID	0.999	NT	0.999	ID	1.000
vision_cifar10	metaclip_b16	ID	0.998	JG	0.999	ID	0.998
vision_cifar10	siglip_base	ID	1.000	NT	1.000	ID	1.000
vision_cifar10	vit_base_patch16_224	ID	1.000	NT	1.000	NE	1.000
vision_dermamnist	beit_base_patch16_224	NE	0.872	ID	0.859	ID	0.863
vision_dermamnist	dinov2_small	ID	0.875	ID	0.869	ID	0.867
vision_dermamnist	metaclip_b16	ID	0.825	NE	0.801	ID	0.830
vision_dermamnist	siglip_base	ID	0.884	ID	0.867	ID	0.878
vision_dermamnist	vit_base_patch16_224	ID	0.857	ID	0.849	NE	0.848

continued on next page

Table 9 continued from previous page

Task	Encoder	Seed 123		Seed 456		Seed 789	
		Reg.	AUC	Reg.	AUC	Reg.	AUC
vision_fashion_mnist	beit_base_patch16_224	JG	0.988	ID	0.988	NT	0.989
vision_fashion_mnist	dinov2_small	NT	0.985	ID	0.985	NT	0.984
vision_fashion_mnist	metaclip_b16	NT	0.984	NE	0.984	NT	0.984
vision_fashion_mnist	siglip_base	NT	0.984	ID	0.985	NT	0.984
vision_fashion_mnist	vit_base_patch16_224	NT	0.974	ID	0.974	JG	0.975
vision_mnist	beit_base_patch16_224	NE	0.996	ID	0.996	ID	0.995
vision_mnist	dinov2_small	ID	0.993	ID	0.993	ID	0.992
vision_mnist	metaclip_b16	ID	0.993	NE	0.994	ID	0.994
vision_mnist	siglip_base	ID	0.998	ID	0.999	ID	0.999
vision_mnist	vit_base_patch16_224	ID	0.988	ID	0.989	NE	0.987
vision_pneumoniamnist	beit_base_patch16_224	JG	0.959	ID	0.960	ID	0.960
vision_pneumoniamnist	dinov2_small	NT	0.978	ID	0.981	ID	0.980
vision_pneumoniamnist	metaclip_b16	NT	0.951	NE	0.955	ID	0.957
vision_pneumoniamnist	siglip_base	NT	0.956	ID	0.959	ID	0.958
vision_pneumoniamnist	vit_base_patch16_224	NT	0.967	ID	0.971	NE	0.969
vision_retinamnist	beit_base_patch16_224	NE	0.760	NT	0.765	ID	0.774
vision_retinamnist	dinov2_small	ID	0.723	NT	0.746	ID	0.748
vision_retinamnist	metaclip_b16	ID	0.703	JG	0.717	ID	0.726
vision_retinamnist	siglip_base	ID	0.726	NT	0.742	ID	0.746
vision_retinamnist	vit_base_patch16_224	ID	0.754	NT	0.727	NE	0.747
vision_svhn	beit_base_patch16_224	NE	0.805	ID	0.789	NT	0.755
vision_svhn	dinov2_small	ID	0.716	ID	0.713	NT	0.668
vision_svhn	metaclip_b16	ID	0.752	NE	0.730	NT	0.719
vision_svhn	siglip_base	ID	0.886	ID	0.885	NT	0.873
vision_svhn	vit_base_patch16_224	ID	0.677	ID	0.696	JG	0.644
Text (44 pairs)							
text_agnews	arctic_embed_m_v2	ID	0.976	ID	0.974	ID	0.976
text_agnews	bge_large_en_v1_5	NE	0.975	ID	0.972	ID	0.973
text_agnews	bge_m3	ID	0.972	NE	0.972	NE	0.971
text_agnews	nomic_embed_v1_5	ID	0.976	ID	0.977	ID	0.977
text_financial_phrasebank	arctic_embed_m_v2	ID	0.969	NT	0.964	ID	0.967
text_financial_phrasebank	bge_large_en_v1_5	NE	0.986	NT	0.985	ID	0.986
text_financial_phrasebank	bge_m3	ID	0.980	JG	0.981	NE	0.980
text_financial_phrasebank	nomic_embed_v1_5	ID	0.980	NT	0.979	ID	0.982
text_imdb	arctic_embed_m_v2	ID	0.989	ID	0.990	ID	0.989
text_imdb	bge_large_en_v1_5	NE	0.992	ID	0.992	ID	0.993
text_imdb	bge_m3	ID	0.976	NE	0.976	NE	0.976
text_imdb	nomic_embed_v1_5	ID	0.983	ID	0.983	ID	0.983
text_rotten_tomatoes	arctic_embed_m_v2	ID	0.927	ID	0.930	NT	0.929
text_rotten_tomatoes	bge_large_en_v1_5	NE	0.941	ID	0.943	NT	0.943
text_rotten_tomatoes	bge_m3	ID	0.925	NE	0.928	JG	0.926
text_rotten_tomatoes	nomic_embed_v1_5	ID	0.915	ID	0.915	NT	0.917

continued on next page

Table 9 continued from previous page

Task	Encoder	Seed 123		Seed 456		Seed 789	
		Reg.	AUC	Reg.	AUC	Reg.	AUC
text_sst2	arctic_embed_m_v2	ID	0.966	ID	0.965	ID	0.967
text_sst2	bge_large_en_v1_5	NE	0.985	ID	0.985	ID	0.985
text_sst2	bge_m3	ID	0.977	NE	0.978	NE	0.975
text_sst2	nomic_embed_v1_5	ID	0.969	ID	0.970	ID	0.967
text_tweet_emotion	arctic_embed_m_v2	ID	0.895	ID	0.890	ID	0.891
text_tweet_emotion	bge_large_en_v1_5	NE	0.928	ID	0.924	ID	0.926
text_tweet_emotion	bge_m3	ID	0.900	NE	0.895	NE	0.901
text_tweet_emotion	nomic_embed_v1_5	ID	0.913	ID	0.908	ID	0.915
text_tweet_hate	arctic_embed_m_v2	ID	0.817	ID	0.820	ID	0.817
text_tweet_hate	bge_large_en_v1_5	NE	0.829	ID	0.825	ID	0.827
text_tweet_hate	bge_m3	ID	0.811	NE	0.810	NE	0.813
text_tweet_hate	nomic_embed_v1_5	ID	0.817	ID	0.819	ID	0.819
text_tweet_irony	arctic_embed_m_v2	NT	0.684	ID	0.669	ID	0.690
text_tweet_irony	bge_large_en_v1_5	JG	0.714	ID	0.712	ID	0.712
text_tweet_irony	bge_m3	NT	0.718	NE	0.718	NE	0.730
text_tweet_irony	nomic_embed_v1_5	NT	0.682	ID	0.690	ID	0.686
text_tweet_offensive	arctic_embed_m_v2	ID	0.760	ID	0.748	ID	0.768
text_tweet_offensive	bge_large_en_v1_5	NE	0.778	ID	0.778	ID	0.786
text_tweet_offensive	bge_m3	ID	0.800	NE	0.802	NE	0.796
text_tweet_offensive	nomic_embed_v1_5	ID	0.799	ID	0.800	ID	0.800
text_tweet_sentiment	arctic_embed_m_v2	NT	0.761	ID	0.762	NT	0.753
text_tweet_sentiment	bge_large_en_v1_5	JG	0.817	ID	0.821	NT	0.816
text_tweet_sentiment	bge_m3	NT	0.819	NE	0.820	JG	0.820
text_tweet_sentiment	nomic_embed_v1_5	NT	0.795	ID	0.794	NT	0.800
text_yelp_polarity	arctic_embed_m_v2	ID	0.989	NT	0.989	ID	0.989
text_yelp_polarity	bge_large_en_v1_5	NE	0.994	NT	0.993	ID	0.993
text_yelp_polarity	bge_m3	ID	0.992	JG	0.988	NE	0.990
text_yelp_polarity	nomic_embed_v1_5	ID	0.999	NT	0.998	ID	0.998
Audio (28 pairs)							
audio_crema_d	ast_audioset	ID	0.829	JG	0.820	ID	0.826
audio_crema_d	clap_general	ID	0.812	NT	0.805	ID	0.817
audio_crema_d	wavlm_base_plus	NE	0.854	NT	0.840	ID	0.852
audio_crema_d	whisper_large_v3_turbo	ID	0.894	NT	0.885	NE	0.894
audio_emodb	ast_audioset	NT	0.939	NE	0.942	ID	0.939
audio_emodb	clap_general	NT	0.930	ID	0.945	ID	0.940
audio_emodb	wavlm_base_plus	JG	0.916	ID	0.936	ID	0.938
audio_emodb	whisper_large_v3_turbo	NT	0.972	ID	0.981	NE	0.972
audio_esc10	ast_audioset	ID	1.000	NE	1.000	ID	1.000
audio_esc10	clap_general	ID	1.000	ID	1.000	ID	1.000
audio_esc10	wavlm_base_plus	NE	0.996	ID	0.996	ID	0.996
audio_esc10	whisper_large_v3_turbo	ID	0.996	ID	0.997	NE	0.998
audio_gtzan	ast_audioset	ID	0.984	NE	0.982	ID	0.983
audio_gtzan	clap_general	ID	0.989	ID	0.987	ID	0.987
audio_gtzan	wavlm_base_plus	NE	0.934	ID	0.937	ID	0.934
audio_gtzan	whisper_large_v3_turbo	ID	0.932	ID	0.933	NE	0.937
audio_ravdess	ast_audioset	ID	0.906	NE	0.911	ID	0.911
audio_ravdess	clap_general	ID	0.913	ID	0.922	ID	0.920

continued on next page

Table 9 continued from previous page

Task	Encoder	Seed 123		Seed 456		Seed 789	
		Reg.	AUC	Reg.	AUC	Reg.	AUC
audio_ravdess	wavlm_base_plus	NE	0.911	ID	0.919	ID	0.918
audio_ravdess	whisper_large_v3_turbo	ID	0.951	ID	0.955	NE	0.953
audio_urbansound8k	ast_audioset	ID	0.994	NE	0.995	ID	0.995
audio_urbansound8k	clap_general	ID	0.995	ID	0.995	ID	0.994
audio_urbansound8k	wavlm_base_plus	NE	0.944	ID	0.944	ID	0.945
audio_urbansound8k	whisper_large_v3_turbo	ID	0.960	ID	0.963	NE	0.961
audio_voxforge_lang	ast_audioset	ID	0.626	NE	0.602	NT	0.577
audio_voxforge_lang	clap_general	ID	0.763	ID	0.779	NT	0.744
audio_voxforge_lang	wavlm_base_plus	NE	0.787	ID	0.853	NT	0.790
audio_voxforge_lang	whisper_large_v3_turbo	ID	1.000	ID	1.000	JG	1.000
Graphs (40 pairs)							
graph_aids	pyg_gat	ID	0.970	ID	0.960	NT	0.952
graph_aids	pyg_gcn	ID	0.981	NE	0.979	NT	0.979
graph_aids	pyg_gin	ID	1.000	ID	1.000	JG	1.000
graph_aids	pyg_sage	NE	0.975	ID	0.976	NT	0.971
graph_bzr	pyg_gat	ID	0.874	NT	0.913	ID	0.856
graph_bzr	pyg_gcn	ID	0.940	JG	0.949	ID	0.906
graph_bzr	pyg_gin	ID	0.936	NT	0.933	NE	0.890
graph_bzr	pyg_sage	NE	0.947	NT	0.933	ID	0.920
graph_cox2	pyg_gat	ID	0.786	ID	0.774	ID	0.747
graph_cox2	pyg_gcn	ID	0.814	NE	0.802	ID	0.804
graph_cox2	pyg_gin	ID	0.807	ID	0.824	NE	0.849
graph_cox2	pyg_sage	NE	0.828	ID	0.806	ID	0.783
graph_dd	pyg_gat	ID	0.823	ID	0.811	ID	0.816
graph_dd	pyg_gcn	ID	0.849	NE	0.836	ID	0.855
graph_dd	pyg_gin	ID	0.830	ID	0.833	NE	0.826
graph_dd	pyg_sage	NE	0.838	ID	0.827	ID	0.834
graph_dhfr	pyg_gat	ID	0.756	ID	0.706	NT	0.698
graph_dhfr	pyg_gcn	ID	0.812	NE	0.768	NT	0.771
graph_dhfr	pyg_gin	ID	0.780	ID	0.768	JG	0.789
graph_dhfr	pyg_sage	NE	0.790	ID	0.751	NT	0.743
graph_enzymes	pyg_gat	NT	0.674	ID	0.744	ID	0.707
graph_enzymes	pyg_gcn	NT	0.659	NE	0.768	ID	0.764
graph_enzymes	pyg_gin	NT	0.691	ID	0.759	NE	0.728
graph_enzymes	pyg_sage	JG	0.657	ID	0.738	ID	0.749
graph_mutag	pyg_gat	NT	0.704	ID	0.784	ID	0.827
graph_mutag	pyg_gcn	NT	0.667	NE	0.747	ID	0.840
graph_mutag	pyg_gin	NT	0.907	ID	0.889	NE	0.876
graph_mutag	pyg_sage	JG	0.624	ID	0.747	ID	0.840
graph_nci1	pyg_gat	ID	0.742	ID	0.746	ID	0.715
graph_nci1	pyg_gcn	ID	0.831	NE	0.833	ID	0.829
graph_nci1	pyg_gin	ID	0.851	ID	0.853	NE	0.855
graph_nci1	pyg_sage	NE	0.828	ID	0.834	ID	0.828
graph_proteins	pyg_gat	ID	0.795	NT	0.791	ID	0.799
graph_proteins	pyg_gcn	ID	0.837	JG	0.807	ID	0.839
graph_proteins	pyg_gin	ID	0.845	NT	0.835	NE	0.859
graph_proteins	pyg_sage	NE	0.829	NT	0.819	ID	0.853
graph_ptc_mr	pyg_gat	ID	0.517	ID	0.577	ID	0.515
graph_ptc_mr	pyg_gcn	ID	0.583	NE	0.583	ID	0.537
graph_ptc_mr	pyg_gin	ID	0.610	ID	0.626	NE	0.636
graph_ptc_mr	pyg_sage	NE	0.538	ID	0.525	ID	0.526
Time Series (36 pairs)							
ts_chinatown	chronos_t5_small	ID	0.956	NE	0.951	ID	0.968
ts_chinatown	minirocket	NE	0.994	ID	0.997	ID	0.995
ts_chinatown	rocket	ID	0.998	ID	0.998	ID	1.000
ts_ecg200	chronos_t5_small	ID	0.650	NE	0.650	ID	0.650
ts_ecg200	minirocket	NE	0.725	ID	0.725	ID	0.725
ts_ecg200	rocket	ID	0.750	ID	0.750	ID	0.750
ts_electric_devices	chronos_t5_small	NT	0.981	NE	0.978	ID	0.982
ts_electric_devices	minirocket	JG	0.989	ID	0.989	ID	0.989
ts_electric_devices	rocket	NT	0.979	ID	0.979	ID	0.979

continued on next page

Table 9 continued from previous page

Task	Encoder	Seed 123		Seed 456		Seed 789	
		Reg.	AUC	Reg.	AUC	Reg.	AUC
ts_ford_a	chronos_t5_small	ID	0.961	NE	0.956	ID	0.962
ts_ford_a	minirocket	NE	0.973	ID	0.974	ID	0.973
ts_ford_a	rocket	ID	0.979	ID	0.979	ID	0.979
ts_ford_b	chronos_t5_small	NT	0.920	NE	0.930	NT	0.927
ts_ford_b	minirocket	JG	0.951	ID	0.948	NT	0.952
ts_ford_b	rocket	NT	0.960	ID	0.966	NT	0.964
ts_gunpoint	chronos_t5_small	ID	1.000	NE	1.000	ID	1.000
ts_gunpoint	minirocket	NE	1.000	ID	1.000	ID	1.000
ts_gunpoint	rocket	ID	1.000	ID	1.000	ID	1.000
ts_italy_power	chronos_t5_small	ID	1.000	NE	1.000	ID	0.950
ts_italy_power	minirocket	NE	1.000	ID	1.000	ID	1.000
ts_italy_power	rocket	ID	1.000	ID	1.000	ID	1.000
ts_plane	chronos_t5_small	ID	1.000	NE	1.000	ID	1.000
ts_plane	minirocket	NE	1.000	ID	1.000	ID	1.000
ts_plane	rocket	ID	1.000	ID	1.000	ID	1.000
ts_sony_aibo	chronos_t5_small	ID	0.977	NE	0.975	ID	0.980
ts_sony_aibo	minirocket	NE	1.000	ID	1.000	ID	1.000
ts_sony_aibo	rocket	ID	1.000	ID	1.000	ID	1.000
ts_strawberry	chronos_t5_small	ID	0.968	NE	0.969	ID	0.967
ts_strawberry	minirocket	NE	0.980	ID	0.985	ID	0.978
ts_strawberry	rocket	ID	0.989	ID	0.988	ID	0.992
ts_two_lead_ecg	chronos_t5_small	ID	0.999	JG	0.999	ID	0.999
ts_two_lead_ecg	minirocket	NE	1.000	NT	0.999	ID	1.000
ts_two_lead_ecg	rocket	ID	1.000	NT	1.000	ID	1.000
ts_wafer	chronos_t5_small	ID	0.997	JG	0.995	NT	0.995
ts_wafer	minirocket	NE	1.000	NT	1.000	NT	1.000
ts_wafer	rocket	ID	1.000	NT	1.000	NT	1.000