WHEN DOES THE AUXILIARY ENCODER HELP IN MULTIMODAL MOLECULAR PRETRAINING?

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal molecular pretraining often improves graph representations by aligning them with a second view of the same molecule, such as a SMILES string or 3D conformation. These gains are usually attributed to information learned by the auxiliary encoder. We show that this explanation is often incomplete: in several settings, a frozen, randomly initialized auxiliary network recovers much of the benefit of a trained one. We test this in controlled experiments that keep the graph model and training procedure fixed while replacing the trained auxiliary encoder with an untrained network of the same architecture. Markedly, on QM9 and Alchemy, random SMILES targets reduce mean raw-feature linear-probe error by 5.0% and 3.0% relative to graph-only training, respectively. The effect varies by settings: trained targets matter more for 3D supervision on PCQM4Mv2, for longer training on QMugs, and under GraphMVP fine-tuning. Additional controls show that random graph encoders and fixed random codes can also help, while simple representation expansion does not explain the gains. Together, these results show that cross-modal improvements can arise even when the second encoder has learned no molecular information. A frozen random target should therefore be included as a baseline before attributing such gains to transferred cross-modal content.

1 INTRODUCTION

Training with several views of the same object can improve a model even when it uses only one view at test time. Images paired with sound or text, and molecular graphs paired with 3D geometry, are examples of this approach (Ngiam et al., 2011; Aytar et al., 2016; Desai & Johnson, 2021; Radford et al., 2021; Liu et al., 2022; Stärk et al., 2022; Huang et al., 2023). For molecules, the deployed model often receives only a graph of atoms and bonds. During training, a second encoder can also read the molecule's 3D structure, a SMILES string—a textual description of its graph (Weininger, 1988)—or other text (Liu et al., 2022; Stärk et al., 2022; Zhu et al., 2023; Liu et al., 2023a). Encouraging the two encoders to produce similar representations can help the graph model predict molecular properties.

It is natural to attribute this improvement to useful information learned by the second encoder. Yet that explanation leaves an important question open: *Can a second encoder improve a graph model without learning anything itself?* Our central contribution is to test this question with a simple control. We replace the trained auxiliary encoder with a randomly initialized network of the same architecture and keep its weights fixed. This network still processes a molecular input, but has never been trained. Comparing the two versions shows how much of the observed benefit is available without learning the auxiliary representation.

Learning from untrained targets has precedents in noise-as-targets, random network distillation, BEST-RQ, and random-teacher self-distillation (Bojanowski & Joulin, 2017; Burda et al., 2019; Chiu et al., 2022; Sarnthein et al., 2023). We bring this idea to multimodal molecular learning as a way to assess the contribution of auxiliary training. Figure 1 summarizes the framework.

An auxiliary encoder can help without being trained. In several of the molecular learning settings we study, random targets recover much of the improvement obtained with trained targets. Learned encoders retain an advantage elsewhere, and the comparison changes with the task and with how

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.

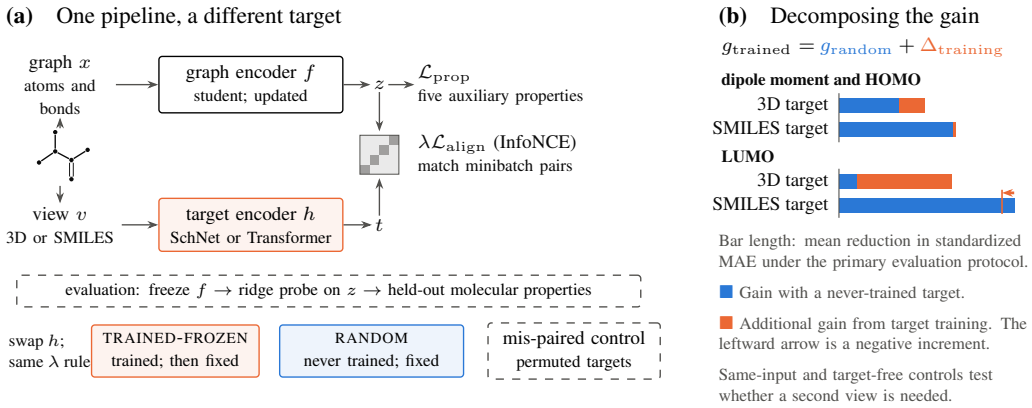


Figure 1: **How much of the gain requires a trained auxiliary encoder?** (a) During training, the graph model predicts molecular properties while learning to match a second view. We compare trained and frozen random auxiliary encoders of the same architecture; downstream prediction uses only graph features. (b) Mean gains over GRAPH-ONLY across eight QM9 cohorts. Blue shows the gain obtained with a random target; orange shows the additional gain from training it, which is negative for SMILES on LUMO. This split describes the observed gains without identifying their causes. Figures 2 and 7 show variation across cohorts.

the graph model is trained and evaluated. Random targets therefore provide a useful control for interpreting gains, rather than a general replacement for learned auxiliary encoders.

Our further controls show that some improvements do not require a second molecular view at all: targets computed from the graph itself, and even fixed random vectors, can improve prediction. Several helpful interventions also produce more varied graph features, but this change alone does not explain their predictive benefit. These findings motivate a practical baseline: evaluate a frozen random auxiliary encoder in the setting of interest before attributing the full improvement to learned cross-modal content.

1.1 Related work

Multimodal learning. GraphMVP (Liu et al., 2022) and 3D Infomax (Stärk et al., 2022) use geometry to train encoders for graph-only deployment. MoleculeSTM (Liu et al., 2023a) and MolCA (Liu et al., 2023b) connect molecular structure to text. Shared parameters can also improve unimodal models using unpaired multimodal data (Gupta et al., 2026). Benchmarks emphasize strong graph-only baselines (Sun et al., 2022; Praski et al., 2025; Khanna et al., 2025). Our random-target control retains the auxiliary network without training it.

Shared representations. The Platonic Representation Hypothesis proposes that increasingly capable models may learn shared representations across modalities (Huh et al., 2024); related work finds shared chemical organization in atomistic models (Li & Walsh, 2026). However, representation alignment need not track task performance (Tjandrasuwita et al., 2025), and similarity measures can show local agreement without a consistent global scaling trend (Gröger et al., 2026). We ask a complementary question: how much of the downstream gain requires the auxiliary encoder to have learned its representation?

Random teachers and features. Random network distillation (Burda et al., 2019), noise-as-targets (Bojanowski & Joulin, 2017), BYOL's random-target ablation (Grill et al., 2020), and BEST-RQ (Chiu et al., 2022) learn from untrained targets. Sarnthein et al. (2023) directly study transferable representations learned from random teachers. For molecules, SimSGT constructs reconstruction targets with fixed graph operations on the encoder's shared node embeddings (Liu et al., 2023c). We compare untrained and learned targets in a matched molecular setting and examine where the untrained targets fall short. Random-feature methods (Rahimi & Recht, 2007) provide the basis for the idealized account in Appendix I.

Representation geometry. Uniformity penalties (Wang & Isola, 2020) and VICReg's variance and covariance terms (Bardes et al., 2022) encourage features to spread out and carry less redundant information. We use them as controls that require no target network. Theoretical work relates contrastive learning to the views, augmentation graph and encoder architecture (Tian et al., 2020; HaoChen et al., 2021; Saunshi et al., 2022). Studies of dimensional collapse show that agreement between views can coexist with limited feature diversity (Jing et al., 2022). We measure this diversity through participation-ratio rank (Gao et al., 2017), alongside uniformity, similarity, and prediction error. Each captures a different property of the representation; none identifies the cause of a gain on its own.

2 THE RANDOM-TARGET CONTROL

We compare three ways to continue training the same graph model: on molecular properties alone, with a trained auxiliary encoder, or with an untrained auxiliary encoder. The last comparison tests how much of the auxiliary benefit remains when the second encoder has never learned from data.

2.1 TRAINING INTERVENTION

Figure 1a illustrates the training framework: the graph model learns from both molecular properties and an auxiliary representation. The graph encoder, which we call the *student*, maps atom and bond features x to a molecular representation $z = f(x)$. It uses a GINE-style architecture (Xu et al., 2019; Hu et al., 2020). The auxiliary encoder maps a second view v of the same molecule to a *target* representation $t = h(v)$. For 3D inputs it is a SchNet-style network that reads interatomic distances (Schütt et al., 2017); for SMILES it is a Transformer that reads the canonical SMILES string as a sequence of bytes (Weininger, 1988; Vaswani et al., 2017). SMILES describes the graph, whereas an equilibrium geometry computed by density functional theory (DFT) can carry information absent from the graph input. In both cases, the auxiliary view is used only for training. Downstream prediction uses the graph encoder alone.

We first train the graph encoder to predict five auxiliary molecular properties. Each experimental condition then starts from these same weights and continues property training, with or without an alignment loss. The alignment loss encourages graph and target representations of the same molecule to match, using other molecules in the batch as alternatives. With $\bar{z}$ and $\bar{t}$ denoting unit-normalized representations and τ a temperature, the objective is

$$\mathcal{L} = \mathcal{L}_{\text{prop}} + \lambda \mathcal{L}_{\text{align}}, \qquad \mathcal{L}_{\text{align}} = \tfrac{1}{2}\{\text{CE}(S) + \text{CE}(S^\top)\}, \qquad S_{ij} = \langle \bar{z}_i, \bar{t}_j \rangle / \tau, \tag{1}$$

where each cross-entropy identifies the matching molecule in a minibatch (van den Oord et al., 2018; Radford et al., 2021). We set λ once for each condition so that their initial alignment-to-property gradient ratio is the same across conditions. This common calibration rule can produce different weights for different targets; it does not keep the gradient ratio fixed later or make the optimization problems equally difficult. Appendix A gives the architectures and training details.

In TRAINED-FROZEN, the target is first trained on the same five properties and then frozen. In RANDOM, the same architecture is randomly initialized and frozen without training. The GRAPH-ONLY baseline omits alignment. Table 1 lists further controls: a target trained alongside the student, a target that reads the student's graph input, fixed random codes assigned to molecules, and penalties that require no target.

2.2 EVALUATION AND COMPARISONS

The evaluation path in Figure 1a uses only the graph encoder. We freeze its features and fit a ridge probe, a linear predictor with a penalty on its weights (Alain & Bengio, 2017). It predicts two properties withheld from encoder training: dipole moment and HOMO energy. The penalty is selected within the fitting pool, without using the test set. For each property, we divide mean absolute error by the standard deviation of its fitting labels, then average the two standardized errors. The primary protocol uses 1,200 graph-training updates after pretraining and 4,096 labels for fitting the probe. Smaller label budgets, longer training, and testing on a shared pool of larger molecules are secondary comparisons (Appendices B and H).

The controlled comparison uses a 19,456-molecule QM9 subset (Ramakrishnan et al., 2014) and eight *cohorts*, each with a data partition and model seed. Their test pools are disjoint, but their training pools overlap. Means and standard deviations across cohorts therefore describe variation within this study draw. The overlap also means that standard confidence intervals and tests across cohorts have no demonstrated coverage or error-rate guarantee here. Appendix A.1 reports these only as sensitivity analyses under additional assumptions, including Fieller intervals (Fieller, 1954) for ratios of gains and tests of whether random targets recover at least half the trained gain.

We define a target's *gain* as its error reduction over graph-only training, $g = e_{\mathrm{GO}} - e_{\mathrm{target}}$. The *recovered fraction* is the ratio of mean random-target and trained-target gains, $g_{\mathrm{random}}/g_{\mathrm{trained}}$. Figure 1b illustrates this comparison by splitting the trained gain into the part obtained with a random target and the additional gain from training. A fraction near one means the two observed gains are close; it does not establish equivalence. QM9 3D was the prespecified primary comparison. We applied its half-gain rule retrospectively to QM9 SMILES (Appendix A.2).

A separate confirmation tunes each condition on a dedicated molecular block, then evaluates it on three fresh, molecule-disjoint blocks per dataset (Section 3.2). Each block result averages three model seeds. We report all three block values and do not apply the cohort-level tests to them.

3 Experiments

3.1 Random SMILES targets yield similar mean gains on QM9

We first ask how much of a trained target's gain remains when the target has never been trained. On QM9 the answer depends on the view (Figure 2): random and trained SMILES targets reduce error on raw graph features by similar mean amounts, whereas a random 3D target helps less than its trained counterpart. The comparison fixes the graph architecture, starting checkpoint, minibatches, auxiliary-property objective, downstream probes and calibration rule. Within each row, only the target's training history changes under this shared calibration rule.

The five cohorts added after the pilot still show a positive SMILES mean gain, though their ratio of random to trained gain is lower than in the three pilot cohorts (Appendix B). Appendices G, H and B.2 extend the comparison to graph-generated conformers, template text, wider encoders and an externally pretrained ChemBERTa-2 target.

3.2 Random SMILES gains persist after separate validation tuning

We test whether a separately tuned graph-only model closes the gap on fresh QM9 and Alchemy molecules, giving every arm the same number of candidate training settings. Each selects among twelve training settings using validation data from a dedicated tuning block. We then fix these choices and evaluate on three blocks with no molecules in common, each with three new model seeds. The primary evaluation stays the same: ridge probes on raw graph features after 1,200 training updates from the pretrained checkpoint. Appendix C specifies the search, the molecule exclusions and the secondary protocols.

The random SMILES target still helps. With both SMILES targets frozen during graph training, the random target improves the primary two-task mean in every block of both datasets (Figure 3a), by about five percent on QM9 and three percent on Alchemy; each task also improves on its own. Its mean gains are close to, and slightly above, those of the trained SMILES target, although the third Alchemy block favors the trained target (Table 7). These three block averages per dataset describe what we observed; they do not establish equivalence or superiority. On Alchemy a random graph target gives the largest primary mean gain, suggesting that gains of this kind need not require a second view (Section 3.4).

The gains still depend on the evaluation choices. Selecting training duration on validation data reduces the random SMILES advantage, especially on Alchemy, and normalizing embeddings before fitting the probe often favors target-free controls (Figure 3b–d). Some searches choose an auxiliary-loss weight of zero, turning off alignment. This happens for every QM9 target arm when both normalization and duration selection are used; those runs reuse graph-only training and provide no evidence of an active teacher effect.

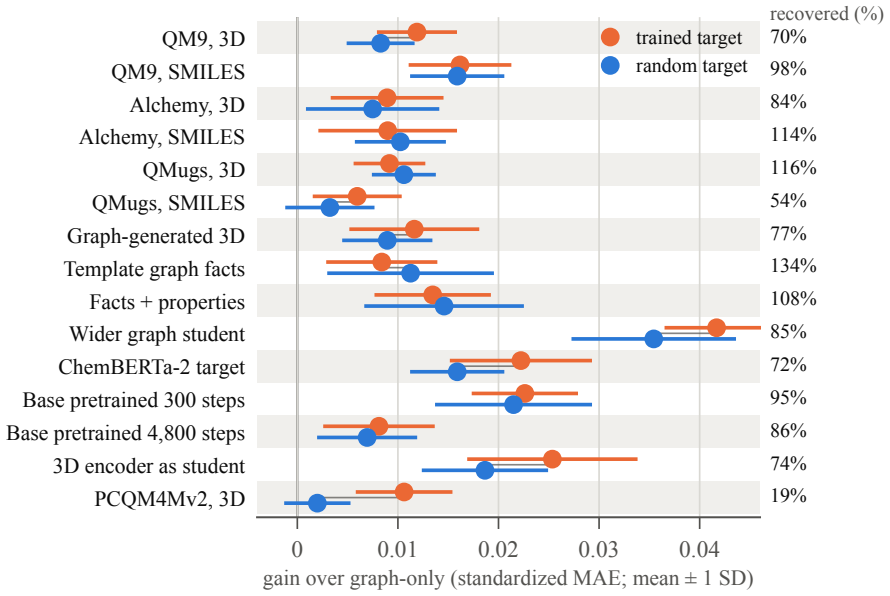


Figure 2: **Mean raw-probe gains vary across targets and molecular tasks.** Trained (orange) and random (blue) gains are means with one standard deviation across eight cohorts per row. On QM9, the trained and random SMILES targets reduce mean error by 3.6% and 3.5% relative to graph-only training. Recovered percentages are ratios of mean gains. Nominal Fieller intervals and half-gain tests appear as sensitivity analyses in Appendix A.1 and Table 2; because the training pools overlap, their independence assumptions are unverified and they do not establish calibrated confidence or equivalence. The QM9 SMILES half-gain rule is retrospective. ChemBERTa-2 (Ahmad et al., 2022) differs in architecture and pretraining corpus.

The result depends on both the task and how the representation is evaluated. Normalizing the graph features reverses the random 3D target's mean gain on QM9, Alchemy and QMugs; the PCQM4Mv2 tasks show little random-3D recovery; and on QMugs the random-SMILES gain shrinks with longer training (Appendices E, H.3 and B.1). The positive QMugs 3D result shows that these differences cannot be attributed to a general failure of 3D targets or of larger molecules.

3.3 Learned targets retain a mean advantage under GraphMVP

We also test whether the comparison changes when downstream training can update the graph encoder. We test this within GraphMVP's contrastive and generative pretraining objective (Liu et al., 2022), replacing its learned targets with random ones before supervised fine-tuning. We evaluate on ClinTox, BACE and SIDER (Wu et al., 2018a), using five partitions by molecular scaffold per dataset to assess robustness to the split. For each auxiliary arm, we combine each of three pretrained encoders with each of three fine-tuning seeds, average within each partition, and weight the five partitions equally. The primary metric is test macro ROC-AUC at the epoch selected by validation macro ROC-AUC; we also evaluate at epoch 100 (Appendix D).

Here learned targets retain a mean advantage. Random targets improve mean primary AUC over training from scratch on ClinTox and give a small positive BACE mean, with improvement in only two of five partitions; they reduce mean AUC on SIDER (Figure 4). Frozen trained targets have higher primary mean AUC than frozen random targets on all three datasets, and higher AUC in every BACE and SIDER partition. GraphMVP with jointly trained encoders also exceeds the random target in all three dataset means. The frozen trained target keeps its mean advantage at epoch 100 on all three datasets. These comparisons are conditional on the available pretrained models because partitions reuse molecules and encoders. They also differ in total compute: the frozen trained target requires additional teacher pretraining.

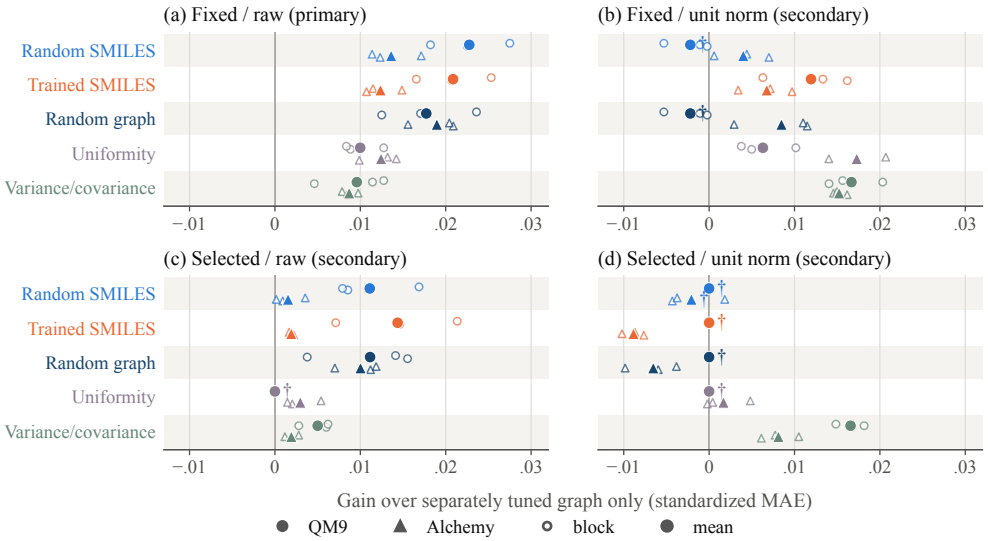


Figure 3: **Random SMILES targets retain a mean raw-probe gain after separate validation tuning.** Frozen graph representations are evaluated with ridge probes for dipole moment and HOMO energy; error averages the two task MAEs, each divided by the standard deviation of its fitting labels. Panels (a,b) evaluate after 1,200 graph-training updates from the pretrained checkpoint, and panels (c,d) select the number of updates from $\{300, 1200, 2400, 4800\}$ on validation data. Panels (a,c) use raw embeddings, while panels (b,d) first rescale each molecule's embedding to unit Euclidean norm. Gain subtracts each arm's error from the separately tuned graph-only error under the same evaluation protocol. Small markers are gains averaged over seeds within each of three fresh molecular blocks, and large markers average the three blocks. The three block values describe observed variation; they are not confidence intervals, and their seed runs are not independent dataset replications. A dagger marks a selected auxiliary-loss weight of zero, which reuses a graph-only fit; its gain can be nonzero because its selected training settings differ from those of the separately tuned graph-only baseline.

3.4 Gains do not always require a second view

We next ask whether these gains require a second molecular view. Under the primary protocol, a frozen random encoder reading the student's own graph input recovers part of the random SMILES target's mean gain and exceeds the random 3D target's (Figure 5). Even fixed Gaussian codes, assigned once to each training molecule without regard to its structure, help. Both controls also lower mean error on Alchemy and QMugs. These results are consistent with the alignment loss helping the student distinguish individual molecules (Wu et al., 2018b). Appendix F tests the role of keeping target assignments fixed through training and of assigning partners with the same elemental composition.

The targets also change how the student represents molecules. Graph-only embeddings vary strongly in only a few directions; random and trained targets spread this variation across more directions and make normalized embeddings more evenly distributed (Figure 6a). We measure these changes by effective rank (the participation ratio of covariance eigenvalues) and uniformity, respectively. Neither measure uses downstream labels, and lower uniformity loss means more spread, not necessarily lower prediction error. Changing pairings within every minibatch and adding ordinary regularizers leave the student near the low-rank baseline. Fixed codes and penalties designed to prevent collapsed representations raise rank (Table 11). Two such target-free penalties, uniformity (Wang & Isola, 2020) and VICReg's variance and covariance terms (Bardes et al., 2022), recover part of the random target's mean gain; the validation-selected ordinary regularizer's gain remains unresolved in the sensitivity analysis (Appendices E.2 and E). Yet the uniformity penalty has the largest rank in Table 11 and a smaller raw-probe gain than the random SMILES target. Spreading the representation across more directions thus accompanies the gains but does not suffice to explain them (Appendix E.3).

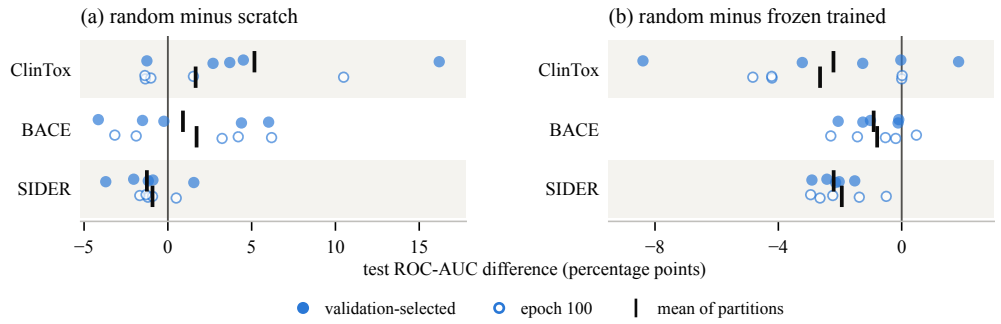


Figure 4: **Learned targets retain a mean advantage after fine-tuning.** Test macro ROC-AUC differences in percentage points, with positive values favoring frozen random targets. Small points are five scaffold-partition means after averaging crossed pretraining and fine-tuning seeds, and larger ticks are their equal-weight mean. Filled markers use the validation-selected checkpoint and hollow markers epoch 100. At the validation-selected checkpoint, the BACE random-minus-scratch difference averages $+0.90$ points and ranges from -4.15 to $+6.01$, with two of five partition means positive. Scratch runs are shared across pretraining seeds. Partitions reuse molecules and pretrained encoders, so these points describe robustness rather than independent replications or confidence intervals. Table 9 gives all arms under both evaluation protocols.

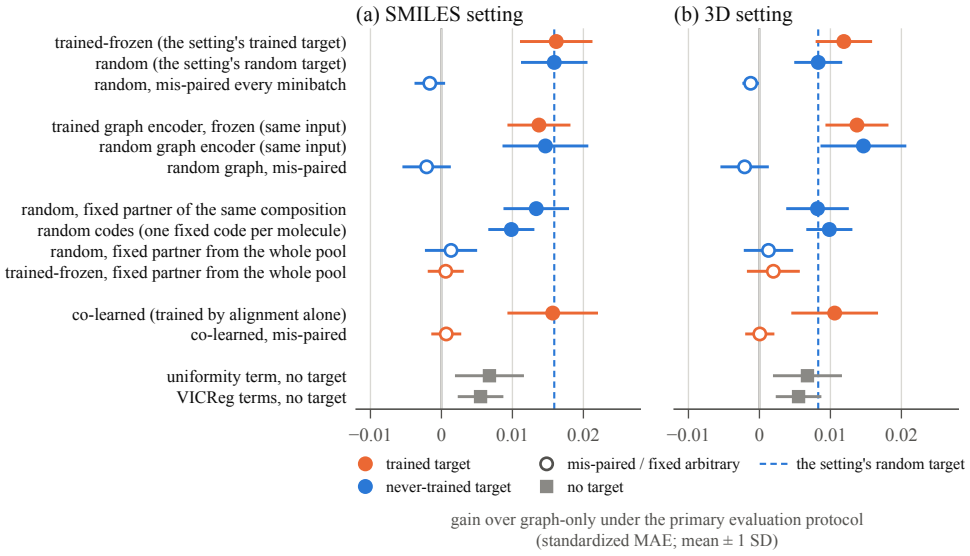


Figure 5: **Same-input targets and fixed codes give positive mean gains.** Reductions in standardized MAE relative to GRAPH-ONLY on QM9 for SMILES (left) and 3D (right). Markers and error bars are means and one standard deviation across eight cohorts. Filled markers are target-based conditions; gray squares are uniformity and the variance and covariance terms of VICReg at the same auxiliary-to-property gradient ratio as alignment. Hollow markers are controls that either resample target assignments within every minibatch or hold an arbitrary assignment fixed throughout training. The composition-matched condition restricts fixed partners to molecules with identical elemental counts and pairs molecules without an eligible partner with themselves. The dashed vertical line marks the mean gain from the correctly paired, frozen random SMILES or 3D encoder. Controls that need no second view use the same runs in both panels.

The probe comparison also depends on whether it can use embedding length (Figure 6b; Table 11). Rescaling every embedding to unit norm before fitting the probe weakens the frozen SMILES targets' gains and reverses the random 3D target's, while fixed codes and the anti-collapse penalties

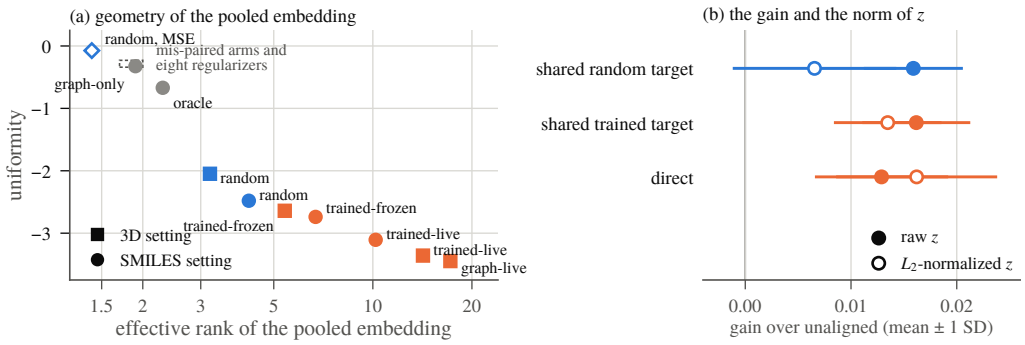


Figure 6: **Expansion and readout give different comparisons.** (a) Raw-embedding participation-ratio rank against uniformity after row normalization; lower uniformity means more spread. Squares mark the 3D setting and circles the SMILES setting; blue marks random targets and orange trained targets. The dashed region contains the mis-paired arms and eight ordinary regularizers, but not the anti-collapse controls. Geometry summaries carry no uncertainty intervals. The GRAPH-LIVE control aligns two trainable graph encoders supervised on the five auxiliary properties, and the supervised ORACLE reference fine-tunes the graph encoder on dipole and HOMO using the 4,096 fitting labels. (b) Graph-student gains, mean ± SD across eight cohorts at 1,200 updates and 4,096 fitting labels. Filled markers use raw embeddings; hollow markers normalize embeddings to unit Euclidean norm before fitting the probe. Each readout is compared with its matched graph-only baseline. Shared targets are frozen SMILES encoders, whereas DIRECT aligns graph and 3D students. Shared-target graph measurements reuse the controlled-comparison arms (Appendix H.2).

keep or improve their advantage. This removes length information and retains direction, even though alignment itself already uses normalized embeddings. A gain on raw features therefore need not mean that their directions better predict the task. Nor does improved prediction require greater student–target similarity as measured by raw linear centered kernel alignment (CKA) (Kornblith et al., 2019): CKA falls while graph-probe prediction improves in the controlled QM9 comparisons (Appendix E.5).

Retrospective diagnostics ask whether predicting the teacher's features explains the gain (Appendix E.6). At 2,400 updates, random-SMILES alignment lowers both teacher-feature prediction error and task error on molecules outside the training size range, measured by normalized MSE. Yet a separate teacher-based correction added to the graph-only predictor's residuals worsens this shifted task error in every cohort. Raw random-geometry targets are already nearly linearly predictable from graph-only features at the same training step; after normalization, their predictability worsens as task error improves. A linear correction through the teacher therefore does not reproduce the student's shifted-pool gain. These diagnostics and the geometry measurements identify no causal mechanism and do not verify the population assumptions of Appendix I.

3.5 TRAINING THE TARGET HELPS SPECIFIC PROPERTIES

Training the target helps some properties more than others (Figure 7). On the additional orbital-energy tasks, random and trained SMILES targets again have similar mean gains, but most of the 3D target's LUMO improvement comes from training. The difference is clearer for rotational constants evaluated on a logarithmic scale: the trained 3D target improves all three property means, while the random target worsens them. Appendix G gives the property-selection rule and explains why the original-scale rotational readout was inconclusive; Table 19 reports the log-scale refit.

This advantage need not come from information missing from the graph. Replacing the DFT-computed geometry with a conformer generated from the graph preserves the trained target's advantage on LUMO and the HOMO–LUMO gap (Appendix G.1), so on those properties it does not require DFT geometry. Conversely, writing the downstream property values into template text does not establish an advantage for the trained target over the random one; this is a diagnostic in-

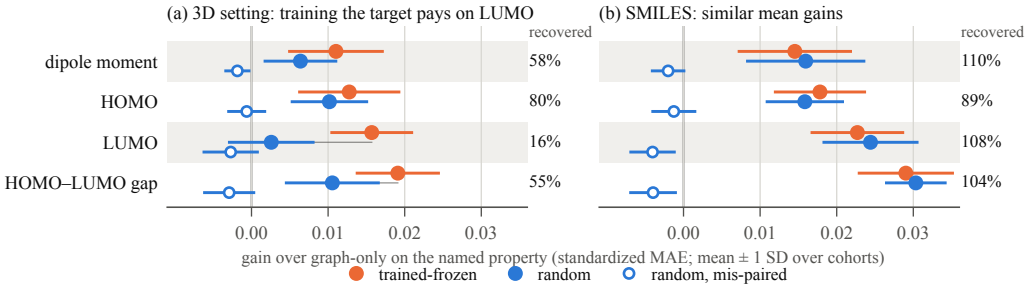


Figure 7: **The value of target training is property-specific.** QM9 gains over GRAPH-ONLY, mean ± SD across eight cohorts. Orange marks trained-frozen targets and blue random targets, with hollow markers for minibatch-wise mis-pairing. Training adds most on LUMO with 3D targets, whereas the SMILES means are close. Log-rotational probes give a further sign reversal (Table 19).

tervention, not a deployable text benchmark (Appendix G.2). Excess kernel–property alignment (Cristianini et al., 2001) also fails to resolve which trained targets add more gain (Table 22).

3.6 Duration, architecture and dataset limit recovery

Starting SMILES alignment after 600 updates gives a mean gain similar to using it for all 1,200 updates, whereas stopping at 300 retains less. At 1,200 updates, random SMILES alignment has lower mean error than 9,600 graph-only updates. The 3D comparison remains unresolved in the sensitivity analysis (Figure 8; Appendix H.1). These update-budget comparisons establish neither wall-clock speedups nor a unique early training window.

Widening both encoders retains positive mean gains, whereas adding global attention to the student reverses the random-target mean gain (Appendix H.2). A 3D student benefits from random SMILES on raw in-distribution probes, but its mean size-shifted gain is negative and normalization reduces recovery (Appendix H.2).

On PCQM4Mv2, trained targets give larger mean gains in both modalities, although the trained-SMILES gain remains unresolved in the sensitivity analysis (Appendix H.3). This comparison changes molecules, auxiliary labels and downstream tasks together, preventing attribution to any one factor. The baseline has lower participation-ratio rank than QM9 despite weaker random-target gains, so low rank alone does not predict the benefit.

4 Discussion and conclusion

In this work, we show that random SMILES targets yield similar mean primary-probe gains to trained targets on QM9 and Alchemy, while trained targets help more elsewhere. A frozen-random control under the intended protocol helps distinguish these settings.

Limitations. Our comparisons isolate the contribution of auxiliary-encoder training within fixed objectives and budgets. Assessing competitiveness with state-of-the-art multimodal methods requires broader evaluation. More extensive optimization could strengthen the trained targets, so recovery remains relative to the targets studied here. Initial gradient matching controls the starting scale but leaves later dynamics unconstrained. Equal candidate counts likewise leave differences in search spaces and compute. Low initial QM9 rank and molecule size provide useful context that warrant further investigation.

Our decomposition and diagnostics help characterize the observed gains. The next step is to identify a causal mechanism. Future work can strengthen these comparisons through independent replications, more extensive target optimization, and evaluations of data augmentation, independently authored text and 3D Infomax.

Reproducibility statement

Section 2 defines the training intervention and evaluation protocol. Appendix A specifies data processing, partitions, model architectures, optimization, calibration and probe fitting. Appendices A.1 and A.2 state the statistical assumptions, prespecified choices, retrospective analyses and exclusions. Appendix B.1 describes the dataset extensions, and Appendices C and D detail validation search, checkpoint selection and the fresh-molecule and GraphMVP comparisons. Appendices E, F, G and H specify the additional controls and diagnostics. Appendix I states the theoretical assumptions and gives the complete proof. Appendix J lists the software, hardware and dataset sources. Reported tables derive from saved metrics or pipeline diagnostics. The supplementary code provides model and analysis implementations, scientific settings, dependency specifications and run instructions. Public data and external weights must be obtained separately, and training outputs must be regenerated.

AI use statement

In this work, we used generative AI tools to develop theoretical models and conceptual frameworks, formulate mathematical claims, contribute key ideas and arguments for their proofs, assist in writing proofs, propose and refine hypotheses, design and critique research methodology and experiments, implement methods, clean and reformat datasets, and interpret results. We used AI-assisted code to generate synthetic control data, consisting of fixed random codes and template descriptions. Molecular datasets and property labels came from public sources. We did not use generative AI for translation, and qualitative and thematic data analysis is not applicable to this work. Additionally, we used generative AI to identify relevant literature, summarize existing work, create and modify scientific figures, and draft and edit the manuscript. We checked AI-assisted claims against cited sources, reviewed proof assumptions and derivations, tested experiment and analysis code, and compared reported values with saved experimental outputs. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

Ethics statement

The experiments use public molecular datasets and involve no human subjects or personal data.

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 Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations (ICLR), Workshop Track*, 2017. arXiv:1610.01644.

Simon Axelrod and Rafael Gómez-Bombarelli. GEOM, energy-annotated molecular conformations for property prediction and molecular generation. *Scientific Data*, 9:185, 2022. doi: 10.1038/s41597-022-01288-4.

Yusuf Aytar, Carl Vondrick, and Antonio Torralba. SoundNet: Learning sound representations from unlabeled video. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. URL https://arxiv.org/abs/1607.06450.

Christoph Bannwarth, Sebastian Ehlert, and Stefan Grimme. GFN2-xTB—an accurate and broadly parametrized self-consistent tight-binding quantum chemical method with multipole electrostatics and density-dependent dispersion contributions. *Journal of Chemical Theory and Computation*, 15(3):1652–1671, 2019. doi: 10.1021/acs.jctc.8b01176.

Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2022.

Yoshua Bengio and Yves Grandvalet. No unbiased estimator of the variance of K-fold cross-validation. *Journal of Machine Learning Research*, 5:1089–1105, 2004. URL https://www.jmlr.org/papers/v5/grandvalet04a.html.

Piotr Bojanowski and Armand Joulin. Unsupervised learning by predicting noise. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 517–526. PMLR, 2017.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *International Conference on Learning Representations (ICLR)*, 2019.

A. Colin Cameron, Jonah B. Gelbach, and Douglas L. Miller. Bootstrap-based improvements for inference with clustered errors. *The Review of Economics and Statistics*, 90(3):414–427, 2008. doi: 10.1162/rest.90.3.414.

Jeng-Da Chai and Martin Head-Gordon. Long-range corrected hybrid density functionals with damped atom–atom dispersion corrections. *Physical Chemistry Chemical Physics*, 10:6615–6620, 2008. doi: 10.1039/B810189B.

Guangyong Chen, Pengfei Chen, Chang-Yu Hsieh, Chee-Kong Lee, Benben Liao, Renjie Liao, Weiwen Liu, Jiezhong Qiu, Qiming Sun, Jie Tang, Richard Zemel, and Shengyu Zhang. Alchemy: A quantum chemistry dataset for benchmarking AI models. *arXiv preprint arXiv:1906.09427*, 2019.

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

Chung-Cheng Chiu, James Qin, Yu Zhang, Jiahui Yu, and Yonghui Wu. Self-supervised learning with random-projection quantizer for speech recognition. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 3915–3924. PMLR, 2022.

Youngmin Cho and Lawrence K. Saul. Kernel methods for deep learning. In *Advances in Neural Information Processing Systems*, volume 22, 2009. URL https://papers.nips.cc/paper/2009/hash/5751ec3e9a4feab575962e78e006250d-Abstract.html.

Nello Cristianini, John Shawe-Taylor, André Elisseeff, and Jaz S. Kandola. On kernel-target alignment. In *Advances in Neural Information Processing Systems*, volume 14, 2001.

Karan Desai and Justin Johnson. VirTex: Learning visual representations from textual annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11162–11173, 2021.

E. C. Fieller. Some problems in interval estimation. *Journal of the Royal Statistical Society: Series B (Methodological)*, 16(2):175–185, 1954.

Michael J. Frisch, John A. Pople, and J. Stephen Binkley. Self-consistent molecular orbital methods 25. supplementary functions for Gaussian basis sets. *The Journal of Chemical Physics*, 80(7): 3265–3269, 1984. doi: 10.1063/1.447079.

Peiran Gao, Eric M. Trautmann, Byron M. Yu, Gopal Santhanam, Stephen Ryu, Krishna Shenoy, and Surya Ganguli. A theory of multineuronal dimensionality, dynamics and measurement. *bioRxiv*, 2017. doi: 10.1101/214262.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent – a new approach to self-supervised learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Fabian Gröger, Shuo Wen, and Maria Brbić. Revisiting the Platonic representation hypothesis: An Aristotelian view. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2602.14486.

Sharut Gupta, Shobhita Sundaram, Chenyu Wang, Stefanie Jegelka, and Phillip Isola. Better together: Leveraging unpaired multimodal data for stronger unimodal models. In *International Conference on Learning Representations*, 2026. URL https://unpaired-multimodal.github.io/.

Thomas A. Halgren. Merck molecular force field. I. basis, form, scope, parameterization, and performance of MMFF94. *Journal of Computational Chemistry*, 17(5–6):490–519, 1996.

Jeff Z. HaoChen, Colin Wei, Adrien Gaidon, and Tengyu Ma. Provable guarantees for self-supervised deep learning with spectral contrastive loss. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. URL https://arxiv.org/abs/1512.03385.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*, 2016. URL https://arxiv.org/abs/1606.08415.

Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970. doi: 10.1080/00401706.1970.10488634.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. Strategies for pre-training graph neural networks. In *International Conference on Learning Representations (ICLR)*, 2020.

Weihua Hu, Matthias Fey, Hongyu Ren, Maho Nakata, Yuxiao Dong, and Jure Leskovec. OGB-LSC: A large-scale challenge for machine learning on graphs. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/hash/db8e1af0cb3aca1ae2d0018624204529-Abstract-round2.html.

Po-Yao Huang, Vasu Sharma, Hu Xu, Chaitanya Ryali, Haoqi Fan, Yanghao Li, Shang-Wen Li, Gargi Ghosh, Jitendra Malik, and Christoph Feichtenhofer. MAViL: Masked audio-video learners. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

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

Clemens Isert, Kenneth Atz, José Jiménez-Luna, and Gisbert Schneider. QMugs, quantum mechanical properties of drug-like molecules. *Scientific Data*, 9:273, 2022.

Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2022.

Smayan Khanna, Doruk Efe Gökmen, Risi Kondor, and Vincenzo Vitelli. Graph contrastive learning versus untrained baselines: The role of dataset size. *arXiv preprint arXiv:2509.01541*, 2025.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. URL https://arxiv.org/abs/1412.6980.

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

Gregory A. Landrum et al. RDKit: Open-source cheminformatics. https://www.rdkit.org, n.d.

Zhenzhu Li and Aron Walsh. Platonic representation of foundation machine learning interatomic potentials. *Nature Machine Intelligence*, 8:830–840, 2026. doi: 10.1038/s42256-026-01235-7. URL https://www.nature.com/articles/s42256-026-01235-7.

Shengchao Liu, Hanchen Wang, Weiyang Liu, Joan Lasenby, Hongyu Guo, and Jian Tang. Pre-training molecular graph representation with 3D geometry. In *International Conference on Learning Representations (ICLR)*, 2022.

Shengchao Liu, Weili Nie, Chengpeng Wang, Jiarui Lu, Zhuoran Qiao, Ling Liu, Jian Tang, Chaowei Xiao, and Animashree Anandkumar. Multi-modal molecule structure–text model for text-based retrieval and editing. *Nature Machine Intelligence*, 5(12):1447–1457, 2023a. doi: 10.1038/s42256-023-00759-6.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pre-training approach. *arXiv preprint arXiv:1907.11692*, 2019. URL https://arxiv.org/abs/1907.11692.

Zhiyuan Liu, Sihang Li, Yanchen Luo, Hao Fei, Yixin Cao, Kenji Kawaguchi, Xiang Wang, and Tat-Seng Chua. MolCA: Molecular graph-language modeling with cross-modal projector and uni-modal adapter. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 15623–15638. Association for Computational Linguistics, 2023b. doi: 10.18653/v1/2023.emnlp-main.966.

Zhiyuan Liu, Yaorui Shi, An Zhang, Enzhi Zhang, Kenji Kawaguchi, Xiang Wang, and Tat-Seng Chua. Rethinking tokenizer and decoder in masked graph modeling for molecules. In *Advances in Neural Information Processing Systems*, volume 36, 2023c.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted Boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning*, 2010. URL https://icml.cc/2010/papers/432.pdf.

Maho Nakata and Tomomi Shimazaki. PubChemQC project: A large-scale first-principles electronic structure database for data-driven chemistry. *Journal of Chemical Information and Modeling*, 57 (6):1300–1308, 2017.

Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. Multimodal deep learning. In *Proceedings of the 28th International Conference on Machine Learning*, pp. 689–696, 2011. URL https://icml.cc/Conferences/2011/papers/399_icmlpaper.pdf.

Karl Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901. doi: 10.1080/14786440109462720.

Mateusz Praski, Jakub Adamczyk, and Wojciech Czech. Benchmarking pretrained molecular embedding models for molecular representation learning. *arXiv preprint arXiv:2508.06199*, 2025.

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

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, volume 20, 2007.

Ali Rahimi and Benjamin Recht. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In *Advances in Neural Information Processing Systems*, volume 21, 2008. URL https://papers.nips.cc/paper/2008/hash/0efe32849d230d7f53049ddc4a4b0c60-Abstract.html.

Raghunathan Ramakrishnan, Pavlo O. Dral, Matthias Rupp, and O. Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1:140022, 2014. doi: 10.1038/sdata.2014.22.

Ladislav Rampášek, Mikhail Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a general, powerful, scalable graph transformer. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Sereina Riniker and Gregory A. Landrum. Better informed distance geometry: Using what we know to improve conformation generation. *Journal of Chemical Information and Modeling*, 55 (12):2562–2574, 2015.

Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *15th European Signal Processing Conference (EUSIPCO)*, Poznań, Poland, 2007.

Lars Ruddigkeit, Ruud van Deursen, Lorenz C. Blum, and Jean-Louis Reymond. Enumeration of 166 billion organic small molecules in the chemical universe database GDB-17. *Journal of Chemical Information and Modeling*, 52(11):2864–2875, 2012. doi: 10.1021/ci300415d.

Felix Sarnthein, Gregor Bachmann, Sotiris Anagnostidis, and Thomas Hofmann. Random teachers are good teachers. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 30022–30041. PMLR, 2023.

Nikunj Saunshi, Jordan Ash, Surbhi Goel, Dipendra Misra, Cyril Zhang, Sanjeev Arora, Sham Kakade, and Akshay Krishnamurthy. Understanding contrastive learning requires incorporating inductive biases. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 19250–19286. PMLR, 2022.

Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. SchNet: A continuous-filter convolutional neural network for modeling quantum interactions. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

C. Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101, 1904. doi: 10.2307/1412159.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014. URL https://jmlr.org/papers/v15/srivastava14a.html.

Hannes Stärk, Dominique Beaini, Gabriele Corso, Prudencio Tossou, Christian Dallago, Stephan Günnemann, and Pietro Lió. 3D Infomax improves GNNs for molecular property prediction. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 20479–20502. PMLR, 2022.

P. J. Stephens, F. J. Devlin, C. F. Chabalowski, and M. J. Frisch. Ab initio calculation of vibrational absorption and circular dichroism spectra using density functional force fields. *The Journal of Physical Chemistry*, 98(45):11623–11627, 1994. doi: 10.1021/j100096a001.

Student. The probable error of a mean. *Biometrika*, 6(1):1–25, 1908. doi: 10.1093/biomet/6.1.1.

Ruoxi Sun, Hanjun Dai, and Adams Wei Yu. Does GNN pretraining help molecular representation? In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Megan Tjandrasuwita, Chanakya Ekbote, Liu Ziyin, and Paul Pu Liang. Understanding the emergence of multimodal representation alignment. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 59723–59760. PMLR, 2025. URL https://proceedings.mlr.press/v267/tjandrasuwita25a.html.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

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

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9929–9939. PMLR, 2020.

Matthew D. Webb. Reworking wild bootstrap-based inference for clustered errors. *Canadian Journal of Economics/Revue canadienne d'économique*, 56(3):839–858, 2023. doi: 10.1111/caje.12661.

Florian Weigend and Reinhart Ahlrichs. Balanced basis sets of split valence, triple zeta valence and quadruple zeta valence quality for H to Rn: Design and assessment of accuracy. *Physical Chemistry Chemical Physics*, 7:3297–3305, 2005. doi: 10.1039/B508541A.

David Weininger. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, 1988. doi: 10.1021/ci00057a005.

Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay Pande. MoleculeNet: a benchmark for molecular machine learning. *Chemical Science*, 9(2):513–530, 2018a. doi: 10.1039/C7SC02664A.

Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3733–3742, 2018b.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 10524–10533. PMLR, 2020. URL https://proceedings.mlr.press/v119/xiong20b.html.

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

Jinhua Zhu, Yingce Xia, Lijun Wu, Shufang Xie, Wengang Zhou, Tao Qin, Houqiang Li, and Tie-Yan Liu. Dual-view molecular pre-training. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023. doi: 10.1145/3580305.3599317.

A Training, evaluation and statistical comparisons

Data and cohorts. We use 19,456 QM9 molecules (Ramakrishnan et al., 2014; Ruddigkeit et al., 2012), deduplicated by canonical SMILES. Of these, 17,408 have at most eight heavy atoms and 2,048 have nine. Only H, C, N, O and F occur, and we do not count hydrogen as a heavy atom. We repeat the comparison in eight cohorts, each defined by a data partition and a model seed. The in-distribution pools contain 9,216 pretraining molecules, 4,096 fitting molecules, 1,024 validation molecules, 2,048 test molecules and 1,024 unused molecules. We split pretraining equally between the graph and target encoders, then use the whole pretraining pool for subsequent graph training.

Table 1: **What each intervention changes.** All controlled arms start from the same graph checkpoint and use the same minibatch stream. Frozen targets receive no updates during graph training. Appendix E specifies additional controls and calibration checks.

arm	target or additional constraint
GRAPH-ONLY	none (the property objective alone)
TRAINED-FROZEN	auxiliary encoder trained on the five properties, then frozen
RANDOM	the same auxiliary architecture, initialized randomly and frozen
TRAINED-LIVE/ CO-LEARNED	target learns from property labels / learns through alignment alone
RANDOM GRAPH	frozen random encoder of the student's own graph input
RANDOM CODES	one fixed Gaussian vector per training molecule
GRAPH-LIVE	two trainable graph encoders with property and alignment losses
ORACLE	graph encoder fine-tuned on dipole and HOMO fitting labels
UNIFORMITY / VICREG	target-free anti-collapse penalties
MIS-PAIRED	target assignment changed as detailed in Section F

Test pools are disjoint across cohorts, while training pools can overlap. The same 2,048 nine-heavy-atom molecules form every cohort's size-shifted test pool. Model seeds are 101, 202, 303, 404, 505, 606, 707 and 808. Table 3 separates the three motivating pilot cohorts from the five later cohorts.

Inputs. The graph encoder reads atomic numbers, bond types (single, double, triple, aromatic) and an atom mask. It does not use molecular distances: replacing the cached distances leaves its checkpoints identical. The 3D input is the interatomic distance matrix of the QM9 equilibrium geometry. We generate canonical, non-isomeric SMILES from the graph with RDKit (Landrum et al., n.d.) and encode one token per byte with a padding token (vocabulary 257). This input contains no external annotation.

Architectures. All encoders have width 128 and depth three unless we specify otherwise. The graph encoder embeds atom and bond types and uses three GINE-style blocks. Each block sums neighboring states after $\mathrm{ReLU}(a_j + e_{ij})$ (Nair & Hinton, 2010). A two-layer MLP then updates $(1+\epsilon)a_i$ plus the aggregate, and a residual connection (He et al., 2016) and layer normalization (Ba et al., 2016) complete the block. The learnable ϵ starts at zero. The 3D encoder uses 32 Gaussian radial basis functions on $[0, 5]$ Å, a cosine cutoff at 5 Å and three SchNet-style continuous-filter blocks (Schütt et al., 2017), with two-layer filter MLPs. The SMILES encoder is a Transformer (Vaswani et al., 2017) with pre-layer normalization (Xiong et al., 2020), four heads, feed-forward width 256, GELU (Hendrycks & Gimpel, 2016), sinusoidal positions, a final layer normalization and no dropout (Srivastava et al., 2014). We initialize its layers independently. Each encoder concatenates sum and mean pooling and projects to 128 dimensions. We initialize each random target as a fresh instance of its architecture with the cohort seed plus 6,001 and never update it.

Training. We first train the graph encoder and learned target for 1,200 AdamW updates (Loshchilov & Hutter, 2019), with learning rate 10^{-3}, weight decay 10^{-4} and batch size 256. A linear head predicts five properties by mean squared error: polarizability, electronic spatial extent, zero-point vibrational energy, internal energy at 0 K and heat capacity. We standardize labels on the encoder's pretraining half and use its final checkpoint. On supported CUDA hardware, forward passes use BF16 automatic mixed precision; otherwise they use FP32. Thus, recovery is measured against a teacher with this architecture, objective and training budget; we do not establish convergence or optimal teacher quality. Further graph training uses AdamW with learning rate $3 \cdot 10^{-4}$ (primary) or 10^{-3}, the same weight decay unless we vary it, and identical minibatch draws across arms. Unless we specify pretraining, update counts start from the pretrained checkpoint. Checkpoints are at 300, 600, 1,200 and 2,400 updates. In addition, GRAPH-ONLY, RANDOM, TRAINED-FROZEN and TRAINED-LIVE reach 4,800 updates in the 3D setting. We update a live target on the same minibatches using its own property loss and the alignment loss. We exclude frozen targets from the optimizer and compute their representations without gradients.

Alignment calibration. We set the initial strength of alignment relative to the property objective. The InfoNCE temperature is 0.1. Before further graph training, each arm's alignment coefficient makes its alignment-gradient norm 0.2 times its own-objective gradient norm. We use the median coefficient from four calibration minibatches, measuring gradients on the parameters alignment can update and using that arm's target. At the primary learning rate, mean coefficients are 2.0 and 1.4

for random and trained frozen 3D targets, and 2.7 and 0.95 for SMILES targets. The squared-error arm requires about 40. These values vary across cohorts. This rule matches the initial ratio and gives different numerical weights across arms. It does not keep gradients matched throughout training, so the comparison depends on the calibration convention. Appendix C varies the ratio through separate validation search.

Evaluation. We freeze encoders for evaluation and fit a ridge head (Hoerl & Kennard, 1970) per property on the first 64, 256, 1,024 or 4,096 fitting molecules. We center and scale each feature coordinate using the training portion of each fold. Three-fold cross-validation minimizes standardized squared error to select the penalty from $\{0, 10^{-8}, 10^{-6}, 10^{-4}, 10^{-3}, 10^{-2}, 0.1, 1, 10, 100\}$. We then refit the scaler and head on all fitting molecules and evaluate the same head on both test pools. Secondary validation-selected comparisons minimize mean validation standardized MAE over the intersection of the two arms' checkpoint grids and the two learning rates. Appendix E.4 compares raw, row-normalized and norm-only probes without selecting among them. Linear evaluation measures frozen representations (Alain & Bengio, 2017; Chen et al., 2020; Praski et al., 2025), whereas the GraphMVP experiment fine-tunes the graph encoder. Downstream dipole and HOMO labels do not enter encoder training, apart from the supervised ORACLE reference and the explicitly identified property-text diagnostic (Appendix G.2). The study compares arms under this readout rather than claiming competitive absolute QM9 accuracy.

Standardized MAE is dimensionless. Physical errors use debye for dipole, electronvolts for orbital energy and gigahertz for rotational constants. We convert QM9 energies from hartree.

A.1 Statistical sensitivity analyses

Our primary summaries are the paired gains and their variation across the eight cohorts. Because training pools overlap and the size-shifted test pool is shared, we cannot treat these comparisons as established independent replications of a learning procedure. Shared data can make naive variance estimates unreliable (Bengio & Grandvalet, 2004). For sensitivity analysis, we also compute Student's t intervals and tests with seven degrees of freedom (Student, 1908) and enumerate all 2^8 Rademacher sign flips of null-restricted residuals (Cameron et al., 2008; Webb, 2023). These calculations do not account for dependence caused by shared data. Their nominal confidence levels and p-values therefore rely on assumptions we have not verified. They describe sensitivity under those assumptions and do not establish recovery in the broader population.

The recovered fraction divides the mean random-target gain by the mean trained-target gain. Nominal Fieller intervals (Fieller, 1954) account for covariance between these paired gains, but assume independence across cohorts. A large fraction can describe recovery of a modest error reduction, and a ratio near 100% does not establish equivalence. The controlled study's primary contrast compares downstream graph-probe errors with a frozen random 3D teacher and a frozen pretrained 3D teacher of the same architecture on QM9, after 1,200 graph-training updates at learning rate $3 \cdot 10^{-4}$ and with 4,096 probe-fitting labels. Our prespecified criterion asks whether the random target retains at least half the trained target's gain over graph-only training. It compares the nominal one-sided 95% lower bound on $e_{\mathrm{TF}} - e_{\mathrm{RND}}$ with $-\hat{\delta}$, where $\hat{\delta} = \frac{1}{2}\overline{(e_{\mathrm{GO}} - e_{\mathrm{TF}})}$. A nonpositive trained gain makes the rule undecidable. Here e_{GO}, e_{TF} and e_{RND} denote per-cohort standardized MAEs. This plug-in rule treats the estimated margin as fixed. To include uncertainty in that margin, we also test whether the mean of the cohort contrasts $\theta = \frac{1}{2}(e_{\mathrm{GO}} + e_{\mathrm{TF}}) - e_{\mathrm{RND}}$ is positive. Neither calculation corrects dependence across cohorts. Both concern half-gain recovery rather than equality with the trained target.

We prespecified the half-gain criterion for QM9 3D and applied it retrospectively to controlled SMILES. The separately tuned SMILES comparison fixes configurations before testing on fresh molecular blocks (Appendix C).

A single prespecified contrast needs no multiplicity correction. For setting-specific half-gain comparisons across the study, Table 2 additionally applies Holm correction (Holm, 1979) to both calculations over all seventeen settings. This family includes two masked-atom student comparisons omitted from Figure 2 because their baseline-admission criterion remains unresolved. Holm permits dependent hypotheses when the individual p-values are valid. It cannot repair their unverified calibration here or the plug-in margin's uncertainty, and it does not test an aggregate effect across settings. The QM9 3D adjusted values exceed 0.05 for both calculations. Its unadjusted plug-in

classification also changes under five of eight leave-one-cohort-out deletions, whereas the QM9 SMILES classification does not change. These checks describe sensitivity rather than establish either recovery or its absence. We summarize the fresh blocks in Appendix C and scaffold partitions in Appendix D separately from these cohort calculations.

A.2 Statistical scope and analysis limitations

We use prespecification to mean choices recorded in study plans; these plans were not externally timestamped preregistrations. The motivating SMILES finding preceded the QM9 3D and dataset-extension plans.

The property-admissibility screen uses squared Spearman correlation (Spearman, 1904) with a 0.95 threshold. We selected its statistic and cohort agreement criterion during analysis, although those choices did not change which properties qualified. We chose logarithmic rotational readouts after observing unstable identity-scale errors (Appendix G). The 3D-student and teacher-fitting diagnostics are secondary evidence, not additional prespecified primary replications. Their conclusions retain the raw-versus-normalized and shared-test-pool qualifications (Appendices H.2 and E.6).

Several results run against our proposed explanations. A graph-generated conformer preserves a trained target's advantage, while inserting downstream values into template text does not ensure one (Appendix G). Same-input targets and fixed codes succeed despite predictions that they would recover less of the gain. Gains from target-free penalties also show that the benefit need not require two modalities (Appendix E). The limited recovery on PCQM4Mv2 does not coincide with a higher-rank baseline (Appendix H.3).

B Controlled recovery across molecular views and datasets

We compare how much of the trained target's gain a random target retains across views and datasets (Tables 4 and 2). Each setting follows Appendix A and averages eight cohorts, with the dependence and statistical limitations in Appendix A.1.

Interpreting recovery and sensitivity summaries. The estimated recovery fraction and the half-gain tests answer different questions. For QM9 SMILES, Alchemy SMILES and QMugs 3D, both adjusted half-gain p-values fall below 0.05 under the assumptions in Appendix A.1. The corresponding QM9 3D values exceed 0.05 despite its estimated 70% recovery. These tests omit cohort dependence, so their thresholds cannot reliably confirm or reject recovery. Even with independent replications, passing a half-gain test would establish neither equivalence nor full recovery.

Pilot sensitivity. We check how much the motivating pilot contributes to the SMILES result by separating its three cohorts from the five later cohorts (Table 3). The ratio of mean gains is 84% in the later cohorts and 124% in the pilot. Across all eight, the 98% ratio corresponds to random and trained reductions of approximately 3.5% and 3.6% in graph-only error. The retrospective SMILES half-gain calculation remains below its nominal threshold on the later five alone, whereas the 3D calculation exceeds it within either subset. These subsets retain the shared-data limitations of the full study.

B.1 Alchemy and QMugs

To test whether the findings extend beyond QM9, we repeat the controlled comparison on Alchemy and QMugs. Both retain the cohort pool sizes, architectures and primary evaluation protocol, with raw probes after 1,200 updates. Within each dataset, the two views share the same graph-only results.

Alchemy (Chen et al., 2019) supplies the same property set from B3LYP/6-31G(2df,p) calculations (Stephens et al., 1994; Frisch et al., 1984). We sample the DGL-LifeSci contest-development mirror after restricting elements to H, C, N, O and F, taking 17,408 molecules with 9–10 heavy atoms and 2,048 with 11. We retain the mirror's standardized labels and check column identities against energy relations and size correlations. Auxiliary properties and downstream dipole/HOMO match QM9. Both random targets give positive mean gains, while the 3D recovery interval is wide.

Table 2: Recovery estimates underlying Figure 2. Ratios divide the observed mean random-target gain by the trained-target gain. The nominal Fieller intervals and Holm-adjusted p-values are assumption-dependent sensitivity summaries, with no established coverage or calibration under the overlapping-cohort design. The two p columns concern the plug-in half-gain rule and the estimated-margin calculation. Both retain all seventeen settings, including the two omitted masked-atom student comparisons. Values below 0.05 concern half-gain recovery under those assumptions and do not establish equivalence or full recovery.

Comparison	Recovered (%)	Nominal 95% Fieller interval	Nominal p_{Holm}	Nominal $p_{\theta,\mathrm{Holm}}$
3D setting, DFT view	70	[50, 93]	0.25	0.150
SMILES setting (retrospective rule)	98	[79, 124]	0.008	0.004
3D setting, Alchemy	84	[30, 145]	0.31	0.23
SMILES setting, Alchemy	114	[81, 234]	0.045	0.005
3D setting, QMugs	116	[100, 140]	<0.001	<0.001
SMILES setting, QMugs	54	[−18, 85]	0.75	0.80
3D setting, RDKit view	77	[59, 106]	0.22	0.054
text view, graph facts	134	[54, 298]	0.22	0.147
text view, facts + properties	108	[64, 167]	0.106	0.075
SMILES, width 512	85	[74, 95]	<0.001	0.002
SMILES, ChemBERTa target	72	[55, 94]	0.22	0.075
SMILES, base 300 updates	95	[75, 114]	0.007	0.013
SMILES, base 4,800 updates	86	[42, 165]	0.31	0.23
3D encoder as student	74	[62, 87]	0.053	0.013
PCQM4Mv2, 3D, DFT view	19	[−10, 37]	0.99	1.00

Table 3: Sensitivity to inclusion of the motivating pilot. Entries are mean gains in standardized MAE over the matched graph-only model. Ratios use unrounded gains. Additional cohorts share the molecular study draw and do not constitute an independent external replication.

SMILES cohorts	Number	Trained gain	Random gain	Recovered (%)
Pilot	3	+0.0151	+0.0188	124
Additional	5	+0.0168	+0.0142	84
All	8	+0.0162	+0.0159	98

QMugs (Isert et al., 2022) combines GFN2-xTB conformers (Bannwarth et al., 2019) with ωB97X-D/def2-SVP single-point properties (Chai & Head-Gordon, 2008; Weigend & Ahlrichs, 2005). We use the first conformer, remove imaginary frequencies and duplicate SMILES, restrict to the same elements, and sample 17,408 molecules with 20–30 heavy atoms and 2,048 with 31. We verify structure–label matching by identifier, atom count and formula. Auxiliary labels are GFN2 polarizability, entropy, C6 coefficient and heat capacity, plus DFT formation energy. Downstream labels remain DFT dipole and HOMO. QMugs therefore changes chemical space, auxiliary properties and geometry source together. The random 3D target has positive mean gain. The random-SMILES gain is smaller than early in training, and its nominal interval includes zero under the primary evaluation protocol. Same-input and pairing controls across these datasets appear in Appendix E.

Table 4: Controlled dataset comparisons at 1,200 updates and 4,096 fitting labels. Gains are mean standardized-MAE reductions over eight cohorts. Parentheses count positive cohort differences, and intervals are nominal 95% paired intervals. Table 2 gives recovered fractions and assumption-dependent test summaries; Appendix A.1 explains the shared-pool dependence.

View	Dataset	Baseline	Trained gain	Random gain	Random − trained (95% CI)
3D	QM9	0.4482	+0.0119 (8)	+0.0083 (8)	−0.0036 [−0.0065, −0.0007] (1)
3D	Alchemy	0.4488	+0.0089 (8)	+0.0075 (7)	−0.0014 [−0.0060, +0.0031] (2)
3D	QMugs	0.6184	+0.0092 (8)	+0.0106 (8)	+0.0014 [+0.0000, +0.0029] (6)
SMILES	QM9	0.4482	+0.0162 (8)	+0.0159 (8)	−0.0003 [−0.0038, +0.0032] (4)
SMILES	Alchemy	0.4488	+0.0090 (7)	+0.0102 (8)	+0.0013 [−0.0024, +0.0049] (5)
SMILES	QMugs	0.6184	+0.0060 (8)	+0.0032 (6)	−0.0027 [−0.0046, −0.0008] (1)

B.2 An externally pretrained SMILES target

We also compare the random SMILES target with an encoder pretrained on a much larger external corpus. The released `ChemBERTa-77M-MTR` checkpoint from ChemBERTa-2 (Ahmad et al., 2022) uses three RoBERTa layers (Liu et al., 2019), width 384 and approximately 3.3M encoder parameters, pretrained by regression of roughly 200 RDKit properties on 77M PubChem SMILES. We mean-pool non-padding tokens and project to 128 dimensions by frozen principal components (Pearson, 1901) fitted on each cohort's pretraining pool. We freeze the target during graph training and compare paired and mis-paired controls under the same calibration rule. Table 5 retains the comparison with the byte Transformer. The ratio of mean gains is 72%, with a nominal Fieller interval of 55–94%. Its adjusted half-gain p-values exceed 0.05, subject to the dependence limitations above. This comparison changes architecture, tokenization, projection and corpus as well as learned weights, so it cannot isolate the effect of target training.

Table 5: External SMILES targets under the same graph-training protocol. Positive differences favor the first arm. These are eight-cohort means and nominal 95% intervals, not a same-architecture isolation of target training.

Contrast	Mean	95% CI	Positive cohorts
ChemBERTa − graph-only	+0.0222	[+0.0163, +0.0281]	8/8
random − graph-only	+0.0159	[+0.0120, +0.0198]	8/8
trained-frozen − graph-only	+0.0162	[+0.0119, +0.0204]	8/8
ChemBERTa, mis-paired − graph-only	−0.0013	[−0.0034, +0.0008]	2/8
random − ChemBERTa	−0.0063	[−0.0117, −0.0010]	1/8

C Fresh-molecule confirmation after equal validation search

Data and separation. This experiment tests whether random-target gains remain when we tune the baseline separately and evaluate on fresh molecules (Section 3.2). Before drawing blocks, we deduplicate canonical non-isomeric identities, exclude all molecules in the controlled QM9 and Alchemy studies, and require finite features and an exact graph–SMILES round trip within 512 bytes. QM9 includes one to nine heavy atoms and HCNOF Alchemy includes nine to fourteen. The expanded size ranges mean that this experiment covers a different molecular population.

Table 6: Validation-selected settings for fresh confirmation. Each tuple lists the learning rate, calibrated auxiliary-to-main gradient ratio, and number of updates after pretraining. A zero ratio omits the auxiliary loss. † marks an exact reused graph-only fit. The auxiliary families share the same candidate grid, whereas graph-only receives its dedicated broader search. We lock these choices using the tuning block and evaluate them on three fresh blocks without further selection.

Arm	Fixed/raw	Fixed/unit norm	Selected/raw	Selected/unit norm
QM9				
Graph only	$(10^{-3}, 0, 1200)$	$(5\times10^{-4}, 0, 1200)$	$(3\times10^{-4}, 0, 4800)$	$(3\times10^{-4}, 0, 4800)$
Random SMILES	$(10^{-4}, 0.8, 1200)$	$(10^{-3}, 0, 1200)^{\dagger}$	$(10^{-4}, 0.8, 2400)$	$(3\times10^{-4}, 0, 4800)^{\dagger}$
Trained SMILES	$(10^{-4}, 0.8, 1200)$	$(3\times10^{-4}, 0.8, 1200)$	$(10^{-4}, 0.8, 2400)$	$(3\times10^{-4}, 0, 4800)^{\dagger}$
Random graph	$(3\times10^{-4}, 0.8, 1200)$	$(10^{-3}, 0, 1200)^{\dagger}$	$(10^{-3}, 0.2, 4800)$	$(3\times10^{-4}, 0, 4800)^{\dagger}$
Uniformity	$(10^{-4}, 0.8, 1200)$	$(10^{-4}, 0.8, 1200)$	$(3\times10^{-4}, 0, 4800)^{\dagger}$	$(3\times10^{-4}, 0, 4800)^{\dagger}$
Variance/covariance	$(10^{-3}, 0.8, 1200)$	$(10^{-3}, 0.8, 1200)$	$(3\times10^{-4}, 0.8, 4800)$	$(3\times10^{-4}, 0.8, 4800)$
Alchemy				
Graph only	$(5\times10^{-4}, 0, 1200)$	$(8\times10^{-4}, 0, 1200)$	$(8\times10^{-4}, 0, 4800)$	$(2\times10^{-3}, 0, 4800)$
Random SMILES	$(10^{-3}, 0.8, 1200)$	$(10^{-3}, 0.2, 1200)$	$(10^{-3}, 0.05, 4800)$	$(10^{-3}, 0, 4800)^{\dagger}$
Trained SMILES	$(10^{-3}, 0.2, 1200)$	$(10^{-3}, 0.2, 1200)$	$(10^{-3}, 0.05, 2400)$	$(10^{-3}, 0.2, 1200)$
Random graph	$(10^{-3}, 0.8, 1200)$	$(10^{-3}, 0.8, 1200)$	$(10^{-3}, 0.2, 4800)$	$(10^{-3}, 0.2, 4800)$
Uniformity	$(10^{-4}, 0.2, 1200)$	$(10^{-4}, 0.8, 1200)$	$(3\times10^{-4}, 0.8, 300)$	$(10^{-4}, 0.8, 1200)$
Variance/covariance	$(3\times10^{-4}, 0.8, 1200)$	$(3\times10^{-4}, 0.8, 1200)$	$(10^{-3}, 0.05, 4800)$	$(10^{-3}, 0.05, 4800)$

For each dataset, we draw one tuning block and three confirmation blocks without shared molecular identities. Each block contains 15,360 molecules divided into graph pretraining (4,096), text pretraining (4,096), ridge fitting (4,096), validation (1,024) and test (2,048). After pretraining, graph training uses the union of the graph and text pretraining pools. The two datasets share 10,391 selected identities, so we summarize them separately rather than pooling six blocks as independent population replications.

Models and objectives. The graph and SMILES encoders retain the three-layer, width-128 GINE and byte Transformer architectures. Five auxiliary properties supervise encoder training, while dipole and HOMO enter only probe fitting, validation selection and test evaluation. We compare graph-only training, frozen random and trained SMILES targets, a frozen random graph target, uniformity, and VICReg's variance and covariance terms without its cross-view invariance term. Random and trained SMILES targets start from the same initialization.

Validation selection. We first select learning rates for graph and text pretraining independently from $\{1, 3, 10\} \times 10^{-4}$, using mean auxiliary-property validation loss over three tuning seeds. Both encoders train for 1,200 base updates and supply their final checkpoints. We do not search teacher training duration or select an intermediate checkpoint, so teacher quality remains limited by this budget and rule. Each arm receives twelve settings and checkpoints at 300, 1,200, 2,400 and 4,800 updates. Graph-only searches learning rates $\{1, 2, 3, 5, 10, 20, 30, 50, 80, 100, 200, 300\} \times 10^{-5}$. Each auxiliary arm crosses learning rates $\{1, 3, 10\}\times10^{-4}$ with initial auxiliary-to-property gradient ratios 0, 0.05, 0.2 and 0.8. These gradient ratios determine the loss coefficients. Zero strength reuses the corresponding graph-only fit. Equal candidate counts do not equalize search spaces or compute costs.

Mean downstream validation error over the three tuning seeds selects graph-training settings separately for raw and row-normalized probes. Three-fold cross-validation within the probe-fitting pool selects each ridge penalty. The evaluation protocols with fixed training duration use 1,200 updates, while those with validation-selected duration also choose the checkpoint. Before evaluating the confirmation blocks, we fix the neural training settings. We then train three new seeds per new block without selecting settings again on its validation data. Table 6 lists the selections.

Evaluation protocols and aggregation. Under the primary evaluation protocol, we use raw features at 1,200 updates and average dipole and HOMO test MAEs, each divided by its fitting-label standard deviation. The three secondary protocols check the effects of normalizing each feature vector to unit Euclidean norm, selecting the number of updates, or doing both. Raw and normalized readouts both subsequently center and scale coordinates using the probe-fitting data. We average model seeds within each block and then average the three blocks. Percentage gains divide the mean paired gain by the mean graph-only error. Figure 3 shows variation in the three observed block contrasts. These are not confidence intervals: the comparison uses only three blocks per dataset and one fixed tuning

Table 7: Fresh-molecule results under the primary evaluation protocol, using raw-feature ridge probes after 1,200 updates from the pretrained checkpoint. We average the two task-wise standardized MAEs within each run, average three model seeds within each molecule-disjoint block, and then average the three blocks. The block columns report these seed-averaged errors. Gain is the paired graph-only error minus the arm error, so positive values favor the arm. Every non-baseline arm improves error in all three blocks of both datasets. The block values describe variation and do not constitute confidence intervals.

Arm	Mean sMAE	Block 1	Block 2	Block 3	Mean gain
QM9					
Graph only	0.4513	0.4554	0.4637	0.4348	0.0000
Random SMILES	0.4285	0.4328	0.4455	0.4073	+0.0228
Trained SMILES	0.4304	0.4347	0.4471	0.4095	+0.0209
Random graph	0.4336	0.4428	0.4466	0.4112	+0.0177
Uniformity	0.4413	0.4465	0.4510	0.4265	+0.0100
Variance/covariance	0.4417	0.4507	0.4523	0.4221	+0.0096
Alchemy					
Graph only	0.4487	0.4549	0.4418	0.4494	0.0000
Random SMILES	0.4351	0.4426	0.4247	0.4381	+0.0136
Trained SMILES	0.4363	0.4442	0.4269	0.4379	+0.0124
Random graph	0.4297	0.4340	0.4262	0.4290	+0.0190
Uniformity	0.4363	0.4450	0.4276	0.4362	+0.0124
Variance/covariance	0.4400	0.4464	0.4320	0.4416	+0.0087

Table 8: Mean paired standardized-MAE gains under the three secondary evaluation protocols. We average three seeds within each block and then three blocks. Positive values favor the arm against its separately tuned graph-only comparator. Figure 3 shows every block contrast. † marks zero strength, which reuses a graph-only fit. A nonzero gain for such a choice reflects a different graph setting from the comparator and cannot establish an active auxiliary effect. We select all settings and training durations on the tuning block before confirmation.

Dataset	Arm	Fixed/unit norm	Selected/raw	Selected/unit norm
QM9	Random SMILES	$-0.0022^{\dagger}$	+0.0111	$0.0000^{\dagger}$
	Trained SMILES	+0.0119	+0.0144	$0.0000^{\dagger}$
	Random graph	$-0.0022^{\dagger}$	+0.0111	$0.0000^{\dagger}$
	Uniformity	+0.0063	$0.0000^{\dagger}$	$0.0000^{\dagger}$
	Variance/covariance	+0.0167	+0.0050	+0.0166
Alchemy	Random SMILES	+0.0040	+0.0015	$-0.0021^{\dagger}$
	Trained SMILES	+0.0068	+0.0019	-0.0089
	Random graph	+0.0085	+0.0100	-0.0065
	Uniformity	+0.0173	+0.0030	+0.0017
	Variance/covariance	+0.0152	+0.0019	+0.0081

choice. We do not treat nine block–seed cells as independent datasets, apply the controlled study's half-gain rule, or estimate uncertainty over future populations and repeated tuning. Similar mean gains do not establish equivalence. Tables 7 and 8 report primary errors and secondary gains.

Interpreting selected zero strength. When validation selects zero strength, the resulting model trains without an auxiliary loss. On QM9, the random SMILES, trained SMILES, random graph and uniformity arms with normalized probes and selected training duration all reuse the selected graph-only fit and therefore have exactly zero gain. Other zero-strength choices select a different graph learning rate from the separately tuned comparator and can have nonzero gains (Table 8). Their negative gains cannot demonstrate harm from active alignment. For the active Alchemy random-SMILES arm with a raw probe and selected training duration, the small positive two-task mean also masks a worse HOMO mean.

D GRAPHMVP PRETRAINING AND DOWNSTREAM ROBUSTNESS

Pretraining and targets. We test random targets in an established molecular pretraining method by using GraphMVP's released objective (Section 3.3). Its encoders are a five-layer, width-300 GIN and a SchNet network with six interactions, 51 radial basis functions and a 10 Å cutoff. We pretrain on 50,000 GEOM molecules (Axelrod & Gómez-Bombarelli, 2022) with five conformers each. The objective combines energy-based contrastive loss and variational reconstruction with equal weights, normalized contrastive features, temperature 0.1, squared reconstruction loss, detached reconstruction targets and variational coefficient one. Pretraining lasts 100 epochs with batch size 256, no masking or dropout, and a SchNet learning-rate multiplier of 0.1. Both pretraining and fine-tuning use Adam (Kingma & Ba, 2015) with learning rate 10^{-3} and zero weight decay.

The co-trained arm updates both encoders. The trained-frozen arm loads the co-trained 3D encoder and trains a fresh graph encoder and reconstruction modules for 100 epochs while holding that target fixed. The random-frozen arm instead freezes a randomly initialized 3D encoder of the same architecture. Scratch skips pretraining. The trained-frozen arm therefore incurs additional teacher-pretraining cost, and the comparison does not match total compute.

Downstream protocol and matching. For every data partition, we fine-tune each of three pretrained graph encoders per arm using three downstream seeds. Three scratch fits per partition serve all pretraining seeds. Full graph-encoder fine-tuning lasts 100 epochs with batch size 256 and dropout 0.5, using mean pooling over the final layer. We match downstream head initialization, minibatch order and dropout seeds across arms. Graph initialization also matches by pretraining seed, but we restore only the CPU random-number state around random-target initialization. Reconstruction-module initialization and subsequent training streams need not match across pretraining arms.

Scaffold partitions and evaluation protocols. For ClinTox, BACE and SIDER, we accept the first five distinct random scaffold partitions in which every task has both classes in training, validation and test. We check this requirement before fitting any model. SIDER rejects one candidate, and some accepted SIDER test tasks contain only one minority-class example. The results therefore concern partitions that satisfy this label-coverage requirement. Both evaluation protocols use the same fitted models.

The primary evaluation protocol reports test macro ROC-AUC at the first full-precision validation maximum. The secondary evaluation protocol uses epoch 100, and no test trajectory selects a model. We average crossed seeds within each partition, then average the five partitions equally. Partitions share molecules and pretrained models. Their contrasts, ranges and counts describe variation for this fixed set of pretrained models; we do not report confidence intervals or tests requiring independent partitions.

Results. Under validation selection, random targets improve mean AUC over scratch by 5.17 percentage points on ClinTox and 0.90 points on BACE, but lower it by 1.25 points on SIDER. The respective partition contrasts range from -1.25 to $+16.19$, -4.15 to $+6.01$, and -3.71 to $+1.55$ points. Random targets improve over scratch in four, two and one of five partitions, so the positive BACE mean does not describe most partitions. Frozen trained targets have higher mean AUC than random targets on all three datasets under both evaluation protocols (Table 9). Random targets beat them in one ClinTox partition and no BACE or SIDER partitions under validation selection. Random targets thus help relative to scratch on ClinTox, but these results do not support replacing trained targets in general.

Table 9: GraphMVP test macro ROC-AUC (%). We average seeds within each partition and then five partitions with equal weight. Selected denotes each run's first validation maximum and final denotes epoch 100. Trained and random denote frozen auxiliary targets. Differences use unrounded means. Figure 4 shows the partition-level contrasts. Partitions share molecules and pretrained encoders, so their variation is descriptive rather than a confidence interval from independent replications.

Dataset	Evaluation protocol	Scratch	Co-trained	Trained	Random	Random − trained
ClinTox	Selected	64.14	71.65	71.52	69.30	-2.21
ClinTox	Final	70.46	74.47	74.76	72.11	-2.65
BACE	Selected	83.70	85.91	85.51	84.60	-0.91
BACE	Final	83.15	86.05	85.66	84.87	-0.80
SIDER	Selected	60.56	61.37	61.52	59.31	-2.21
SIDER	Final	59.83	60.99	60.86	58.92	-1.94

E SAME-INPUT CONTROLS, REPRESENTATION GEOMETRY AND READOUT

These controls examine the need for a second view, feature geometry and readout choice (Section 3.4). They retain the common-calibration protocol and eight cohorts. Controls without an auxiliary view share runs across the 3D and SMILES comparisons. Training pools overlap, so means and variation are descriptive; nominal intervals and tests have the limitations in Appendix A.1.

E.1 CONTROLS WITHOUT A SECOND VIEW

A random graph target is a separately initialized, frozen copy of the student's architecture reading the same graph. Fixed random codes assign each training molecule one 128-dimensional Gaussian vector independently of its structure. Both remove the second view while retaining the contrastive objective and gradient-ratio calibration. The codes provide no prediction rule for unseen molecules.

Two further controls encourage varied features without any target. Uniformity (Wang & Isola, 2020) penalizes concentration of row-normalized minibatch embeddings with $t = 2$. Our VICReg-derived control (Bardes et al., 2022) uses only the variance and covariance terms, with weights 25:1 and variance threshold $\gamma = 1$. We scale embeddings by the base encoder's mean per-dimension standard deviation and omit the cross-view invariance term. We add either penalty to the property loss and calibrate its gradient ratio to 0.2.

Random graph targets and fixed codes improve the raw-probe mean on QM9, Alchemy and QMugs (Table 10). On QM9 their error reductions are approximately 3.3% and 2.2%, respectively. The target-free penalties have smaller positive Alchemy means with nominal intervals spanning zero, whereas uniformity gives a larger QMugs mean gain than either cross-modal random target. In the half-gain sensitivity analysis against the random 3D target, QM9 uniformity has an unadjusted $p < 0.05$ but a Holm-adjusted $p = 0.084$ over the two regularizer comparisons. These observed gains show that a second modality is unnecessary in some tested settings, without establishing a consistent advantage of alignment over these penalties.

E.2 ORDINARY REGULARIZERS

For SMILES, we also test familiar ways to regularize training: three weight-decay strengths, two dropout rates, attraction to the initialization, an exponential moving average and Gaussian embedding noise. We calibrate coefficients by gradient ratio where applicable and select the best option on validation. Dropout at 0.1 gives a mean gain of 0.0015 with nominal 95% interval $[-0.0002, 0.0032]$, compared with 0.0159 for random-SMILES alignment. Changing the random-target objective to squared-error matching gives a gain of -0.0066 with interval $[-0.0098, -0.0035]$. This restricted comparison excludes the stronger uniformity and variance/covariance controls above.

Table 10: Same-input targets and target-free penalties at 1,200 updates and 4,096 fitting labels. Entries give mean standardized-MAE reductions relative to graph-only training and nominal paired-t 95% intervals across eight cohorts. The 3D and SMILES comparisons share these runs. These nominal intervals assume independent cohorts and do not account for overlapping training pools (Appendix A.1).

Dataset	Control	Gain over graph-only	95% CI
QM9	RANDOM GRAPH	$+0.0146$	$[+0.0096, +0.0197]$
	RANDOM CODES	$+0.0098$	$[+0.0071, +0.0126]$
	UNIFORMITY	$+0.0068$	$[+0.0027, +0.0108]$
	VICREG	$+0.0055$	$[+0.0028, +0.0082]$
Alchemy	RANDOM GRAPH	$+0.0137$	$[+0.0085, +0.0190]$
	RANDOM CODES	$+0.0100$	$[+0.0063, +0.0136]$
	UNIFORMITY	$+0.0044$	$[-0.0024, +0.0112]$
	VICREG	$+0.0018$	$[-0.0017, +0.0052]$
QMugs	RANDOM GRAPH	$+0.0132$	$[+0.0099, +0.0165]$
	RANDOM CODES	$+0.0118$	$[+0.0070, +0.0166]$
	UNIFORMITY	$+0.0148$	$[+0.0129, +0.0167]$
	VICREG	$+0.0065$	$[+0.0039, +0.0091]$

Table 11: Readout choice changes the comparison on QM9. Gains in standardized MAE are means $\pm$ SD of paired differences across eight cohorts, and positive values favor the intervention. Rank is the participation ratio of the raw embedding, with baseline 1.9. The two settings share controls requiring no auxiliary view. Each readout uses its own matched graph-only baseline.

Setting	Intervention	Raw probe	Unit-norm probe	Rank
SMILES	RANDOM	$+0.0159 \pm 0.0047$	$+0.0065 \pm 0.0077$	4.2
	TRAINED-FROZEN	$+0.0162 \pm 0.0051$	$+0.0135 \pm 0.0051$	6.7
3D	RANDOM	$+0.0083 \pm 0.0034$	-0.0238 ± 0.0058	3.2
	TRAINED-FROZEN	$+0.0119 \pm 0.0040$	-0.0020 ± 0.0066	5.4
Shared	RANDOM GRAPH	$+0.0146 \pm 0.0060$	$+0.0027 \pm 0.0073$	5.2
	RANDOM CODES	$+0.0098 \pm 0.0032$	$+0.0118 \pm 0.0044$	7.8
	UNIFORMITY	$+0.0068 \pm 0.0049$	$+0.0133 \pm 0.0039$	21.8
	VICREG	$+0.0055 \pm 0.0032$	$+0.0189 \pm 0.0033$	8.2

E.3 RANK AND UNIFORMITY MEASURE DIFFERENT CHANGES

Participation-ratio rank (Gao et al., 2017) measures how broadly feature variance spans directions. For fitting-pool embeddings with covariance eigenvalues s_i, it is

$$r_{\mathrm{PR}} = \frac{(\sum_i s_i)^2}{\sum_i s_i^2}.$$

This measure is unchanged by global rescaling and differs from entropy-based effective rank (Roy & Vetterli, 2007). Uniformity instead measures separation after row normalization through $\log \mathbb{E}_{i \neq j} \exp(-2\|\bar{z}_i - \bar{z}_j\|_2^2)$ (Wang & Isola, 2020); lower values indicate greater spread on the sphere. Neither diagnostic uses downstream labels.

The QM9 baseline has participation-ratio rank about 1.9. Several successful interventions increase it, but uniformity produces the largest rank without the largest raw-probe gain (Table 11). These measurements describe representation changes without establishing that expansion causes the gains.

E.4 CHANGING THE READOUT

To check whether feature magnitudes matter, we fit ridge regression to the raw embedding z, the row-normalized embedding $z/\|z\|_2$, or the scalar norm $\|z\|_2$. We report all three without selecting among them. Each uses a matched graph-only baseline and fitting-pool ridge selection.

Row normalization reverses the random-3D mean gain on QM9, Alchemy and QMugs (Table 12). The trained-3D means also become negative. Random-SMILES means remain positive on QM9 and

Table 12: Readout ablation after 1,200 updates with 4,096 fitting labels. Entries give mean standardized-MAE reductions across eight cohorts, using a matched graph-only baseline for each probe. The ChemBERTa row compares trained ChemBERTa-2 with the random scratch SMILES Transformer. Generated 3D uses graph-generated conformers.

	Raw		Unit norm		Norm only	
Setting	Trained	Random	Trained	Random	Trained	Random
QM9, 3D	$+0.0119$	$+0.0083$	-0.0020	-0.0238	-0.0343	-0.0307
QM9, SMILES	$+0.0162$	$+0.0159$	$+0.0135$	$+0.0065$	-0.0353	-0.0308
QM9, ChemBERTa	$+0.0222$	$+0.0159$	$+0.0361$	$+0.0065$	-0.0378	-0.0308
QM9, generated 3D	$+0.0116$	$+0.0089$	-0.0025	-0.0238	-0.0343	-0.0306
Alchemy, 3D	$+0.0089$	$+0.0075$	-0.0013	-0.0313	-0.0544	-0.0495
Alchemy, SMILES	$+0.0090$	$+0.0102$	$+0.0125$	$+0.0025$	-0.0615	-0.0593
QMugs, 3D	$+0.0092$	$+0.0106$	-0.0148	-0.0205	-0.0027	-0.0026
QMugs, SMILES	$+0.0060$	$+0.0032$	-0.0080	-0.0178	-0.0019	-0.0035

Alchemy but turn negative on QMugs. On QM9, fixed codes and the two anti-collapse penalties retain or improve their mean gains (Table 11). Feature magnitude therefore matters to the comparison. Yet magnitude alone does not explain the gains: the norm-only probe does not recover them.

E.5 Prediction gains need not increase student–target similarity

We measure student–target similarity with linear centered kernel alignment (CKA) (Kornblith et al., 2019). Each arm uses its frozen target and the same 4,096 fitting molecules for the aligned student, matched graph-only model and mis-paired control. For raw embedding matrices Z, T and the centering matrix H, we compute

$$\mathrm{CKA}(Z,T) = \frac{\langle HZZ^\top H, HTT^\top H\rangle_F}{\|HZZ^\top H\|_F \|HTT^\top H\|_F}.$$

This retrospective diagnostic uses raw embedding rows, whereas the contrastive objective normalizes them.

For standard QM9 3D and SMILES targets, random and trained targets lower CKA and improve raw-probe prediction in every cohort (Table 13). Alchemy repeats the decrease in every cohort, with prediction gains in seven or eight per arm. Dynamically mis-paired controls largely retain baseline CKA.

ChemBERTa instead increases CKA in every cohort, and QMugs random SMILES increases it in seven. The PCQM4Mv2 random-3D arm lowers CKA in every cohort but gives a small positive mean prediction gain whose nominal interval spans zero. CKA therefore changes in different directions across settings and does not provide a common explanation for the gains. It is invariant to orthogonal transformations and global rescaling, but not arbitrary invertible linear maps, so a decrease does not rule out useful linear transfer.

E.6 Teacher fitting and residual utility

These retrospective QM9 checks examine whether students predict the teacher, retain the graph-only predictor, and gain features that correct its errors, as relevant to Proposition 1. We use the 2,400-update comparison, chosen by checkpoint availability without inspecting metric values. The learning rate is 3×10^{-4}, with 4,096 fitting labels, 1,024 validation molecules for ridge-penalty selection, and 2,048 test molecules per pool. Here the metric averages dipole and HOMO MSE divided by each task's fitting-label variance; it differs from the standardized MAE in Section 2.2. Downstream task probes use graph embeddings alone. The eight cohorts have disjoint in-distribution tests with at most eight heavy atoms, overlapping training pools, and one shared nine-heavy-atom test pool. We therefore report descriptive means and direction counts.

Task gains and teacher fitting. Paired random SMILES lowers the two-task error in both test pools and every cohort, whereas shuffled pairing worsens both pool means (Table 14). Random 3D targets give smaller positive mean gains. Random-SMILES training also lowers raw and row-normalized teacher-prediction error (Table 15) and raises entropy effective rank (Roy & Vetterli, 2007) from

Table 13: **Global similarity and prediction gains can move in opposite directions.** Raw linear CKA at 1,200 updates, averaged across eight cohorts. All columns in a row use the same frozen target. Δ subtracts graph-only CKA from paired CKA, with nominal paired-t 95% intervals. The final column counts cohorts with lower paired CKA. Mis-pairing redraws partners each minibatch. A dash denotes an unavailable diagnostic. The nominal intervals do not account for overlapping training pools (Appendix A.1), and cohort counts are descriptive.

Setting	Target	Graph-only	Paired	Mis-paired	Δ CKA [95% CI]	Lower CKA
QM9, 3D	Random	0.933	0.658	0.939	-0.275 $[-0.309, -0.242]$	8/8
	Trained	0.934	0.736	0.938	-0.198 $[-0.248, -0.149]$	8/8
QM9, SMILES	Random	0.883	0.793	0.904	-0.091 $[-0.121, -0.060]$	8/8
	Trained	0.914	0.700	–	-0.215 $[-0.252, -0.177]$	8/8
Alchemy, 3D	Random	0.982	0.683	0.983	-0.298 $[-0.323, -0.274]$	8/8
	Trained	0.948	0.770	0.945	-0.178 $[-0.206, -0.149]$	8/8
Alchemy, SMILES	Random	0.947	0.870	0.956	-0.077 $[-0.092, -0.061]$	8/8
	Trained	0.948	0.649	0.954	-0.298 $[-0.333, -0.264]$	8/8
QMugs, 3D	Random	0.959	0.650	0.969	-0.309 $[-0.327, -0.291]$	8/8
	Trained	0.964	0.704	0.971	-0.260 $[-0.284, -0.236]$	8/8
QMugs, SMILES	Random	0.891	0.921	0.926	$+0.031$ $[+0.008, +0.053]$	1/8
	Trained	0.918	0.806	0.938	-0.112 $[-0.148, -0.076]$	8/8
PCQM4Mv2, 3D	Random	0.928	0.607	0.933	-0.321 $[-0.354, -0.288]$	8/8
	Trained	0.940	0.714	0.944	-0.226 $[-0.244, -0.209]$	8/8
PCQM4Mv2, SMILES	Random	0.835	0.818	0.854	-0.017 $[-0.111, +0.077]$	4/8
	Trained	0.893	0.662	0.903	-0.232 $[-0.299, -0.164]$	8/8
QM9, generated 3D	Random	0.933	0.658	0.939	-0.275 $[-0.308, -0.242]$	8/8
	Trained	0.948	0.733	0.950	-0.214 $[-0.263, -0.166]$	8/8
QM9, ChemBERTa	Trained	0.154	0.534	0.138	$+0.380$ $[+0.366, +0.393]$	0/8

3.35 to 6.59. Entropy rank differs from the participation ratio in Figure 6. For random 3D targets, graph-only features already give low raw prediction error, while row-normalized prediction worsens after auxiliary training despite task gains. For raw features, we divide each coordinate's prediction error by its centered fitting-pool variance. Thus, a large constant teacher mean cannot by itself explain the low error.

Preservation and residual utility. The random-SMILES student's linear probe predicts graph-only ridge outputs with mean R^2 of 0.933 on in-distribution and 0.935 on shifted molecules. The shuffled control predicts these outputs more accurately while worsening the task. This comparison concerns a finite-label predictor at the same update count, not the proposition's initial oracle predictor.

We test whether teacher features correct graph-only errors by fitting a teacher-only ridge correction to nested cross-fitted graph-only residuals. Random and trained SMILES corrections worsen shifted task NMSE in every cohort, although their auxiliary-trained graph representations improve in every cohort. The random correction increases mean shifted NMSE by 0.01437 (Table 15). This correction does not reproduce the student's gain. It also does not test the proposition directly: we do not estimate the population approximation term or coefficient bound, and coordinate-normalized ridge error is not the vector fitting error ϵ^2 assumed there.

Table 14: Retrospective QM9 diagnostics at 2,400 updates. Positive values denote relative two-task NMSE reduction against the same-step graph-only representation, computed within each cohort and then averaged. Counts show cohorts with a positive reduction out of eight. ID denotes the in-distribution test pool, and Shift denotes the nine-heavy-atom test pool. These NMSE diagnostics do not replace the standardized-MAE results under the primary evaluation protocol.

View	Target	ID gain (%)	ID count	Shift gain (%)	Shift count
3D	Frozen trained	+2.66	6/8	+1.10	5/8
3D	Frozen random	+1.90	6/8	+1.29	6/8
3D	Shuffled random	-0.69	2/8	-0.97	0/8
SMILES	Frozen trained	+5.62	8/8	+3.53	8/8
SMILES	Frozen random	+4.66	8/8	+4.62	8/8
SMILES	Shuffled random	-1.29	1/8	-1.38	1/8

Table 15: Predicting a frozen random teacher from QM9 graph features at 2,400 updates. Each cell averages eight cohorts. Arrows run from the same-step graph-only representation to the auxiliary-trained student. Raw error averages coordinate MSE divided by centered fitting-pool variance. The normalized diagnostic first row-normalizes both teacher and student. Residual gain is the absolute task-NMSE reduction from adding a separate teacher-based correction to the graph-only predictor, not from auxiliary student training.

View	Test	Raw teacher NMSE	Normalized teacher NMSE	Residual gain
3D	ID	$0.000742 \rightarrow 0.000673$	$0.0134 \rightarrow 0.0661$	$+0.00578$
3D	Shift	$0.001040 \rightarrow 0.000907$	$0.0120 \rightarrow 0.0687$	$+0.00024$
SMILES	ID	$0.1035 \rightarrow 0.0756$	$0.1671 \rightarrow 0.1424$	$+0.00259$
SMILES	Shift	$0.2305 \rightarrow 0.2123$	$0.2569 \rightarrow 0.2485$	-0.01437

Downstream GraphMVP geometry. At each arm's validation-selected checkpoint on SIDER, random-target pretraining raises entropy effective rank from 7.85 to 8.79 relative to scratch and lowers uniformity loss while lowering AUC. Here, more varied features accompany worse prediction, showing why expansion alone is not enough to explain a gain. We lack paired downstream 3D teacher features for teacher-prediction or residual-utility diagnostics.

F Target assignment and pairing controls

We test whether the correct partner matters and whether a fixed assignment is enough to help (Section 3.4). We retain the paired arm's graph checkpoint, minibatches, target network and calibration rule. Nominal intervals remain sensitivity summaries subject to the dependence assumptions in Appendix A.1.

Three assignment controls. Dynamic mis-pairing gives each molecule another molecule's target, chosen anew within every minibatch. Fixed arbitrary assignment draws one random cycle over the graph-training pool, so each molecule keeps the same other molecule's target throughout training. Fixed composition assignment restricts these cycles to groups with identical H, C, N, O and F counts; singleton groups keep their own targets. The last control preserves compositional information and some correct pairs.

Dynamic mis-pairing produces nonpositive mean gains in all six dataset–view comparisons (Table 16). Keeping arbitrary assignments fixed gives small positive means whose nominal intervals include zero. Except on QMugs SMILES, composition matching gives larger positive means and nominal intervals excluding zero. A stable assignment alone therefore does not reproduce the correctly paired target's mean gain at 1,200 updates. Composition matching cannot isolate stability because it also retains molecular information.

Longer training changes the arbitrary-assignment result. On QMugs, arbitrary fixed partners give larger positive mean gains in both views at 2,400 updates than at 1,200 updates (Table 17). The smaller 1,200-update gains therefore do not establish that such assignments cannot help. Benefits depend on assignment and duration, but these comparisons do not directly measure how easily students learn their targets or explain trained targets' property-specific advantages.

G Which molecular properties benefit from target training?

We compare additional properties using the controlled-study embeddings, without retraining the encoders. Means and cohort counts are descriptive; nominal intervals retain the dependence limitations in Appendix A.1. LUMO and three rotational constants pass a screen excluding strong association with the five pretraining labels at threshold 0.95 (Appendix A.2). Atomization energies fail this screen. We also report HOMO–LUMO gap but do not count it as an independent addition because it contains the existing HOMO task.

Raw rotational constants are unstable for nearly linear molecules: one molecule changes constant A's fitting standard deviation by orders of magnitude. We therefore report $\log_{10}$ probes of positive rotational constants (Table 19) alongside the orbital-energy probes (Table 18). All test rotational

Table 16: Pairing controls for frozen random targets at 1,200 updates and 4,096 fitting labels. Entries give mean standardized-MAE reductions relative to graph-only training and nominal paired-t 95% intervals across eight cohorts. Dynamic assignments change every minibatch. Fixed assignments persist, with composition matching restricted to identical elemental counts and self-pairing for singleton groups. Training pools overlap, so these are cohort summaries rather than independent dataset replications.

Dataset	View	Dynamic assignment	Fixed arbitrary	Fixed composition
QM9	3D	-0.0012 $[-0.0022, -0.0003]$	$+0.0013$ $[-0.0016, +0.0042]$	$+0.0082$ $[+0.0045, +0.0119]$
	SMILES	-0.0016 $[-0.0034, +0.0002]$	$+0.0014$ $[-0.0017, +0.0044]$	$+0.0134$ $[+0.0095, +0.0172]$
Alchemy	3D	-0.0036 $[-0.0053, -0.0018]$	$+0.0014$ $[-0.0011, +0.0040]$	$+0.0077$ $[+0.0029, +0.0124]$
	SMILES	-0.0030 $[-0.0048, -0.0012]$	$+0.0028$ $[-0.0012, +0.0068]$	$+0.0078$ $[+0.0038, +0.0118]$
QMugs	3D	-0.0004 $[-0.0029, +0.0020]$	$+0.0020$ $[-0.0005, +0.0046]$	$+0.0093$ $[+0.0057, +0.0129]$
	SMILES	-0.0019 $[-0.0037, -0.0000]$	$+0.0023$ $[-0.0016, +0.0062]$	$+0.0008$ $[-0.0020, +0.0035]$

Table 17: QMugs with arbitrary fixed partners for the random target. Both update counts include all eight cohorts. Entries give mean standardized-MAE reductions relative to graph-only training at the same update count, with nominal paired-t 95% intervals and counts of improved cohorts. Both views have larger mean gains at 2,400 updates, when their nominal intervals exclude zero.

View	Updates	Gain over graph-only	95% CI	Cohorts improved
3D	1,200	$+0.0020$	$[-0.0005, +0.0046]$	7/8
	2,400	$+0.0064$	$[+0.0035, +0.0093]$	8/8
SMILES	1,200	$+0.0023$	$[-0.0016, +0.0062]$	5/8
	2,400	$+0.0086$	$[+0.0037, +0.0135]$	7/8

constants are positive. We chose this scale after inspecting the raw-label errors, so the logarithmic result is a sensitivity analysis rather than a prespecified test.

Table 18: Orbital-energy raw-probe results. Gains are mean standardized-MAE reductions over eight cohorts, with positive-cohort counts in parentheses. Intervals on random minus trained are nominal 95% intervals. Atomization energies fail the contamination screen and do not enter these comparisons.

Property	View	Baseline	Trained gain	Random gain	Random − trained (95% CI)
LUMO	3D setting	0.233	$+0.0157$ (8)	$+0.0026$ (5)	-0.0131 $[-0.0149, -0.0114]$ (0)
	SMILES setting	0.233	$+0.0227$ (8)	$+0.0244$ (8)	$+0.0017$ $[-0.0027, +0.0062]$ (5)
gap	3D setting	0.304	$+0.0191$ (8)	$+0.0106$ (7)	-0.0085 $[-0.0127, -0.0044]$ (0)
	SMILES setting	0.304	$+0.0290$ (8)	$+0.0303$ (8)	$+0.0013$ $[-0.0044, +0.0070]$ (5)

G.1 Geometry computed from the graph

To test whether equilibrium DFT geometry is needed, we replace its distances with graph-generated conformers from fixed-seed RDKit ETKDG (Riniker & Landrum, 2015), relaxed with MMFF94 (Halgren, 1996). We recover formal charges from valence before embedding and keep atoms, bonds, labels and splits fixed. The trained target receives the same five-property pretraining, and all arms retain the common calibration rule. The random target's primary mean gain remains positive, and the trained target retains larger mean gains on LUMO and gap (Table 20). The mean gains thus persist when the graph supplies the geometry. This result leaves open explanations based on architecture or optimization.

G.2 Template text with and without downstream values

We compare text with and without downstream values. The facts template states formula, heavy-atom count, ring count and bond-type counts, so some isomers share a description. The property template appends dipole and HOMO to three significant figures. We encode descriptions only for

Table 19: Retrospective probes of rotational constants on the $\log_{10}$ scale. All test values are positive. Errors are standardized MAE on logarithmic labels. Gains average eight cohorts, with positive counts in parentheses and nominal 95% intervals. Random 3D targets hurt all three property means while trained targets help.

View	Property	Baseline	Trained gain	Random gain	Random − trained (95% CI)
3D	rotational A ($\log_{10}$)	0.410	+0.0073 (7)	−0.0095 (1)	−0.0168 [−0.0196, −0.0141] (0)
3D	rotational B ($\log_{10}$)	0.362	+0.0064 (7)	−0.0108 (1)	−0.0172 [−0.0223, −0.0121] (0)
3D	rotational C ($\log_{10}$)	0.269	+0.0044 (7)	−0.0101 (0)	−0.0145 [−0.0179, −0.0111] (0)
SMILES	rotational A ($\log_{10}$)	0.410	−0.0038 (2)	−0.0004 (3)	+0.0034 [−0.0030, +0.0097] (6)
SMILES	rotational B ($\log_{10}$)	0.362	−0.0028 (3)	+0.0017 (6)	+0.0046 [−0.0014, +0.0105] (5)
SMILES	rotational C ($\log_{10}$)	0.269	−0.0024 (3)	−0.0007 (5)	+0.0017 [−0.0040, +0.0075] (4)

Table 20: Property-specific gains with DFT or graph-generated geometry. Entries average eight cohorts and retain the paired random-minus-trained nominal 95% interval. Generated geometry preserves the trained target's larger mean gains on LUMO and gap.

Property	Geometry	Trained gain	Random gain	Random − trained (95% CI)
dipole μ	DFT	+0.0110 (8)	+0.0064 (7)	−0.0046 [−0.0092, −0.0000]
	RDKit	+0.0130 (8)	+0.0075 (7)	−0.0055 [−0.0098, −0.0012]
HOMO	DFT	+0.0128 (8)	+0.0102 (8)	−0.0026 [−0.0082, +0.0031]
	RDKit	+0.0103 (7)	+0.0104 (8)	+0.0001 [−0.0033, +0.0035]
LUMO	DFT	+0.0157 (8)	+0.0026 (5)	−0.0131 [−0.0149, −0.0114]
	RDKit	+0.0182 (8)	+0.0025 (5)	−0.0157 [−0.0185, −0.0130]
gap	DFT	+0.0191 (8)	+0.0106 (7)	−0.0085 [−0.0127, −0.0044]
	RDKit	+0.0191 (8)	+0.0104 (8)	−0.0087 [−0.0132, −0.0041]

the pretraining pool, without exposing test labels to encoder training. Both templates use the byte Transformer and five-property objective, with paired and mis-paired frozen random and trained targets. The trained target's mean gain remains below the random target's for both templates, with nominal paired intervals spanning zero (Table 21). This diagnostic does not define a deployable text benchmark or establish whether the target uses the numerical fields.

Table 21: Deterministic template-text controls. Gains average eight cohorts, with positive counts in parentheses and nominal 95% intervals. Appending downstream properties to pretraining descriptions does not establish a trained-target advantage. These descriptions are diagnostic inputs, not deployable annotations.

Template	Trained gain	Random gain	Random − trained (95% CI)
Graph facts	+0.0084 (8)	+0.0113 (7)	+0.0029 [−0.0044, +0.0102] (5)
Facts and properties	+0.0135 (8)	+0.0146 (8)	+0.0011 [−0.0052, +0.0074] (4)

G.3 Testing kernel alignment as an explanation

We ask whether targets that better reflect a property give students larger gains. For raw-label probes, we compare the trained-minus-random difference in fitting-pool kernel–property alignment (Cristianini et al., 2001) with the trained target's additional student gain. Across seven admissible properties, the Spearman correlations are weak and have opposite signs across views (Table 22). The permutation analyses assume exchangeable property-level pairs, even though the properties share molecules and include correlated orbital quantities. Their p-values are therefore exploratory sensitivity summaries; this comparison establishes neither a general association nor its absence.

H Limits across training duration, student and dataset

We vary alignment timing, student architecture, molecules and tasks to see where gains persist. Nominal intervals and tests retain the cohort-dependence limitations in Appendix A.1.

Table 22: Association between a trained target's excess kernel–property alignment and its additional student gain over the random target. Each setting uses seven admissible properties. We enumerate all 7! pair permutations, whose inferential interpretation assumes exchangeable property pairs. Shared molecules and correlated properties limit this assumption, so the p-values serve as exploratory sensitivity summaries.

View	Spearman ρ	Permutation p
3D	-0.18	0.713
SMILES	0.29	0.556

H.1 Training duration and base pretraining

We stop alignment after 100 or 300 updates or start it after 300 or 600 to distinguish early from sustained benefits. Minibatch streams and calibrated coefficients stay fixed. Trained-target schedules stop at 300 or start at 600. We evaluate 25–4,800 updates after pretraining and extend graph-only training to 9,600. The prespecified timing comparison asks whether stopping random alignment at 300 retains at least half the continuous arm's gain at 1,200. Separate comparisons pretrain both base and trained teacher for 300 or 4,800 updates instead of 1,200.

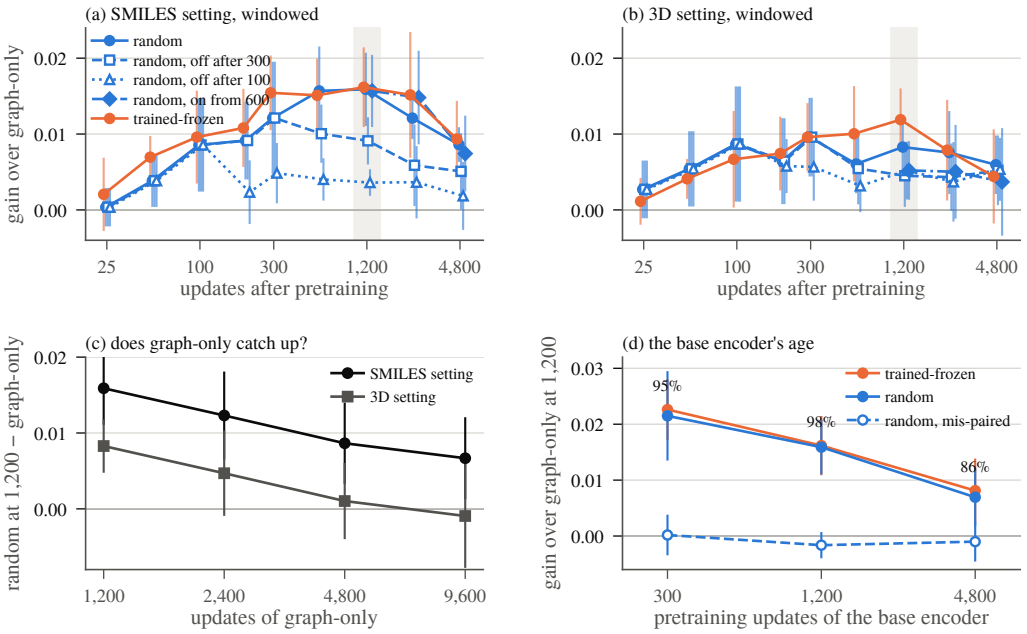


Figure 8: **Timing changes the observed alignment gain.** Stopping at 300 updates retains part of continuous alignment's mean gain at 1,200 updates. Delayed SMILES alignment approaches continuous alignment by point estimate. Compared with graph-only training for 9,600 updates, random alignment for 1,200 gives a positive SMILES mean difference and a slightly negative 3D mean difference. Less-pretrained base encoders show larger mean gains. These update-budget comparisons do not establish wall-clock speedups.

The delayed SMILES result does not support confining the benefit to early training (Table 23). Neither stopping at 300 nor using the longest-pretrained base reaches the half-gain threshold in the assumption-dependent tests.

Label-budget boundary. The number of labels available to fit the probe also changes the result. At 64 fitting labels, every alignment arm worsens the 3D mean. The nominal intervals for raw-probe gain first exclude zero at 1,024 labels for 3D and 256 for SMILES. These thresholds summarize this sweep and do not establish a minimum useful label budget. These are secondary comparisons to the primary evaluation protocol with 4,096 fitting labels.

Table 23: Timing contrasts supporting Figure 8. Positive standardized-MAE differences favor the first arm. Entries are eight-cohort means, nominal 95% intervals and positive-cohort counts. The late-start SMILES mean difference is close to zero. Comparing random alignment at 1,200 updates with graph-only at 9,600 gives a positive SMILES mean and a slightly negative 3D mean. Nominal intervals assume independent cohorts despite overlapping training pools (Appendix A.1).

View	Contrast	Mean	95% CI	Positive
SMILES	Stop at 300 minus continuous, at 1,200	-0.0068	$[-0.0107, -0.0029]$	0/8
SMILES	Start at 600 minus continuous, at 1,200	-0.0002	$[-0.0037, +0.0033]$	5/8
SMILES	Random at 1,200 minus graph-only at 9,600	$+0.0067$	$[+0.0023, +0.0111]$	7/8
3D	Stop at 300 minus continuous, at 1,200	-0.0038	$[-0.0062, -0.0014]$	1/8
3D	Start at 600 minus continuous, at 1,200	-0.0031	$[-0.0070, +0.0009]$	3/8
3D	Random at 1,200 minus graph-only at 9,600	-0.0009	$[-0.0065, +0.0047]$	2/8

H.2 Student architecture

We test two changes to model architecture. The width comparison increases graph and target encoder widths and probe dimensions from 128 to 512. The attention comparison retains width 128 and adds atom self-attention after each GINE block, inspired by the combination of local message passing and global attention in GPS (Rampášek et al., 2022), with unchanged targets. We recalibrate all arms. Widening preserves positive mean gains, but random-target alignment lowers the attention model's mean performance under the primary evaluation protocol (Table 24).

3D student. We also test a student that reads geometry, training graph and SchNet-style students together from their pretrained checkpoints for 2,400 updates at learning rate 3×10^{-4}. The five arms use no alignment, direct live alignment, a shared random SMILES target, separate random SMILES targets or a shared trained SMILES target. We calibrate each encoder separately. The graph encoder reuses the controlled-comparison runs. Each encoder has secondary raw and normalized ridge-probe evaluations. Random SMILES improves the 3D student's mean raw-probe performance on in-distribution molecules, but its size-shifted mean gain is negative. Normalization lowers point estimates of recovered fractions from 98% to 49% for the graph student and from 74% to 49% for the 3D student. These ratios describe mean gains rather than equivalence to the trained target. Appendix E.4 gives the broader readout comparison.

Table 24: Student variants under the primary evaluation protocol with raw-feature probes. Entries give mean standardized-MAE gain over the matched baseline and nominal 95% interval across eight cohorts. The 3D-student probes are secondary. The width experiment widens targets as well as students.

Student and target view	Random gain (95% CI)	Trained gain (95% CI)
3D student, SMILES target	$+0.0187$ $[+0.0134, +0.0239]$	$+0.0254$ $[+0.0183, +0.0324]$
Width 512, SMILES target	$+0.0354$ $[+0.0286, +0.0423]$	$+0.0417$ $[+0.0374, +0.0460]$
Attention student, SMILES target	-0.0124 $[-0.0189, -0.0060]$	-0.0023 $[-0.0078, +0.0032]$

H.3 PCQM4Mv2 with different tasks and supervision

From PCQM4Mv2 (Hu et al., 2021; Nakata & Shimazaki, 2017), we take 17,408 molecules with 10–16 heavy atoms and 2,048 with 17, restricted to H, C, N, O and F and selected in source order. We retain the controlled-study pool sizes and cohorts. Downstream tasks are HOMO–LUMO gap and the DFT conformer's radius of gyration. Auxiliary labels are Crippen $\log P$, polar surface area, rotatable-bond count, sp^3 fraction and donor/acceptor count. This setting changes molecules, auxiliary labels and tasks together, so it does not isolate molecular size. The trained 3D target gives larger mean gains on both properties, and the random target recovers 19% of its two-task mean gain, with a nominal Fieller interval of $[-10\%, 37\%]$ (Tables 25 and 2). The random-target mean is positive but its nominal gain interval spans zero. Recovery is therefore limited here, although the mean still improves on graph-only training. QMugs uses larger molecules and has a 116% random-3D point recovery (Appendix B), so molecular size alone does not explain the contrast. The

baseline participation-ratio rank is 1.6 versus QM9's 1.9, so a higher baseline rank does not explain this boundary. The SMILES mean gains are also smaller for random targets (0.0005) than trained targets (0.0056), with respective nominal 95% intervals $[-0.0052, 0.0063]$ and $[0.0008, 0.0103]$.

Table 25: PCQM4Mv2 property gains under the primary evaluation protocol. Entries average eight cohorts, with positive counts in parentheses and nominal 95% intervals. Both random-target property means are positive but smaller than the trained-target means.

property	baseline error	gain: trained	gain: random	random − trained (95 % CI)	gain: mis-paired	rec. %
HOMO–LUMO gap	0.389	+0.0072 (7)	+0.0016 (5)	−0.0056 [−0.0110, −0.0001] (1)	−0.0020 (2)	23
radius of gyration	0.402	+0.0140 (8)	+0.0023 (6)	−0.0117 [−0.0158, −0.0076] (0)	−0.0004 (5)	17

I Supporting theory: random features and student readouts

I.1 When can random targets supply useful features?

A random network can supply useful nonlinear features without learning them (Rahimi & Recht, 2008; Cho & Saul, 2009). We adapt this argument to a student that learns those features while retaining its initial best linear predictor. The target must help predict what that initial predictor misses.

Let $\mu(X) = \mathbb{E}[Y \mid X]$ be the regression function of the downstream label, let $\mathcal{H}_0$ be the span of linear readouts of the initial graph representation, and let p_0 be the L_2 projection of μ onto $\mathcal{H}_0$. For a representation Z, write $\mathcal{E}(Z) = \inf_w \mathbb{E}[(Y - w^\top Z(X))^2] - \mathbb{E}[(Y - \mu(X))^2]$ for the excess risk of its oracle linear probe.

Proposition 1 (An idealized sufficient condition)**.** Assume Y and the initial and trained student feature coordinates are square integrable, and draw teacher parameters from a fixed law ρ independently of (X, V, Y). Suppose the random scalar teacher features satisfy $\mathbb{E}_V \psi_\omega(V)^2 \le 1$ for ρ-almost every ω. Set $\phi_\omega(X) = \mathbb{E}[\psi_\omega(V) \mid X]$ and assume a measurable coefficient function a, fixed before drawing the features, satisfies

$$\|\mu - p_0 - \mathbb{E}_\omega[a(\omega)\phi_\omega]\|_{L_2} \le \eta, \qquad |a(\omega)| \le B.$$

Draw $\omega_1, \dots, \omega_m$ independently from ρ and set $T(V) = m^{-1/2}(\psi_{\omega_1}(V), \dots, \psi_{\omega_m}(V))$. Suppose p_0 remains a linear readout of the trained student Z for almost every teacher draw, and Z admits a linear head A with $\mathbb{E}_{\omega_{1:m},X,V}\|AZ(X) - T(V)\|^2 \le \epsilon^2$. Both Z and A may depend on the sampled teacher parameters. Then

$$\mathbb{E}_{\omega_{1:m}} \mathcal{E}(Z) \le \left(\eta + B/\sqrt{m} + B\epsilon\right)^2. \tag{2}$$

The bound separates three sources of error: what the target cannot capture (η), its finite width ($B/\sqrt{m}$), and the student's population fitting error ($B\epsilon$). When their squared sum falls below the initial probe's excess error $\|\mu - p_0\|_{L_2}^2$, random-target training improves the oracle probe on average over teacher draws. The auxiliary view may be the graph itself. Preserving p_0 and reaching a small *population* fitting error are assumptions, and neither follows from an unchanged architecture or from our contrastive objective. The proposition gives no guarantee for a probe fitted with finitely many labels or for optimizer convergence, and it does not predict the recovery fractions we measured. The following subsections give the proof and explain why an increase in rank alone is insufficient.

I.2 Scope and proof of Proposition 1

Fix a population distribution of (X, V, Y) with $\mathbb{E}Y^2 < \infty$. The initial feature coordinates are square integrable, and $\mathcal{H}_0$ is their finite-dimensional span, including a constant if the probe uses an intercept. Draw $\Omega = (\omega_1, \dots, \omega_m) \sim \rho^m$ independently of (X, V, Y) and fix the measurable coefficient function a before these draws. The feature and coefficient bounds hold for ρ-almost every ω. All random functions are jointly measurable. The student Z_Ω and head A_Ω may depend on the entire draw. For almost every draw, its coordinates are square integrable and its linear-readout span contains p_0.

The proof combines the standard random-feature averaging argument (Rahimi & Recht, 2008) with two steps: predicting auxiliary features from the graph and fitting those predictions with the student. Independent nonlinear units in a single layer satisfy the sampling model; coordinates of a deep random encoder need not.

Proof. Let $\|\cdot\|_2$ denote the scalar $L_2(P_X)$ norm and set

$$r = \mu - p_0, \qquad g = \mathbb{E}_\omega[a(\omega)\phi_\omega], \qquad g_\Omega = \frac{1}{m}\sum_{j=1}^{m} a(\omega_j)\phi_{\omega_j}.$$

Conditional Jensen gives $\|\phi_\omega\|_2^2 \le \mathbb{E}_V \psi_\omega(V)^2 \le 1$. Independence and centering in the Hilbert space $L_2(P_X)$ therefore give

$$\mathbb{E}_\Omega\|g_\Omega - g\|_2^2 = \frac{1}{m}\mathbb{E}_\omega\|a(\omega)\phi_\omega - g\|_2^2 \le \frac{1}{m}\mathbb{E}_\omega\|a(\omega)\phi_\omega\|_2^2 \le \frac{B^2}{m}. \tag{3}$$

Define $\Phi_\Omega(X) = \mathbb{E}[T(V) \mid X, \Omega]$ and $\beta_\Omega = m^{-1/2}(a(\omega_1), \dots, a(\omega_m))$. Then $g_\Omega = \beta_\Omega^\top \Phi_\Omega$ and $\|\beta_\Omega\| \le B$ for every draw satisfying the coefficient bound. Since $A_\Omega Z_\Omega(X)$ is a function of (X, Ω), conditional Jensen also gives

$$\mathbb{E}_{\Omega,X}\|A_\Omega Z_\Omega(X) - \Phi_\Omega(X)\|^2 \le \mathbb{E}_{\Omega,X,V}\|A_\Omega Z_\Omega(X) - T(V)\|^2 \le \epsilon^2.$$

Consequently $\mathbb{E}_{\Omega,X}|\beta_\Omega^\top A_\Omega Z_\Omega(X) - g_\Omega(X)|^2 \le B^2\epsilon^2$. The preservation assumption supplies a readout of Z_Ω equal to p_0. Thus $q_\Omega = p_0 + \beta_\Omega^\top A_\Omega Z_\Omega$ is another linear readout of the same student. The triangle inequality in $L_2(P_\Omega \otimes P_X)$ yields

$$\begin{aligned}\left(\mathbb{E}_\Omega\|q_\Omega - \mu\|_2^2\right)^{1/2} &\le \|g - r\|_2 + \left(\mathbb{E}_\Omega\|g_\Omega - g\|_2^2\right)^{1/2} \\ &\quad + \left(\mathbb{E}_\Omega\|\beta_\Omega^\top A_\Omega Z_\Omega - g_\Omega\|_2^2\right)^{1/2} \\ &\le \eta + B/\sqrt{m} + B\epsilon.\end{aligned}$$

For any predictor $q(X)$, squared-loss orthogonality gives $\mathbb{E}[(Y - q(X))^2] - \mathbb{E}[(Y - \mu(X))^2] = \|q - \mu\|_2^2$. Apply this identity conditional on Ω, then average over draws. The best linear readout is no worse than q_Ω, so squaring proves the proposition. The initial best readout has excess risk $\|\mu - p_0\|_2^2$, giving the stated sufficient condition for average improvement. □

I.3 What expansion does and does not imply

The proposition requires more than a varied set of features: their bounded linear combinations must approximate what the initial predictor misses. For example, $\psi_\omega(v) = \sigma(\omega^\top v)$ gives a random nonlinear network layer whose feature distribution and activation determine which residuals it can approximate. The kernel interpretation of random threshold and rectified networks (Cho & Saul, 2009) does not guarantee that they can approximate every downstream residual. The $m^{-1/2}$ term bounds one source of approximation error. It does not say that actual probe accuracy must improve whenever width increases.

Rank provides a separate, more limited statement about matching the target on a fixed collection of examples. Let $T \in \mathbb{R}^{n\times m}$ contain their teacher outputs and let $Z \in \mathbb{R}^{n\times d}$ contain student features. If Z has rank at most r, then every linear head A satisfies

$$\|ZA - T\|_F^2 \ge \sum_{j>r} s_j(T)^2, \tag{4}$$

where $s_j(T)$ are the singular values in decreasing order. To see this directly, let P project onto the columns of Z. Orthogonality gives $\|ZA - T\|_F^2 \ge \|(I - P)T\|_F^2$. A rank-$r$ projector captures at most the sum of the largest r eigenvalues of $TT^\top$. Thus a sufficiently accurate linear match to a target with substantial spectral mass beyond rank r rules out a rank-r student on those examples. It says nothing by itself about unseen examples or effective-rank statistics used in our plots.

Even full-rank features can contain no information useful for a property. Let $S_1, \dots, S_m, U$ be independent symmetric signs, let $X = V = (S_1, \dots, S_m, U)$ and let $Y = U$. A frozen random

linear network $T(V) = m^{-1/2}W(S_1, \ldots, S_m)^\top$, where $W \in \mathbb{R}^{m\times m}$ has independent rows sampled uniformly from the unit sphere, has full-rank covariance almost surely. A student $Z = (1, T)$ exactly predicts this teacher and preserves a constant baseline, yet its best linear probe still has excess risk 1 because all teacher coordinates are uncorrelated with Y. The downstream approximation assumption fails, despite perfect matching and maximal rank in the target dimension. Independent instance codes likewise have no assumed population relation to new molecules. Fitting them on a finite training set alone therefore does not establish the proposition's population condition.

Two gaps separate the proposition from our experiments. It assumes that training preserves the initial best predictor and achieves a population squared-error fit, neither of which we establish. It also does not analyze the normalized contrastive loss in Equation 1, feature norms or optimizer convergence. It does not require preserving every initial readout, since at fixed width this stronger condition would preclude a strict oracle gain whenever the initial feature span already has full dimension. The result concerns population squared error, whereas we measure standardized MAE with ridge probes fitted on finitely many labels. Its transferred coefficient involves A, for which we provide no norm bound or bound on the number of labels needed. For a stochastic auxiliary view, components unpredictable from X contribute irreducible matching error rather than new graph information.

J Details on Experimental Setups

Software and versions. We implement the molecular models in Python. The controlled experiments and fresh-molecule confirmation use PyTorch 2.5.1 with CUDA 12.4 (`2.5.1+cu124`). GraphMVP uses Python 3.10. Its downstream environment records PyTorch 2.4.1 with CUDA 12.4, PyTorch Geometric 2.6.1, NumPy 2.2.6, SciPy 1.15.3, scikit-learn 1.7.2 and RDKit 2026.3.6. These records cover the named pipelines rather than a common environment across all experiments. The supplementary code lists the dependency installation requirements.

Our controlled graph encoders implement message passing directly in PyTorch, while GraphMVP uses PyTorch Geometric. Molecular processing uses RDKit (Landrum et al., n.d.), with ASE supplying atomic data in GraphMVP. NumPy, SciPy, pandas and scikit-learn support numerical processing and evaluation. NetworkX supports graph utilities. The ChemBERTa-2 comparison uses Hugging Face Transformers. We use AdamW (Loshchilov & Hutter, 2019) for the controlled experiments and Adam (Kingma & Ba, 2015) for GraphMVP, with the settings in Appendices A and D.

Hardware. Table 26 lists the compute-server configuration for GPU experiments.

Table 26: Compute-server hardware specifications. GPU memory is reported in MiB.

Component	Specification
Architecture	x86_64
Operating system	Rocky Linux 8.10 (Green Obsidian)
CPU	AMD EPYC 9374F 32-Core Processor (×2)
GPU	NVIDIA H200
GPU memory	143771 MiB
System RAM	1.5 TiB

Datasets and auxiliary views. The controlled experiments use QM9 (Ramakrishnan et al., 2014), Alchemy (Chen et al., 2019) and QMugs (Isert et al., 2022). The larger-molecule comparison uses PCQM4Mv2 (Hu et al., 2021). The separate validation-search comparison uses fresh molecular blocks from QM9 and Alchemy. GraphMVP pretrains on GEOM conformers (Axelrod & Gómez-Bombarelli, 2022) and evaluates on the MoleculeNet tasks ClinTox, BACE and SIDER (Wu et al., 2018a). We generate canonical SMILES from molecular structures and construct template descriptions from specified molecular fields. Appendices A, B.1, C, D and H.3 give the processing rules and partitions. Appendix G.2 identifies the diagnostic templates that also contain downstream property values.

Table 27: Main software licenses. Package names link to the corresponding upstream license texts. Bundled components retain their own license notices.

Software	License
PyTorch	BSD-3-Clause
PyTorch Geometric	MIT
NetworkX	BSD-3-Clause
NumPy	BSD-3-Clause
SciPy	BSD-3-Clause
pandas	BSD-3-Clause
RDKit	BSD-3-Clause
scikit-learn	BSD-3-Clause
ASE	LGPL-2.1-or-later
Transformers	Apache-2.0
GraphMVP reference code	MIT