

Structural Hierarchy and Latent Geometry in Self-Supervised Molecular Representation Learning

by

David Sulu

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday, 18 September 2026 at 15:00.

Thesis committee:	Dr. J. M. Weber,	Pattern Recognition and Bioinformatics, TU Delft
	L. Di Fruscia,	Pattern Recognition and Bioinformatics, TU Delft
	Dr. K. A. Hildebrandt,	Computer Graphics and Visualisation, TU Delft

Faculty:	Faculty of Electrical Engineering, Mathematics and Computer Science, TU Delft
-----------------	---

Research group:	Pattern Recognition and Bioinformatics
------------------------	--



Structural Hierarchy and Latent Geometry in Self-Supervised Molecular Representation Learning

David Sulu, Lorenzo Di Fruscia, Jana Weber
Delft University of Technology

ABSTRACT

Chemical structure admits useful hierarchical abstractions: atoms assemble into rings, rings into scaffolds, and scaffolds give rise to families of related compounds that share a common core. Molecular self-supervised contrastive learning typically ignores this structure, forming positive pairs through random augmentations, while embedding molecules in Euclidean space. We study whether explicitly encoding a molecule’s Bemis–Murcko scaffold, and using it to supervise the molecular embedding, changes what the model learns, and whether this effect depends on the embedding geometry, comparing Euclidean and Lorentz contrastive objectives. Pre-trained on 750,000 molecules drawn from a one-million-molecule subset of PubChem and evaluated on MoleculeNet, the resulting model improves downstream property-prediction performance over a matched non-scaffold baseline on most benchmarks, while also learning a more scaffold-structured representation space in which molecules embed closer to their underlying scaffolds and to other molecules built on related cores. Training with the Lorentz objective sharpens this scaffold organization only in combination with the scaffold signal, and the effect does not translate into a consistent downstream advantage over Euclidean training.

1 INTRODUCTION

Learning good molecular representations underlies a wide range of computational chemistry tasks, from drug discovery to materials design, since nearly every property-prediction task begins by turning a molecule into a vector. Because the space of drug-like molecules is astronomically large [3, 24] while labeled data is scarce and expensive to obtain, the field has moved from fixed, hand-engineered descriptors such as ECFP [25] and SMILES strings [40] towards representations learned directly from data [26, 42, 52]. Among learned approaches, Graph Neural Networks (GNNs) have become a leading choice for operating directly on the molecular graph, since they respect its topology by construction [13, 28, 44, 46]. Sequence models over SMILES remain competitive [6], with the better inductive bias depending on the task, the data, and the compute budget.

A general strategy for the label-scarcity problem is self-supervised learning (SSL), which learns transferable representations from unlabeled data by solving a pretext task. Contrastive learning is a prominent instance: two augmented “views” of the same input are pulled together while different instances are pushed apart, yielding general-purpose features that transfer well after fine-tuning [5, 33]. This paradigm has been transferred to molecules by a now substantial body of work that pre-trains graph encoders on millions of unlabeled compounds [6, 19] and consistently improves downstream

property prediction over training from scratch [26, 30, 38, 39]. MolCLR [39] is the canonical instance of this family and remains its standard reference point, pre-training a Graph Isomorphism Network with edge features (GINE) [18, 45] on roughly ten million unlabeled compounds.

When the molecular graph is the only input, contrastive learning has no positive pairs without augmenting the graph, which makes the augmentation operator more than an implementation detail: it is the working definition of chemical equivalence that the model is trained to respect. Because the objective only pulls augmented views together rather than reconstructing what differs between them, whatever a transformation destroys becomes invisible to the representation. The operators in general use, such as atom masking, bond deletion, and subgraph removal, were inherited from computer vision and from generic graph contrastive learning [10, 16, 29, 48, 51], domains in which the input is highly redundant: occluding a patch of an image or dropping an edge of a social network leaves the underlying object intact. A molecular graph offers no such redundancy, since its connectivity represents its chemistry. Deleting a single bond can disconnect the graph, open an aromatic ring, or yield a structure that no longer corresponds to a chemically valid compound.

This absence of redundancy has a direct consequence for the training objective. A perturbation that is too aggressive produces a false positive: two views that no longer share a chemical identity, which the objective nevertheless pulls together, teaching invariance to precisely the structural changes that determine a molecule’s behavior [50]. The severity of this effect is governed by the augmentation strength, the fraction of the graph that is perturbed. This is a molecular problem: the same perturbation ratio that is harmless on an image or a social network is destructive on a graph whose connectivity is its semantics.

The treatment of negatives is a separate matter. Because they are drawn indiscriminately from the remainder of the batch, structurally related compounds are pushed apart irrespective of their similarity [38]. This is not a consequence of how the views are built but of how the contrastive objective is defined, and it would arise in any domain where distinct samples can be semantically equivalent. What is chemistry-specific is the criterion for deciding which negatives are spurious: molecules, unlike images, come with structural abstractions that state when two compounds belong to the same family, and such a criterion acts on the objective rather than on the augmentation.

A line of research has responded by making the augmentation itself chemically informed, and three broad strategies can be distinguished. The first constrains the perturbation so that the resulting graph remains chemically coherent, replacing arbitrary edits with substitutions licensed by chemical domain knowledge, such as bioisosteric replacement or edits guided by an element-level knowledge graph, so that variation is introduced without altering the semantics of the molecule [11, 12, 31]. The second raises the granularity of the perturbation from atoms and bonds to chemically meaningful fragments, using decomposition schemes such as BRICS [7] to ensure that what is added or removed is a functional unit rather than an arbitrary vertex [38]. The third preserves a designated structural invariant by construction: the multi-channel framework (MCL) [37] perturbs only the terminal side chains of a molecule, replacing them with fragments drawn from a candidate pool and bounding the edit to a few atoms, so that the Bemis–Murcko core [2] is retained in every generated view, and contrasts these scaffold-invariant perturbations using a margin derived from fingerprint similarity.

These approaches are effective, but they share two properties. The first is a matter of cost: fragment libraries, decomposition algorithms, and validity checks all have to run inside the view generation step. This is a weak objection on its own, since the cost is incurred once, at pre-training, and the resulting encoder is no more expensive to fine-tune or deploy than any other. The second property is the substantive one. In all three strategies the chemical constraint is enforced at the point where views are produced. Where the objective is modified at all, as in the fingerprint-derived margin of MCL [37] or the similarity-weighted negatives of [38], it reweights pairs that a chemically restricted sampler has already produced; it does not itself state what the representation must preserve. The invariance the encoder acquires is therefore whatever the restricted perturbation happened to leave intact, and it can only be tightened by restricting the sampler further.

Independently of how views are generated, chemistry supplies a structural abstraction that requires no domain machinery to obtain. The Bemis–Murcko scaffold [2] is the core ring system obtained by stripping a molecule’s side chains, and it is shared across an entire family of related compounds. Medicinal chemistry treats it as an organizing unit of chemical space: a scaffold defines a compound series, serves as a starting point for lead optimization, and underlies scaffold hopping, the search for a new core that retains a desired activity. The same abstraction governs the standard evaluation protocol on molecular benchmarks, where scaffold splits place molecules with distinct cores in different folds precisely in order to test generalization to unseen chemistry [18, 42]. Crucially for the self-supervised setting, the scaffold is computed from the molecule itself without labels, fragment libraries, or validity filtering, so any signal built on it leaves pre-training fully unsupervised. It is therefore available as a signal to the objective, and can be used whatever the augmentation does to the graph: even when a perturbation damages a view, the scaffold of the underlying molecule remains a fixed reference that the representation can be held to.

Existing work that brings the scaffold into the objective ties it to the sampler that produced the views. MCL [37] pairs it with a sampler that preserves the core by construction, so the signal reinforces an invariance the augmentation has already guaranteed. HiMol

[49] contrasts molecules against scaffold-derived structure within a fixed hierarchical scheme. Whether an explicit scaffold signal is sufficient on its own, applied to unconstrained augmentations, is left open.

The space itself is the second assumption. Both methods embed molecules in Euclidean space and score pairs by cosine similarity, as does the contrastive literature they build on. Chemistry, however, admits a coarse-to-fine organization resembling a hierarchy: atoms form rings, rings form scaffolds, and scaffolds give rise to compound families. Such hierarchies embed in Euclidean space only with high distortion, since the number of nodes grows exponentially with depth while Euclidean volume grows polynomially [27]. This makes hyperbolic geometry, whose volume grows exponentially and which acts as a continuous analogue of a tree [22, 23], a plausible inductive bias for such hierarchical structure; it has recently been shown to recover interpretable hierarchies in large-scale multimodal contrastive models [9].

Hyperbolic representations have begun to appear in the chemical and biomedical sciences, but predominantly either as a drop-in encoder for supervised property prediction [41] or in settings where the hierarchy is external to the molecule, such as link prediction over biomedical knowledge graphs. The hierarchy internal to a molecule has not been used to organize a self-supervised molecular representation space.

Three observations follow from the above. First, the augmentation strength that governs how much chemistry a view retains has been carried over between implementations without ever being checked against what it does to the molecular graph, so it is not known whether any operating point yields views that are both structurally intact and useful as positives. Second, even at a defensible operating point the augmentation remains chemically blind, and the response need not be to constrain the sampler: the chemical knowledge that existing methods inject through the augmentation could instead be supplied as a separate signal, leaving the stochastic augmentation unconstrained. Third, whatever signal is used, the space into which molecules are embedded is itself an untested assumption. This thesis therefore asks:

Can chemically grounded structural priors, introduced through the learning signal and through the geometry of the representation space rather than through the augmentation operator, produce molecular representations that are measurably better organized around chemical scaffolds and more useful for downstream property prediction?

We decompose this into three sub-questions:

- RQ1.** How does the strength of the stochastic graph augmentation affect the chemical validity of the generated views, and do any strengths preserve enough chemical meaning to serve as positives?
- RQ2.** Does aligning each molecule to its Bemis–Murcko scaffold, as an explicit positive in the objective, improve the learned representation when the stochastic augmentation is left unconstrained?
- RQ3.** Does training with a hyperbolic (Lorentz) rather than a Euclidean contrastive objective yield a representation that is more strongly organized around scaffolds, and does any

such gain in organization translate into better downstream performance?

We address these questions with a self-supervised framework that crosses scaffold alignment with the choice of latent geometry. Concretely, our contributions are:

- We quantify the structural cost of the standard graph augmentations directly on the graph the encoder consumes, measuring how far each strength fragments the molecule, and use this to bracket the usable range of augmentation strengths. Because the resulting bounds are necessary but not sufficient conditions on a usable view, we pre-train at both ends of the bracket rather than committing to one.
- We introduce an explicit Bemis–Murcko scaffold signal into the contrastive objective, aligning each molecule to its scaffold while leaving the stochastic augmentation unconstrained, and additionally examine a variant that withholds same-scaffold compounds from the negative set.
- We compare two latent geometries for this objective: the standard Euclidean space with cosine similarity, and the Lorentz model of hyperbolic space with the negative Lorentzian distance.
- We introduce a molecule–molecule retrieval probe that measures directly how far a representation space is organized around scaffolds, asking which molecules a query retrieves as its nearest neighbours rather than whether it can recover its own scaffold, and use it together with the MoleculeNet property-prediction benchmarks [42] to separate the effect of the scaffold supervision from that of the geometry.

2 BACKGROUND

This section introduces the machinery our method builds upon: the graph representation of a molecule and the Bemis–Murcko scaffold derived from it, the fingerprint similarity we use to score chemical relatedness at evaluation time, the graph neural network that encodes both molecule and scaffold, the contrastive objective and the stochastic augmentations that generate its positive pairs, and the Lorentz model of hyperbolic space in which we place the resulting embeddings.

2.1 Molecules as graphs

A molecule is naturally described as an undirected graph $G = (V, E)$, where the node set V corresponds to atoms and the edge set E corresponds to chemical bonds. This representation, unlike string encodings such as SMILES [40], makes the connectivity and topology of the molecule explicit, which is precisely the information that governs much of its chemical behavior.

Starting from a SMILES string, we use RDKit [21] to parse the molecule and extract its atoms and bonds. Each atom $v \in V$ is described by a feature vector x_v that records its atomic number and its chirality tag, and each bond $(u, v) \in E$ carries a feature vector e_{uv} encoding the bond type (single, double, triple, or aromatic) and a bond-direction tag (a stereochemical flag marking the orientation of directional bonds, e.g. the two configurations of a double bond). Because the graph is undirected, every bond is stored as a pair of directed edges (u, v) and (v, u) sharing the same feature vector.

Formally, a molecule is thus a tuple $G = (X, X_E, A)$ with a node-feature matrix $X \in \mathbb{R}^{|V| \times d_v}$, an edge-feature matrix $X_E \in \mathbb{R}^{|E| \times d_e}$, and the connectivity given by the adjacency / edge-index structure A . Atomic numbers are drawn from the set of chemical elements, and an additional mask token is reserved in the atom vocabulary to denote atoms that have been hidden by augmentation.

2.2 Molecular scaffolds and the Bemis–Murcko decomposition

The graph representation makes explicit a coarser structural unit that our method relies on throughout: the molecular scaffold. Bemis and Murcko [2] define the scaffold of a molecule (hereafter the Bemis–Murcko or BM scaffold) as the subgraph obtained by removing all acyclic side chains, the exocyclic substituents that decorate a compound, while retaining every ring system and the linker atoms that connect them. Two molecules that differ only in their peripheral substituents, e.g. a change of halogen or alkyl group at a single position, therefore reduce to the same scaffold, whereas a change to the ring system or to how rings are connected yields a different one. Because the decomposition operates purely on connectivity, it requires no label and no learned model: given a molecule’s graph (or, equivalently, its SMILES string), the scaffold is obtained deterministically and at negligible cost, which is what allows us to use it as a self-supervised training signal without compromising the unsupervised nature of pre-training. In practice we compute scaffolds with RDKit’s implementation of the Bemis–Murcko algorithm [21]. One structural class requires special handling: an acyclic molecule, having no ring system, reduces to an empty scaffold and is treated as scaffold-less throughout this thesis (Appendix A.6 quantifies how common this is in our data). The scaffold sits between the atom/bond level and the level of a full compound: many molecules share one scaffold, and scaffolds themselves can be organized into a hierarchy of increasingly generic cores, which is precisely the coarse-to-fine structure that motivates both the scaffold-aware objectives and the use of hyperbolic geometry.

2.3 Molecular fingerprints and Tanimoto similarity

Several of our evaluations need a model-free measure of how chemically similar two molecules are. We use the standard combination of a circular fingerprint and the Tanimoto coefficient. An *extended-connectivity fingerprint* (ECFP) [25], also called a Morgan fingerprint after the algorithm it builds on, represents a molecule as a fixed-length bit vector: for every atom, the substructure formed by its neighbourhood up to a given radius is hashed to a bit position, so each set bit records the presence of a particular local atom environment. ECFP4 uses a maximum diameter of four bonds (radius two); we compute 2,048-bit ECFP4 fingerprints with RDKit [21]. The *Tanimoto* (Jaccard) similarity between two fingerprints a and b is the fraction of set bits they share,

$$T(a, b) = \frac{|a \wedge b|}{|a \vee b|} \in [0, 1], \quad (1)$$

and is the conventional similarity measure for fingerprints in cheminformatics [1]. Because a fingerprint is a hand-crafted substructure descriptor, ECFP4–Tanimoto retrieval serves in our evaluation as

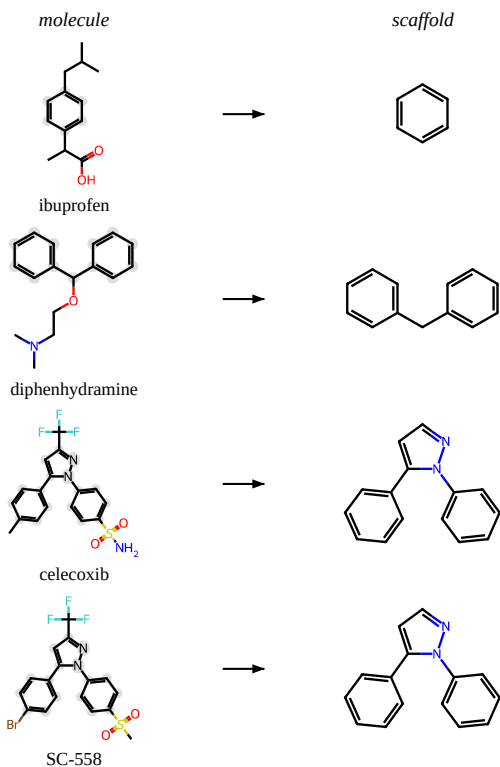


Figure 1: Bemis-Murcko decomposition of four molecules. Acyclic side chains are removed while ring systems and the linkers between them are retained. Celecoxib and SC-558 differ only in their peripheral substituents and therefore map to the same scaffold.

a deterministic reference for chemical similarity rather than as a learned competitor.

2.4 Graph encoder

Our encoder is a Graph Neural Network (GNN), which learns a graph-level representation by iterative message passing over the molecular graph [13, 14, 35]. For consistency with MolCLR we adopt the GIN [45], and its edge-conditioned variant GINE, as the backbone. The encoder maps a molecular graph to a representation $h \in \mathbb{R}^d$ used for downstream tasks and, through a projection head, to an embedding z used by the contrastive objective described next. The message-passing formulation, the GIN and GINE update equations, and the encoder configuration are deferred to Appendix A.1.

2.5 Contrastive learning

Contrastive learning is a self-supervised approach that learns a representation by maximizing agreement between similar (positive) samples while pushing apart dissimilar (negative) ones, without requiring any labels. It can be framed as a dictionary look-up problem [5, 16, 33]: a reference instance is transformed into two augmented views, and the embedding of one view (the query) should match the embedding of the other (its positive key) against a set of negative keys drawn from the other instances in the batch. Positive views

are produced by a stochastic augmentation module $\mathcal{T}$, which transforms an input G into a randomly augmented view $\tilde{G} = t(G)$ with $t \sim \mathcal{T}$; a positive pair is formed by sampling two augmentations $t, t' \sim \mathcal{T}$ of the same molecule. Each view is mapped by the shared encoder into a representation and, through a projection head, into an embedding z on which the loss below operates.

For a minibatch of N molecules, the two augmented views yield $2N$ embeddings. For a positive pair (z_i, z_j) originating from the same molecule, the normalized temperature-scaled cross-entropy loss (NT-Xent) is

$$\ell_{i,j} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} \mathbf{1}_{[k \neq i]} \exp(\text{sim}(z_i, z_k)/\tau)}, \quad (2)$$

where $\text{sim}(u, v) = u^\top v / (\|u\| \|v\|)$ is the cosine similarity and $\tau > 0$ is a temperature hyper-parameter. The total loss is averaged over all $2N$ positive orderings in the batch. Intuitively, the numerator pulls the two views of a molecule together while the denominator pushes that molecule away from all other instances. In practice this is implemented as a cross-entropy over the $2N \times 2N$ similarity matrix in which the target of each row is the index of its positive partner.

A central property of this objective is that the two views produced by $\mathcal{T}$ are pulled towards the same region of the embedding space; the model is therefore encouraged to be invariant to whatever transformations $\mathcal{T}$ applies. This makes the choice of augmentation a decisive design factor: an augmentation encourages invariance to a transformation, which helps when the discarded information is irrelevant to the downstream task but hurts when it is essential. Rather than redesigning the augmentations to resolve this tension, we add a complementary, chemically grounded supervision signal on top of the augmentation-based objective, the scaffold-based positives introduced in this work.

2.6 Graph augmentations

Positive views are produced by perturbing the molecular graph. Adapting graph-contrastive augmentations [15, 30, 36, 47, 48, 53], MolCLR [39] uses three operators: atom masking, bond deletion, and subgraph removal.

- **Atom masking.** A random subset of nodes is selected and each selected atom’s feature is overwritten with the reserved mask token, so the encoder must infer the masked atom from its neighborhood.
- **Bond deletion.** A random subset of edges is removed from the graph, deleting both directed copies of each chosen bond, which forces the model to be robust to missing connectivity.
- **Subgraph removal.** A connected subgraph is selected, typically by sampling a random node and performing a random walk or BFS from it, and every atom and bond within that subgraph is removed from the view, forcing the model to represent the molecule from an incomplete, localized fragment.

While subgraph removal is more widely used and has been reported to yield stronger representations in contrastive learning [39, 48], we opt for the combination of atom masking and bond deletion primarily for computational efficiency: in our experiments, this combination trained roughly twice as fast as subgraph removal,

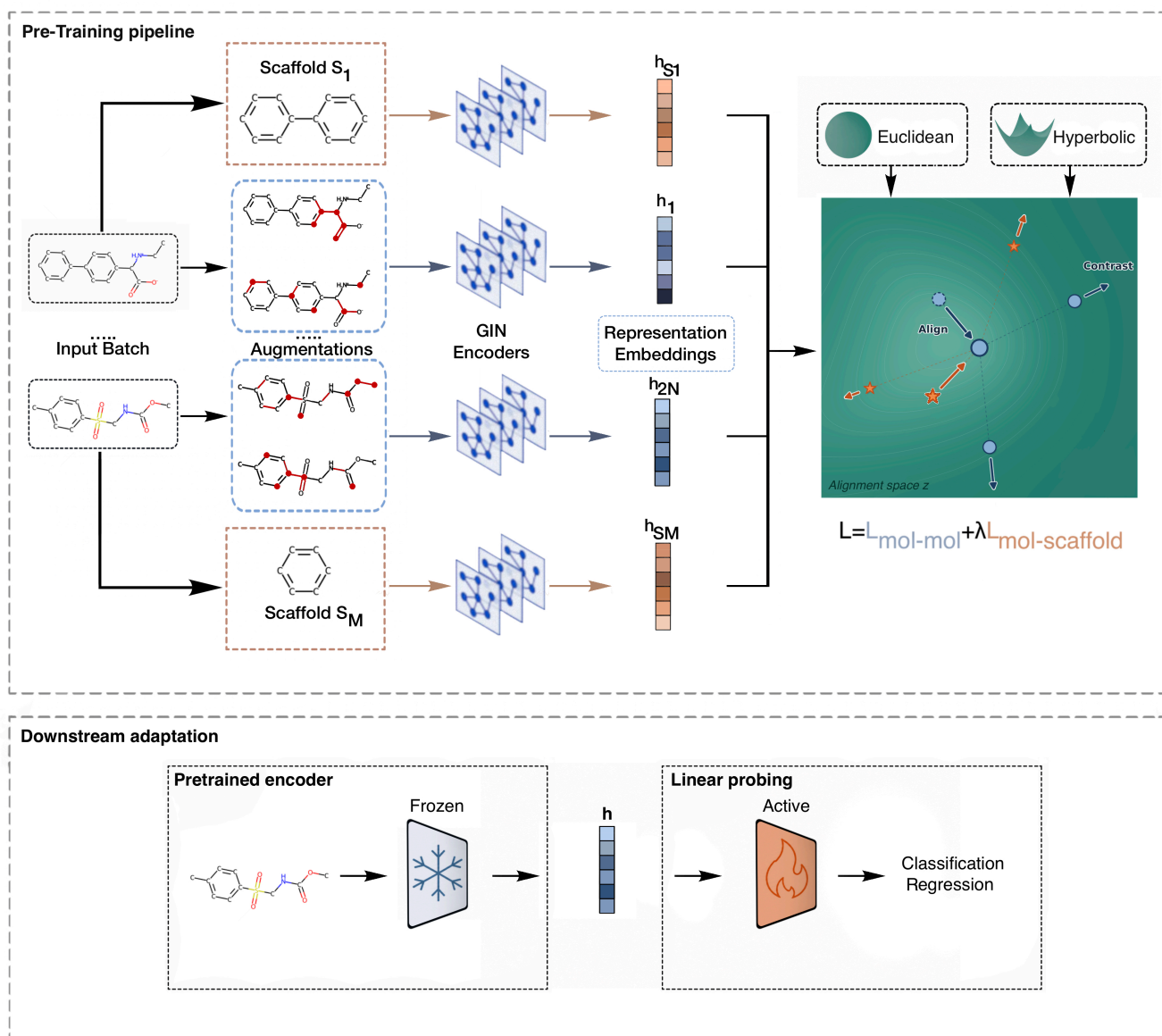


Figure 2: Top: scaffold-aware contrastive pre-training. A mini-batch of N molecules, given as SMILES strings, is converted into molecule graphs. Each molecule is processed along two complementary branches. Two stochastic graph augmentations combining atom masking and bond deletion are applied to every molecule, producing $2N$ correlated views $\tilde{G}_1, \dots, \tilde{G}_{2N}$. In parallel, the Bemis–Murcko scaffold of each molecule is extracted to form a unique scaffold set, providing a positive that shares the molecule’s core ring system. A single shared GIN encoder, followed by a readout, maps every augmented graph and every scaffold to a representation h , and an MLP projection head produces the embeddings z on which the losses act. The contrastive loss maximizes agreement between the two augmented views of the same molecule against the other molecules in the batch, while the scaffold loss aligns each molecule with the embedding of its scaffold. Similarities are measured either by cosine similarity (Euclidean variant) or by the negative Lorentzian geodesic distance (hyperbolic variant). The two terms are combined into the total objective $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{Contrastive}} + \lambda \mathcal{L}_{\text{scaff}}$, where λ balances their contributions. **Bottom: downstream adaptation.** The pre-trained encoder is frozen and the representation h is evaluated by linear probing on classification and regression tasks.

since it avoids the overhead of sampling connected substructures (e.g., via random walks or BFS) at each iteration. Given that our method already introduces a strong structural signal through the scaffold-based positive, we consider this trade-off acceptable, as the

random node-level and edge-level perturbations still provide sufficient local variation to complement it. Both operators are applied to every view at a shared corruption ratio ρ , and each view independently samples which atoms are masked and which bonds are

deleted, so the two views of a molecule differ in which substructures are hidden.

As discussed in the introduction, both operators are chemically blind, and we do not attempt to repair them here: the scaffold supervision we introduce is a complementary signal rather than a fix. The strength ρ navigates a tension: stronger perturbations create a harder contrastive task that yields richer representations [32, 43], but they also increase the risk of severing the graph into disconnected fragments, yielding views that no longer correspond to the same compound. We measure this fragmentation directly and use it to bracket ρ before any pre-training run (Section 3.2).

2.7 Hyperbolic geometry

We summarize only the hyperbolic geometry our method relies on, focusing on the Lorentz model; a self-contained primer on Riemannian and hyperbolic geometry is given in Appendix A.2 [22, 23].

Hyperbolic spaces are Riemannian manifolds of constant negative curvature, in contrast to flat (zero-curvature) Euclidean space and the positively curved sphere. The property we exploit is that the volume of a ball in hyperbolic space grows exponentially with its radius, whereas in Euclidean space it grows polynomially. Because hyperbolic space acts as a continuous analogue of a tree [22, 27], it naturally accommodates hierarchical molecular relationships: generic scaffold cores sit near the origin close to many descendants, while specific molecules are pushed to the periphery, maintaining large distances from unrelated structures. Of the several equivalent models, we adopt the Lorentz (hyperboloid) model [23], which represents the space as a submanifold of $\mathbb{R}^{n+1}$ and enjoys more stable optimization, in line with prior work on hyperbolic representation learning (e.g., MERU) [9].

The Lorentz model. The Lorentz model represents an n -dimensional hyperbolic space as the upper sheet of a two-sheeted hyperboloid in $\mathbb{R}^{n+1}$. Borrowing terminology from special relativity, we split every vector $x \in \mathbb{R}^{n+1}$ into a time component and the remaining space components. In our implementation, the time component is stored first, $x = [x_t, x_s]$ with $x_t \in \mathbb{R}$ and $x_s \in \mathbb{R}^n$, where x_t intuitively measures how generic or specific a molecule’s embedding is, i.e. how close to or far from the origin it lies. The *Lorentzian inner product* is

$$\langle x, y \rangle_{\mathcal{L}} = -x_t y_t + \langle x_s, y_s \rangle, \quad (3)$$

where $\langle \cdot, \cdot \rangle$ is the ordinary Euclidean inner product. For a curvature $-c$ with $c > 0$, the hyperboloid is the set of points

$$\mathcal{L}^n = \{x \in \mathbb{R}^{n+1} : \langle x, x \rangle_{\mathcal{L}} = -\frac{1}{c}, x_t > 0\}, \quad (4)$$

so that every point satisfies $x_t = \sqrt{1/c + \|x_s\|^2}$. The origin of the hyperboloid is $O = [\sqrt{1/c}, \mathbf{0}]$, the point closest to the apex, which we take to represent the most generic concept.

Geodesic distance. A geodesic is the shortest path between two points on the manifold; in the Lorentz model geodesics are the curves cut out by intersecting the hyperboloid with planes through the origin of $\mathbb{R}^{n+1}$. The geodesic (Lorentzian) distance between two points $x, y \in \mathcal{L}^n$ is

$$d_{\mathcal{L}}(x, y) = \frac{1}{\sqrt{c}} \operatorname{arccosh}(-c \langle x, y \rangle_{\mathcal{L}}). \quad (5)$$

Whereas the Euclidean variant of our objective scores a pair by cosine similarity, the hyperbolic variant scores it by the negative geodesic distance $-d_{\mathcal{L}}(x, y)$, so that points lying close together on the hyperboloid are treated as similar.

Lifting onto the manifold. To map a Euclidean vector onto the hyperboloid we use the exponential map at the origin, $\exp_O$. Treating a feature vector $v \in \mathbb{R}^n$ as a tangent vector at O , it is

$$\exp_O(v) = \left[\frac{\cosh(\sqrt{c} \|v\|)}{\sqrt{c}}, \frac{\sinh(\sqrt{c} \|v\|)}{\sqrt{c} \|v\|} v \right]. \quad (6)$$

This origin map is the only one our method uses; the general maps at an arbitrary base point are given in the primer.

3 METHODS

This section outlines our methodology and evaluation protocol. We introduce two separate scaffold-aware mechanisms, implement each across both Euclidean and hyperbolic geometries, and describe the diagnostics and downstream benchmarks used to assess their performance.

3.1 Overview and training objective

As illustrated in Figure 2, each molecule in a mini-batch is processed along two branches. The augmentation branch produces two stochastically augmented views, while the scaffold branch extracts the molecule’s Bemis–Murcko scaffold. Scaffolds are not augmented. Since a scaffold already represents the shared structural core, it is embedded directly, and molecules are aligned to this fixed representation. A single shared GIN encoder maps every augmented graph and every scaffold to a representation, and a projection head produces the embeddings on which the losses are computed. We train with a weighted sum of two terms,

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{contrastive}} + \lambda_s \mathcal{L}_{\text{scaffold}}, \quad (7)$$

where $\mathcal{L}_{\text{contrastive}}$ is the NT-Xent loss of Equation (2) between the two augmented views, $\mathcal{L}_{\text{scaffold}}$ aligns molecules with their scaffolds, and $\lambda_s \geq 0$ (written λ in Figure 2) balances the two. Equation (7) defines our scaffold-alignment model; the scaffold-masked variant, described first below, instead modifies $\mathcal{L}_{\text{contrastive}}$. Both objectives exist in a Euclidean form (cosine similarity) and a hyperbolic form (negative Lorentzian geodesic distance).

All models are pre-trained on the same one-million-molecule subset of the PubChem-10M corpus, of which 25% (250,000 molecules) is held out as a validation split, so every encoder is trained on the same 750,000 molecules. The encoder is the five-layer GINE backbone producing $d = 512$ -dimensional representations; the corpus, the offline graph pre-computation, the encoder configuration, and the optimization hyper-parameters are all detailed in Appendix A.

3.2 Selecting the augmentation strength

The corruption ratio ρ sets the difficulty of the contrastive objective, and is normally inherited from earlier graph-contrastive implementations rather than justified on the molecules at hand. We fix it by measurement before any pre-training run. We apply three operators to 3,000 molecules from the validation split at $\rho \in \{5, 10, \dots, 30\}\%$: the operator we train with (atom masking combined with bond deletion, labelled “node + edge deletion” in Figures 3 and 4; atom

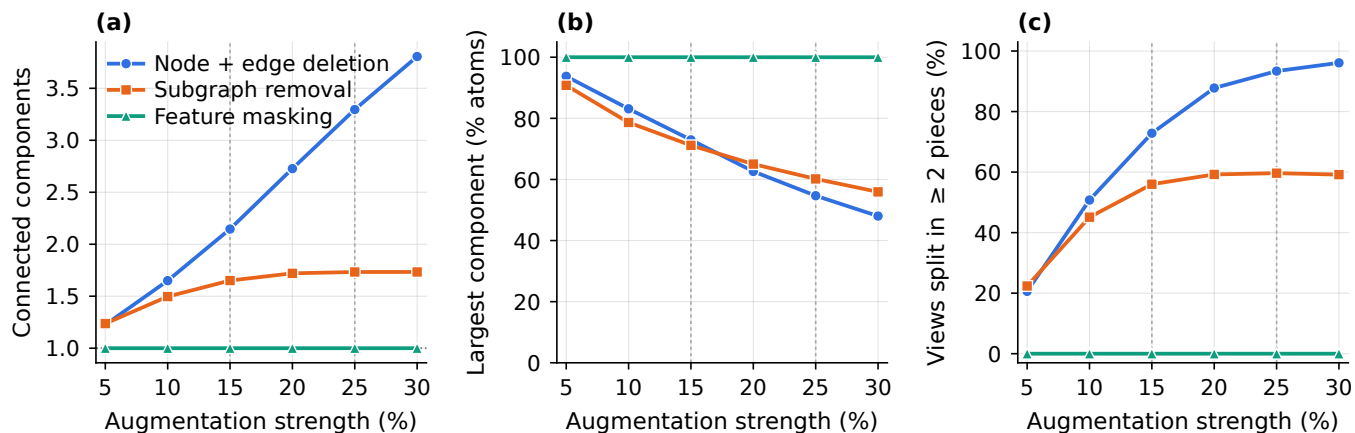


Figure 3: Graph fragmentation against augmentation strength, over 324,000 views of 3,000 validation molecules. (a) Mean connected components per view. (b) Share of atoms in the largest component. (c) Share of views split into two or more pieces. Feature masking overwrites atom and bond features without deleting an edge and is flat by construction, serving as a control. Dashed lines mark the two operating points, $\rho = 15\%$ and $\rho = 25\%$.

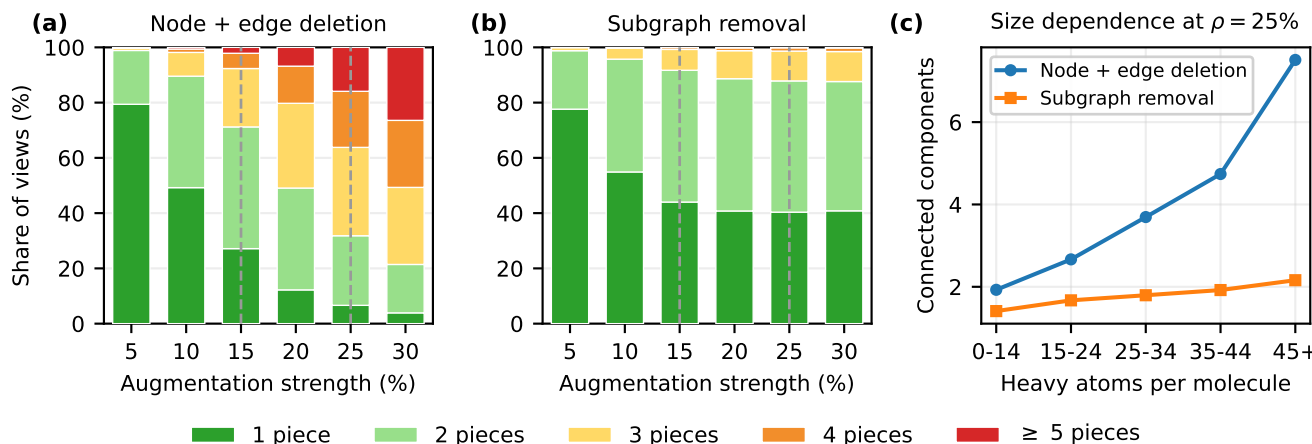


Figure 4: (a, b) Distribution of piece counts per view rather than the mean alone: node + edge deletion shifts continuously towards heavy fragmentation, whereas subgraph removal plateaus from $\rho = 20\%$ onward. (c) At $\rho = 25\%$, random deletion breaks large molecules far more severely than small ones, while subgraph removal is close to size-invariant.

masking leaves connectivity untouched, so all fragmentation arises from the deleted bonds), subgraph removal for comparison, and feature masking as a control.

We draw three independent view pairs per molecule and measure both views separately, for 324,000 augmented views in total, and compute connectivity by union-find over the edge index. A *component* here is a connected piece of at least two bonded atoms: an atom whose bonds have all been deleted stays in the tensor as an isolated node but is not counted as a piece of its own. Every molecule in the sample is a single component before augmentation. Figure 3 shows why strength matters here in a way it does not for images or social graphs. Deleting bonds eventually disconnects the molecule, and the two operators do so differently. Random deletion clips terminal atoms and side chains anywhere in the graph, so pieces accumulate without bound: from 1.2 components at 5% to 3.8 at 30%. Subgraph removal excises one contiguous region, which can only cut the remainder in two, so it plateaus near 1.7 beyond

$\rho = 20\%$. Fragmentation is thus not a side effect of our operator but its dominant behaviour at high ρ , and the largest surviving piece shrinks accordingly.

This brackets a usable range. Below 15% the task is close to trivial: at 5% four-fifths of views are still in one piece, so the two views differ in which atoms were masked and which bonds were deleted but not in connectivity. At 15% most views are split, yet 73% of atoms remain in a single component, so the view is harder without ceasing to be mostly the molecule. That second condition fails above 25%, the last strength at which the largest component still holds a majority of the atoms (55%, against 48% at 30%).

This is our answer to RQ1: within the tested range, no strength keeps most views in one piece while also posing a non-trivial contrastive objective, but 15–25% keeps the majority of the atoms connected. Connectivity is necessary but not sufficient for a useful view, so we treat ρ as a pre-training choice rather than a hyperparameter tuned downstream: we pre-train a separate encoder from

scratch at $\rho = 15\%$ and $\rho = 25\%$, and evaluate each independently with the molecule-molecule retrieval probe and by linear probing on the downstream tasks. Every reported result can thus be attributed to a single augmentation strength. Appendix A.10 reports pre-training runs at additional strengths.

Deletion versus masking. The control in Figure 3 is not only a measurement baseline, it is also a second operator we pre-train. Since fragmentation follows entirely from deleting edges, we train a parallel family of models in which a bond’s type is replaced by a reserved mask token while the edge itself is retained, exactly as atoms are masked. Every view is then a single connected graph carrying the molecule’s full heavy-atom topology, which removes fragmentation as a failure mode. It does not restore chemical identity, since an atom or bond whose type is masked remains undefined. We therefore retain deletion for the main experiments and report the comparison in Appendix A.11.

3.3 Scaffold-masked contrastive loss

The contrastive setup presented maps a graph into a representation $h = f(\tilde{G}) \in \mathbb{R}^d$, and a projection head g maps h into the embedding $z = g(h)$ on which the contrastive loss operates. The projection head outputs unnormalized vectors. Normalization is part of the similarity, not the head: the Euclidean variant normalizes z to unit L_2 norm so that the inner product becomes the cosine similarity, whereas the hyperbolic variant instead lifts z onto the manifold. Under the NT-Xent loss of Equation (2), the objective used by MolCLR, each view’s only positive is the other view of the same molecule and every other embedding in the batch is a negative.

A natural hypothesis concerns the way negatives are chosen. Because negatives are simply “all other molecules in the batch”, two distinct molecules that happen to share the same Bemis-Murcko scaffold are pushed apart even though they are structurally related, regardless of how (dis)similar they otherwise are. One might regard such pairs as false negatives that the objective wrongly separates [38].

Under this hypothesis we mask same-scaffold negatives. For each molecule we look up its scaffold (as a canonical SMILES string) and assign an integer identifier; molecules with no well-defined scaffold (the acyclic compounds) each receive a distinct identifier, so that two scaffold-less molecules are never treated as sharing a scaffold and remain valid negatives for each other. Two embeddings whose molecules share a scaffold identifier are then excluded from each other’s set of negatives. Concretely, if $S(k)$ denotes the scaffold identifier of the molecule behind embedding k , the masked loss for a positive pair (z_i, z_j) is

$$\ell_{i,j}^{\text{mask}} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} m_{ik} \exp(\text{sim}(z_i, z_k)/\tau)}, \quad (8)$$

$$m_{ik} = \begin{cases} 0 & k = i \text{ or } (S(k) = S(i) \text{ and } k \neq j), \\ 1 & \text{otherwise,} \end{cases} \quad (9)$$

so that the true positive z_j is always retained while same-scaffold negatives are removed (in practice by masking those entries of the

similarity matrix before the softmax). This masked objective is a drop-in replacement for the MolCLR contrastive loss.

3.4 Scaffold-alignment loss

Masking same-scaffold negatives only stops the objective from separating related molecules; it supplies no positive signal that organizes molecules around their scaffolds. We therefore consider a second, independent mechanism. Rather than altering the negatives, we leave the contrastive loss of Equation (2) untouched and add a scaffold-alignment term to it, treating each molecule’s own scaffold as a chemically grounded positive alongside its augmented views. The two mechanisms are alternatives, evaluated separately rather than combined.

For each batch we collect the set of $U \leq N$ unique scaffolds present and embed them (without augmentation) with the same shared encoder, yielding scaffold embeddings $\{z_{S_1}, \dots, z_{S_U}\}$; this adds only one encoder forward pass per batch. Let a molecule with a valid scaffold have augmented-view embeddings $z_{G(i)}$, $z_{G(j)}$ and let $S(G)$ be the index of its own scaffold in the unique set. We contrast each view against all unique scaffolds, treating the molecule’s own scaffold as the target:

$$\mathcal{L}_{\text{scaffold}} = -\frac{1}{2} \mathbb{E}_G \sum_{v \in \{i,j\}} \log \frac{\exp(\text{sim}(z_{G(v)}, z_{S(G)})/\tau_s)}{\sum_{u=1}^U \exp(\text{sim}(z_{G(v)}, z_{S_u})/\tau_s)}, \quad (10)$$

where τ_s is a (possibly separate) temperature, the expectation runs over the molecules in the batch that have a valid scaffold, and the inner sum covers the molecule’s two augmented views. Both views are pulled towards the same scaffold embedding and pushed away from all other scaffolds. Because the candidate pool is the set of unique scaffolds rather than the molecules, the denominator is not dominated by the multiplicity of common cores: a frequent scaffold still appears once in the numerator for each molecule that owns it, but it contributes only a single negative, which limits the class-imbalance bias towards common cores. Molecules without a valid scaffold (the acyclic compounds) are excluded from this term, and when a batch contains no scaffold the loss contributes zero gradient.

3.5 Hyperbolic formulation in the Lorentz model

Both objectives above are written through a generic similarity $\text{sim}(\cdot, \cdot)$. The Euclidean variant takes it to be cosine similarity; the hyperbolic variant instead uses the negative Lorentzian geodesic distance, Equation (5).

We do not L_2 -normalize the embeddings: doing so would force every point onto a single shell at fixed distance from the origin and remove the radial degree of freedom that may encode hierarchy. Instead, for the hyperbolic loss, we scale each raw encoder output $z \in \mathbb{R}^d$ by a single learnable feature scale α and lift it onto the hyperboloid with the origin exponential map $\hat{z} = \exp_O(\alpha z)$. A single α and a single curvature c are shared by both losses, so that molecules and scaffolds are mapped onto the manifold identically. The hyperbolic losses are then obtained from Equation (8) and Equation (10) by substituting

$$\text{sim}_{\mathcal{L}}(z, z') = -d_{\mathcal{L}}(\exp_O(\alpha z), \exp_O(\alpha z')) \quad (11)$$

for the cosine similarity. By default we fix the curvature at $c = 1$, a single shared scalar governing the geometry of the whole space, and study its effect by sweeping $c \in \{0.5, 1, 2\}$ in Appendix A.12; a larger c gives a more strongly curved space and a stronger hierarchical bias. The parametrization and initialization of α , and the optional joint learning of c , are detailed in Appendix A.3.

Bounded and unbounded Lorentz formulations. We consider two implementations of the Lorentz objective, differing only in how the scaled projection is mapped onto the hyperboloid. The *unbounded* formulation follows the mapping above, lifting $v = \alpha z$ directly with $\exp_O$. The *bounded* formulation instead replaces v before the exponential map by

$$\tilde{v} = r_{\max} \tanh\left(\frac{\|v\|}{r_{\max}}\right) \frac{v}{\|v\|}, \quad (12)$$

with the continuous value at $v = 0$. Since $\tanh$ has unit slope at the origin, $\tilde{v} \rightarrow v$ as $\|v\| \rightarrow 0$ and the two formulations coincide in the small-norm regime; only the *length* of v is bounded, to $r_{\max}$, while its direction is left unchanged, so the angular content of the embedding is untouched. We set $r_{\max} = 5$ and, unlike the unbounded runs, hold α fixed at its initialization.

Why the bound is necessary. The exponential map of Equation (6) must clamp the argument of $\sinh$ to avoid single-precision overflow. Following existing implementations [9], we clamp it at $\operatorname{arcsinh}(2^{15}) \approx 11.09$. In every unbounded run this clamp became active. Because z is not L_2 -normalized and α is learned jointly with the encoder, $\sqrt{c} \|\alpha z\|$ crossed the threshold within the first ten epochs, after which $\exp_O$ returned the same time component and the same space-vector magnitude for every input. Each embedding then sits at the identical geodesic radius $\operatorname{arcsinh}(2^{15})/\sqrt{c}$ from the origin, and only its direction distinguishes it from any other.

This failure is invisible to the loss and obvious in the hierarchy diagnostics introduced in the next section, which drop to exactly zero once the clamp engages (Appendix A.4 shows the curves and explains the signature). What the unbounded objective then optimizes is easy to characterize: on a shell of fixed hyperbolic radius the geodesic distance between two points is a monotone function of the angle between them, so the loss reduces to a rescaled angular (cosine-like) objective. It still trains, and still separates positives from negatives, but the radial coordinate through which a hyperbolic space expresses hierarchy carries no information.

The bounded formulation removes the failure by construction, since $\|\tilde{v}\| \leq r_{\max} < \operatorname{arcsinh}(2^{15})$ makes the clamp unreachable at any encoder norm. We report both formulations throughout. The unbounded runs remain a valid Lorentz objective and are not discarded, they are simply one whose radial degree of freedom has collapsed, so any claim about molecule-scaffold hierarchy can only be read from the bounded runs. Note that the two formulations differ in two respects, the $\tanh$ bound and whether α is learned, so differences between them cannot be attributed to the bound alone.

3.6 Evaluation and monitoring

Because the pre-training objective is self-supervised, we monitor several label-free diagnostics on the held-out validation split.

Contrastive accuracy. The most direct diagnostic asks whether the model can identify the matching view of a molecule among all other views in the batch. For the $2N$ validation embeddings we compute the pairwise (negative geodesic, or cosine) similarity, mask the diagonal, and for each view take its highest-scoring partner. The contrastive accuracy is the fraction of views whose top-ranked partner is their true positive. A random predictor scores $1/(2N - 1)$, so values well above indicate that the model has learned to discriminate molecules.

Hyperbolic diagnostics. To monitor how the latent space is shaping, we additionally track three quantities once per validation epoch, the first two measured by distance from the origin O (the assumed root of the hierarchy). In this thesis they serve to diagnose the failure of the unbounded formulation (Appendix A.4); all substantive comparisons between geometries are made in the backbone space h with the retrieval probe below:

- Norm separation: the mean distance-from-origin of molecules minus that of scaffolds. A positive value means molecules sit further from the origin than their scaffolds.
- Entailment rate: the fraction of (molecule, scaffold) pairs in which the scaffold is closer to the origin than the molecule. This should trend towards 1 as the hierarchy emerges.
- Similarity gap: the mean similarity of positive pairs minus that of negative pairs, a continuous counterpart of the contrastive accuracy above, and the Lorentzian analogue of the positive-negative similarity gap used to analyze embeddings in Euclidean contrastive frameworks such as SimCLR [5]. A positive value confirms that positives are embedded closer than negatives.

Molecule-molecule retrieval. Our main structural probe asks what a molecule’s nearest neighbours have in common with it. We fix a pool of 30,000 molecules from the held-out validation split and, for each query in it, rank every remaining candidate by similarity in the frozen backbone representation h . The pool is scaffold-balanced by construction: a uniformly random sample would consist almost entirely of molecules that are the sole member of their scaffold class, leaving nothing to retrieve, so 60% of the pool is drawn from Bemis-Murcko scaffold classes with 5–10 members, capped at ten molecules per class so that no single core dominates, and the remainder are singleton-scaffold molecules that act as distractors but are never used as queries. The measure is purely relative and therefore not misled by representation collapse, the failure mode in which the encoder maps all inputs to nearly the same vector: a collapsed encoder assigns high similarity to every pair, which a rank-based probe registers as a near-random ranking rather than as a high score.

Three settings are reported. In the first, a retrieved neighbour counts as correct when it carries the query’s exact BM scaffold, summarized as mean average precision over the top ten (MAP@10). Exact scaffold identity is what the alignment loss is trained on, so this is a check that the intended relation was learned rather than

a finding in itself. The second is stricter: every candidate sharing the query’s exact scaffold is deleted from the pool before ranking, and the surviving top ten are scored on a graded scale by ECFP4–Tanimoto similarity to the query, measured both between their BM scaffolds (BM-Tan@10) and between the complete molecules (Mol-Tan@10). We also report normalized discounted cumulative gain (nDCG@10), which discounts each retrieved candidate’s similarity by its rank and normalizes by the best ordering achievable under the same exclusion. A model can therefore earn no credit for reproducing the relation it was trained on. Direct ECFP4–Tanimoto retrieval over the same pool is reported alongside as a deterministic chemical-similarity reference; being a hand-crafted substructure descriptor it is a reference rather than a competitor, and on the whole-molecule nDCG column it is the ceiling by construction.

The third, controlled setting removes the dependence on how the pool was built. Each query is presented with exactly one molecule sharing its scaffold and k distractors that do not, so chance accuracy is exactly $1/(k+1)$. Distractors are drawn either uniformly from the pool (*random* distractors, which a model can beat by encoding any whole-molecule resemblance) or from the 200 molecules most similar to the query by full-molecule ECFP4–Tanimoto that nonetheless carry a different core (*hard* distractors, for which whole-molecule fingerprint similarity no longer singles out the correct answer, so a model must encode something beyond it, in particular the scaffold). The drop from random to hard distractors is itself diagnostic: a model whose neighbourhood is organized by scaffold loses less, whereas one that only encodes whole-molecule similarity falls towards chance. We use $k = 9$ distractors; Appendix A.14 describes the protocol in full. Two model-free references are scored alongside: the ECFP4 retriever itself and an untrained encoder with random weights, which shows how much of any score is due to the GIN architecture and mean pooling rather than to pre-training. The candidate sets are identical for every model, making the comparison paired. Because queries that share a scaffold see almost the same candidates and the same correct answers, they are not independent observations, and all reported intervals are 95% bootstrap intervals that resample whole scaffold classes rather than individual queries.

Molecule–scaffold ranking. As a secondary diagnostic we record, for each molecule, the rank of its own scaffold among all unique scaffolds in the batch, summarized by the Mean Reciprocal Rank, the Hits@ k rates for $k \in \{1, 5, 10\}$ and the median and mean rank. This is the relation the alignment loss states directly, so we report it as a sanity check rather than as evidence of emergent structure. The protocol and the results are given in Appendix A.15.

Visualizing the embedding space. The retrieval probe quantifies how the space is organized but does not show it. We therefore project the embeddings of a held-out sample of validation molecules to two dimensions with t-SNE [34] for qualitative inspection, reading only local structure from the result: nearby points reflect genuine neighborhoods, whereas absolute positions and distances between far-apart clusters are not meaningful.

3.7 Downstream evaluation protocol

We evaluate the pre-trained encoders on seven datasets from MoleculeNet [17, 42], under scaffold splitting and with the encoder frozen:

only a linear head is trained on top of the pooled representation, so that differences in performance reflect the representation rather than the fine-tuning. Throughout, “MolCLR” denotes our own re-training of the MolCLR objective on the same subset and encoder, a controlled baseline at an equal data and architecture budget rather than the published model. The head architecture, optimizer, split ratios, and seed protocol are given in Appendix A.

The panel comprises three classification tasks (BBBP, BACE, ClinTox), evaluated with ROC-AUC (higher is better), and four regression tasks (ESOL, FreeSolv, Lipo, QM7), evaluated with RMSE (lower is better) except QM7, which uses MAE. These benchmarks are not interchangeable, and the difference matters here because our scaffold prior is a statement about the ring system of a molecule and is therefore more relevant to some tasks than to others. By the kind of chemical information a model must expose to solve them, they fall into four groups: whole-molecule physicochemical properties (BBBP, ESOL, Lipo and, in a purer form, FreeSolv), target binding driven by a shared core (BACE), local structural alerts (ClinTox), and a composition-driven quantum property (QM7), for which a two-dimensional core carries little information. The individual datasets, their sizes, and the chemistry behind each label are described in Appendix A.8.

4 RESULTS

4.1 Molecule–molecule retrieval

Molecule–scaffold alignment, the relation the scaffold loss states directly, is confirmed for two representative models in Appendix A.15. The question here is a stricter one: whether the space places a molecule near *other molecules* built on a related core. No term of the MolCLR or scaffold-alignment objective compares two molecule embeddings under a scaffold-derived label; molecules sharing an exact scaffold are, however, aligned to the same scaffold embedding, so only retrieval beyond the exact scaffold can be called emergent. We use the scaffold-balanced pool of 30,000 held-out molecules, ranking by cosine similarity on the backbone representation h , which is Euclidean in all twelve runs; the hyperbolic runs are therefore evaluated on the same Euclidean readout as the others, and what the probe measures is the effect of the Lorentz *objective* on that readout. Every contrast quoted below is a paired difference between two encoders on identical queries, with a 95% interval from a bootstrap that resamples whole scaffold classes (2,000 resamples); an untrained encoder with random weights is scored alongside as a reference for what the architecture alone provides.

Exact-scaffold retrieval. As expected, scaffold supervision strongly increases retrieval of molecules sharing the query’s exact BM scaffold, by +0.030 to +0.198 MAP@10 depending on the geometry and augmentation strength, with every interval excluding zero (Appendix Table 5). For scale, the untrained encoder reaches 0.047 MAP@10, the Euclidean MolCLR baselines 0.088–0.099, and the best scaffold model 0.263, above the 0.242 of direct ECFP4 retrieval. Because two molecules with the same scaffold are pulled towards the same scaffold embedding during training, this follows closely from the trained relation and is a check rather than a finding.

Neighbourhoods beyond the exact scaffold. We next exclude all molecules sharing the query’s exact scaffold. Under this stricter

Table 1: Molecular retrieval in the frozen embedding space h , after excluding molecules with the query’s exact BM scaffold. BM-Tan@10 compares the retrieved scaffolds, while Mol-Tan@10 compares the complete molecules. One pre-trained encoder per cell. Direct ECFP4 retrieval is included as a deterministic chemical-similarity reference and an untrained encoder with random weights as an architecture-only reference. Boldface marks the best learned model in each column; the ECFP4 reference is excluded.

Model	BM-Tan@10 $\uparrow$		Mol-Tan@10 $\uparrow$	
	25%	15%	25%	15%
MolCLR (Euclidean)	0.3213	0.3276	0.2314	0.2346
Scaffold (Euclidean)	0.3386	0.3715	0.2391	0.2461
MolCLR (Lorentz-unbounded)	0.3029	0.2932	0.2330	0.2140
Scaffold (Lorentz-unbounded)	0.3522	0.3901	0.2471	0.2471
MolCLR (Lorentz-bounded)	0.3077	0.3030	0.2328	0.2185
Scaffold (Lorentz-bounded)	0.3401	0.3937	0.2358	0.2479
Untrained encoder	0.2720	0.2720	0.1962	0.1962
Direct ECFP4 retrieval	0.3991	0.3991	0.3665	0.3665

setting, scaffold-supervised models retrieve molecules with more similar, but nonidentical, BM scaffolds than their corresponding MolCLR models (Table 1). The same retrieved neighbours are also more similar when the complete molecules are compared using ECFP4. This effect is positive in all six geometry $\times$ augmentation cells and every interval excludes zero: the smallest BM-Tan@10 gain is +0.017 [+0.016, +0.018] (Euclidean, 25%) and the largest +0.097 [+0.094, +0.100] (Lorentz-unbounded, 15%). The untrained encoder scores 0.272 BM-Tan@10, so about two thirds of the baselines’ distance from chance (0.155) is already present in the untrained encoder, before any pre-training. Scaffold supervision therefore changes molecular neighbourhoods beyond simply grouping molecules with the exact scaffold used during training. The change is concentrated in the core rather than being a uniform increase in chemical similarity: in every cell the gain is at least twice as large on the retrieved scaffolds as on the complete molecules (+0.044 against +0.012 under Euclidean training at 15%, and the same ordering in all six cells).

The geometry helps only in combination with the scaffold signal. The gain from scaffold supervision is larger under both Lorentz variants than under Euclidean training, at both augmentation strengths: the interaction (Scaffold_L – MolCLR_L) – (Scaffold_E – MolCLR_E) is positive and excludes zero in all four cells (+0.053 [+0.051, +0.055] and +0.047 [+0.045, +0.048] BM-Tan@10 at 15%, +0.032 [+0.031, +0.033] and +0.015 [+0.014, +0.017] at 25%). It is, however, driven more by the matched MolCLR baseline than by the scaffold model: hyperbolic MolCLR is the weakest learned model on BM-Tan@10 in Table 1 and falls 0.014–0.034 below Euclidean MolCLR, and in every cell that drop accounts for more than half of the interaction. The scaffold-supervised Lorentz models are nevertheless the two best learned models at 15%, ahead of the Euclidean scaffold model by 0.019–0.022 BM-Tan@10; at 25% the margin shrinks to 0.014 [0.012, 0.015] (unbounded) and 0.002 [0.000, 0.003] (bounded), the latter statistically distinguishable from zero but of no practical size. At 25% the same bounded model is in fact slightly *below* the Euclidean scaffold model on whole-molecule similarity (–0.003 Mol-Tan@10, interval excluding zero), so at the stronger augmentation the geometry buys a marginally more scaffold-specific neighbourhood at the cost of a slightly less chemically similar one. The Lorentz objective therefore sharpens scaffold organization once a signal names the relation to be organized, most clearly at the milder augmentation; on its own it yields less scaffold structure than Euclidean

training. Bounded and unbounded training are close throughout; since the two also differ in whether α is learned, we read this only as the bound not being a source of gain in retrieval.

The scaffold signal prefers the milder augmentation. Every scaffold-supervised model retrieves better at 15% than at 25% augmentation (–0.033, –0.038 and –0.054 BM-Tan@10 for the Euclidean, unbounded and bounded variants, all intervals excluding zero), whereas the MolCLR baselines are nearly unaffected (–0.006 Euclidean) and move in the opposite direction under Lorentz training (+0.010 and +0.005). This is consistent with what the alignment term does: its target is the un-augmented scaffold, so the stronger perturbation degrades exactly the views that are asked to match an intact core.

Distance to a deterministic fingerprint. The best scaffold-supervised model reaches 0.3937 BM-Tan@10, against 0.3991 for direct ECFP4–Tanimoto retrieval over the same pool, while remaining far below it on whole-molecule similarity (0.2479 against 0.3665). The learned space is therefore not a rediscovery of the fingerprint: it selects neighbours that are markedly less similar overall yet nearly as similar in their cores, which is precisely the selectivity the scaffold objective was designed to induce. The controlled setting makes the same point without depending on how the pool was balanced (Table 6). With nine random distractors every learned model scores well above the 10% chance rate, and the untrained encoder already reaches 61.0%, so this column largely reflects whole-molecule resemblance. Replacing the distractors by the query’s own ECFP4 nearest neighbours that do *not* share its core is what separates the models: the MolCLR baselines fall to 24–35%, within a few points of the untrained encoder’s 28.6% and in two cases below it, whereas the scaffold-supervised models at 15% augmentation retain 48–60%, the best (Lorentz-bounded) succeeding 60.5% of the time against 31.9% for its matched baseline (+0.286 [+0.277, +0.295]). The ECFP4 reference reaches 38.8%, though the hard distractors are by construction adversarial to it and that figure should be read with care. Appendix A.14 describes the protocol in full and reports both kinds of distractor for every encoder.

4.2 Effect of scaffold-aware masking

We first isolate the contribution of scaffold-aware masking, which removes molecules that share a scaffold from each other’s set of negatives. We compare the standard MolCLR contrastive loss against

the masked variant, both trained with 25% augmentation and otherwise identical settings, across all seven benchmarks. Table 2 reports the results.

Scaffold-aware masking performs worse than the unmodified baseline on six of the seven benchmarks: clearly on BBBP, FreeSolv and QM7, and by margins within seed variance on BACE, ESOL and Lipo; the exception is ClinTox, where the masked variant is ahead by less than one standard deviation. We offer one hypothesis, which these runs do not test directly. The NT-Xent objective draws much of its signal from hard negatives [33], and molecules that share a scaffold but differ in their substituents are exactly such negatives: chemically similar yet not identical, so that contrasting against them forces the encoder to represent the substituent-level differences between members of one family. Masking removes precisely these pairs. The corpus statistics of Appendix A.6 indicate how many: the most frequent cores are small generic ring systems (benzene alone carries 7.2% of molecules), so a batch of 512 contains dozens of molecules sharing such a core, and masking them out removes a sizeable set of negatives that are only loosely related. Under this reading the masked objective loses informative hard negatives without gaining a positive signal in exchange, and a small net cost follows. This motivates our second design, which keeps all negatives in place and instead injects the scaffold as an explicit positive.

4.3 Scaffold alignment and hyperbolic geometry

We now evaluate the scaffold-alignment loss in a design that crosses the learning signal (the plain MolCLR objective vs. the scaffold-alignment objective) with three latent-geometry formulations (Euclidean, Lorentz-unbounded, Lorentz-bounded). This gives six models that share the same encoder and pre-training budget; the two hyperbolic MolCLR baselines let us separate the effect of the geometry from that of the scaffold supervision. All six are evaluated at both 15% and 25% augmentation; Table 3 collects the results, with benchmarks as columns and models as rows.

Effect of the scaffold signal. Holding the geometry Euclidean, the scaffold-alignment loss is the more reliable of the two learning signals: it matches or improves on the MolCLR baseline on most benchmarks at both augmentation strengths and is the best of all twelve runs on ESOL, Lipo, and QM7. This is the opposite of the masking result above: adding the scaffold as an explicit positive helps, whereas removing same-scaffold negatives did not. Together these two experiments answer RQ2. The pattern is broadly coherent with what each benchmark measures. The most consistent gain is on BACE, the one benchmark whose label is determined by a shared binding core: the scaffold model improves on its matched baseline in all six geometry $\times$ augmentation cells, by +0.013 to +0.099 ROC-AUC, although the smallest of these (Euclidean, 25%) is within seed variance. Under Euclidean training ESOL, where the ring system contributes a coarse but real part of the signal, improves at both strengths (−0.075 and −0.185 RMSE), while QM7, the composition-driven quantum task for which a two-dimensional core carries little information, improves by about one MAE point, comparable to its seed variance. The bounded hyperbolic pair at 25% is the one cell in which the scaffold model improves on its matched baseline on *all seven* benchmarks, by +0.020 ROC-AUC on BBBP, +0.061 on BACE and +0.023 on ClinTox, and by 0.205 RMSE on ESOL, 0.170

on FreeSolv, 0.017 on Lipo and 6.6 MAE on QM7. Two caveats apply to that sweep: the bounded MolCLR baseline is the weakest baseline in the panel, and two of its seven cells are the high-variance ones noted below, so the sweep is best read as one more positive cell rather than as a stronger result than the Euclidean pair. Across the six cells the scaffold signal beats its matched baseline, on point estimates, in 31 of the 42 model–benchmark comparisons.

Effect of the geometry. The four hyperbolic baselines isolate the geometry alone, and they disagree with each other in an informative way. At 25% augmentation the *unbounded* hyperbolic MolCLR baseline improves on its Euclidean counterpart on several tasks (BBBP, BACE, ESOL, FreeSolv) while regressing on others (ClinTox, Lipo, QM7), which would suggest that part of the hyperbolic benefit is geometric and does not require the scaffold term. The *bounded* baseline does not reproduce this: at the same strength it beats Euclidean MolCLR on only two of the seven benchmarks, and at 15% on two as well, matching the unbounded baseline’s behaviour at 15%, where it too is worse on almost every task. The unbounded formulation is the one whose radial coordinate saturates within ten epochs (Appendix A.4), so that what it optimizes is in effect a rescaled angular objective on a shell of fixed radius. The apparent geometric gain at 25% is therefore a property of that parameterization rather than evidence for curvature as such. We note that the two formulations also differ in whether α is learned, so this attribution rests on the saturation diagnosis rather than on a single-variable ablation.

Combining both ingredients gives the best single values in the panel on FreeSolv (2.503 at 15%, 2.787 at 25%) and BACE (0.742 at 25%), all under the unbounded formulation, but at both strengths that model is also the weakest of the three scaffold models on QM7 and ESOL. This answers the second half of RQ3: the geometry is not a free improvement, its benefit is conditional on the augmentation strength, and, as the two formulations show, on how the projection onto the manifold is implemented. The first half, whether the Lorentz objective yields a more scaffold-organized representation, is answered affirmatively but conditionally by the retrieval probe: only in combination with the scaffold signal, and most clearly at 15%.

Bounded versus unbounded. The two formulations differ more downstream than they do in retrieval, where they were close throughout. The bounded scaffold model at 25% is the more evenly balanced of the two hyperbolic scaffold models: it has the best ClinTox point estimate in the panel (0.580, within seed variance of the other two scaffold models at that strength) and is the only scaffold-supervised model to reach 0.65 on BBBP, where the unbounded scaffold model is the weakest of all six at 0.611. The unbounded model is the spikier of the two, winning BACE by a wide margin (0.742 against 0.654) and losing clearly on BBBP, and on QM7 relative to the Euclidean models. Bounding also reduces the hyperbolic QM7 penalty in three of the four hyperbolic cells (most visibly 104.2 $\rightarrow$ 91.5 for MolCLR and 94.8 $\rightarrow$ 88.4 for the scaffold model at 15%), though every hyperbolic model remains far behind the Euclidean $\sim$ 76, so neither formulation makes the geometry competitive on a composition-driven target. Four cells carry visible seed variance and should not be over-read: bounded scaffold ESOL at 25% ($\pm$ 0.122), bounded

Table 2: Effect of scaffold-aware negative masking: our MolCLR baseline versus the masked contrastive loss, both with 25% augmentation. Each column is a benchmark; classification is ROC-AUC ($\uparrow$), regression is RMSE ($\downarrow$) except QM7 (MAE, $\downarrow$). Values are mean $\pm$ std over three seeds; the better value per column is in bold.

Model	BBBP ROC-AUC $\uparrow$	BACE ROC-AUC $\uparrow$	ClinTox ROC-AUC $\uparrow$	ESOL RMSE $\downarrow$	FreeSolv RMSE $\downarrow$	Lipo RMSE $\downarrow$	QM7 MAE $\downarrow$
MolCLR (25%)	0.651 $\pm$ 0.002	0.606 $\pm$ 0.022	0.521 $\pm$ 0.033	1.652 $\pm$ 0.002	3.374 $\pm$ 0.044	1.009 $\pm$ 0.002	76.148 $\pm$ 0.095
Masked (25%)	0.627 $\pm$ 0.002	0.556 $\pm$ 0.069	0.537 $\pm$ 0.028	1.661 $\pm$ 0.008	3.581 $\pm$ 0.028	1.017 $\pm$ 0.006	79.206 $\pm$ 0.598

Table 3: Effect of the scaffold signal and of the latent geometry: learning signal (MolCLR vs. scaffold alignment) crossed with three geometry formulations (Euclidean, Lorentz-unbounded, Lorentz-bounded), each at 15% and 25% augmentation. Each column is a benchmark; classification is ROC-AUC ($\uparrow$), regression is RMSE ($\downarrow$) except QM7 (MAE, $\downarrow$). Values are mean $\pm$ std over three seeds; the best value per column (across all rows) is in bold.

Model	BBBP ROC-AUC $\uparrow$	BACE ROC-AUC $\uparrow$	ClinTox ROC-AUC $\uparrow$	ESOL RMSE $\downarrow$	FreeSolv RMSE $\downarrow$	Lipo RMSE $\downarrow$	QM7 MAE $\downarrow$
<i>25% augmentation</i>							
MolCLR (Euclidean)	0.651 $\pm$ 0.002	0.606 $\pm$ 0.022	0.521 $\pm$ 0.033	1.652 $\pm$ 0.002	3.374 $\pm$ 0.044	1.009 $\pm$ 0.002	76.148 $\pm$ 0.095
Scaffold (Euclidean)	0.636 $\pm$ 0.002	0.619 $\pm$ 0.014	0.566 $\pm$ 0.006	1.577 $\pm$ 0.002	3.194 $\pm$ 0.011	0.984 $\pm$ 0.001	75.011 $\pm$ 1.117
MolCLR (Lorentz-unbounded)	0.656 $\pm$ 0.001	0.690 $\pm$ 0.002	0.489 $\pm$ 0.004	1.541 $\pm$ 0.001	2.896 $\pm$ 0.036	1.022 $\pm$ 0.004	81.901 $\pm$ 0.972
Scaffold (Lorentz-unbounded)	0.611 $\pm$ 0.001	0.742 $\pm$ 0.003	0.568 $\pm$ 0.010	1.682 $\pm$ 0.006	2.787 $\pm$ 0.060	0.992 $\pm$ 0.004	84.965 $\pm$ 3.803
MolCLR (Lorentz-bounded)	0.632 $\pm$ 0.006	0.593 $\pm$ 0.050	0.557 $\pm$ 0.011	1.778 $\pm$ 0.009	3.218 $\pm$ 0.054	1.016 $\pm$ 0.002	88.795 $\pm$ 1.323
Scaffold (Lorentz-bounded)	0.652 $\pm$ 0.003	0.654 $\pm$ 0.012	0.580 $\pm$ 0.011	1.573 $\pm$ 0.122	3.048 $\pm$ 0.018	0.999 $\pm$ 0.009	82.151 $\pm$ 1.028
<i>15% augmentation</i>							
MolCLR (Euclidean)	0.619 $\pm$ 0.002	0.608 $\pm$ 0.003	0.544 $\pm$ 0.012	1.606 $\pm$ 0.004	3.439 $\pm$ 0.052	0.992 $\pm$ 0.002	76.610 $\pm$ 1.193
Scaffold (Euclidean)	0.635 $\pm$ 0.003	0.646 $\pm$ 0.001	0.560 $\pm$ 0.003	1.421 $\pm$ 0.003	3.683 $\pm$ 0.030	1.004 $\pm$ 0.005	76.093 $\pm$ 0.634
MolCLR (Lorentz-unbounded)	0.638 $\pm$ 0.007	0.544 $\pm$ 0.003	0.480 $\pm$ 0.005	1.703 $\pm$ 0.007	3.660 $\pm$ 0.046	1.043 $\pm$ 0.002	104.229 $\pm$ 4.927
Scaffold (Lorentz-unbounded)	0.619 $\pm$ 0.006	0.643 $\pm$ 0.084	0.543 $\pm$ 0.033	1.712 $\pm$ 0.035	2.503 $\pm$ 0.015	1.016 $\pm$ 0.007	94.785 $\pm$ 0.764
MolCLR (Lorentz-bounded)	0.631 $\pm$ 0.001	0.579 $\pm$ 0.024	0.548 $\pm$ 0.007	1.674 $\pm$ 0.012	3.664 $\pm$ 0.007	1.041 $\pm$ 0.007	91.490 $\pm$ 1.892
Scaffold (Lorentz-bounded)	0.604 $\pm$ 0.004	0.656 $\pm$ 0.002	0.534 $\pm$ 0.014	1.639 $\pm$ 0.049	2.828 $\pm$ 0.025	1.049 $\pm$ 0.002	88.352 $\pm$ 0.666

MolCLR BACE at 25% (± 0.050), unbounded scaffold BACE at 15% (± 0.084) and unbounded scaffold QM7 at 25% (± 3.80).

The scaffold weight λ_s and the curvature c are themselves hyper-parameters; sensitivity analyses for both, together with the per-percentage augmentation results, are reported in Appendices A.9, A.10 and A.12.

4.4 Visualizing the learned representation

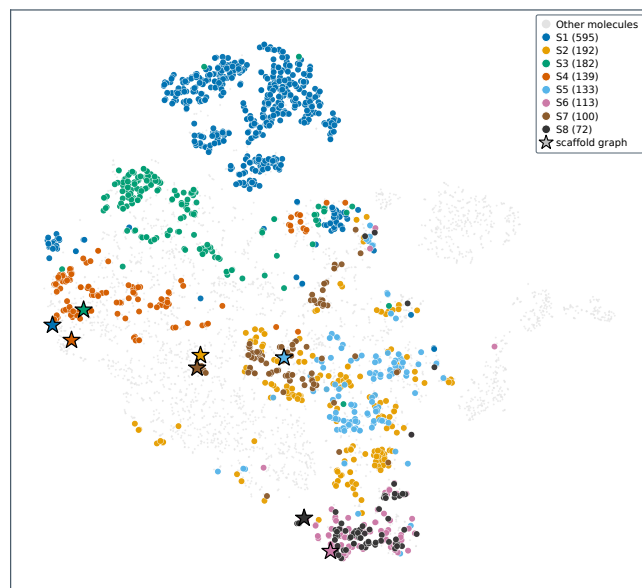
The retrieval probe quantifies how the embedding space is organized; Figure 5 shows it directly, for both geometries at 15% augmentation. For each model we take the frozen backbone representation h of 150,000 held-out molecules, L_2 -normalize it, reduce it to 50 principal components and project with t-SNE (perplexity 30, PCA initialization), so the map reflects the same cosine geometry used for retrieval. Eight Bemis–Murcko scaffolds are highlighted. They are frequent, but constrained to have pairwise distinct fused ring systems and ECFP4–Tanimoto similarity below 0.40, and are drawn from across the frequency distribution so that group sizes span 595 down to 72 members rather than clustering at one scale. Every molecule carrying one of them is coloured, 3,000 further molecules provide grey context, and each scaffold graph is passed through the same encoder and overlaid as a star. The selection uses no model output, so all four panels show the same molecules, the same eight cores and identical t-SNE settings; both hyperbolic models are the *bounded* Lorentz variant.

Two differences separate the scaffold-supervised panels from their matched baselines. First, the families become compact and nearly disjoint. Under either MolCLR baseline the colours run in interleaved streaks whereas under scaffold alignment the same

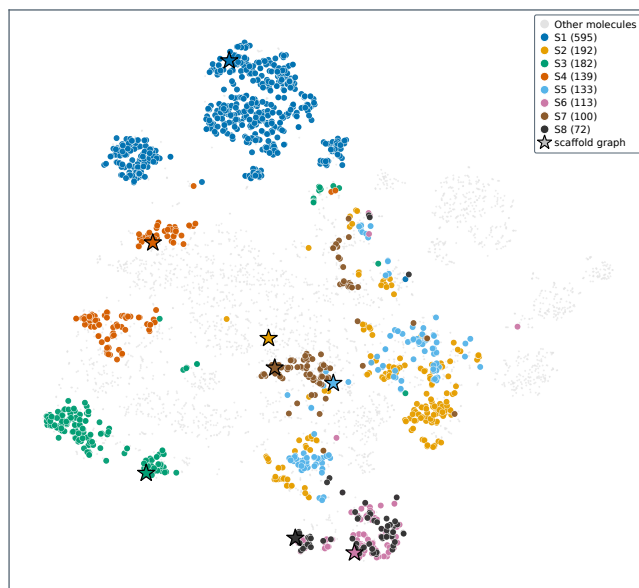
families collapse into isolated islands separated by grey space. Second, and more directly the relation the loss encodes, the scaffold markers move into their families: in both baselines only the two smallest cores have their star among their own molecules, the rest sitting in sparse regions away from the compounds that contain them, as expected for a model that never sees a scaffold during training. Under scaffold alignment most stars lie inside or on the boundary of their own colour, which is the geometric statement behind the molecule–scaffold ranking. Qualitatively, both effects appear strongest for the bounded hyperbolic pair: its baseline (c) is the least organized of the four panels and its scaffold model (d) the most cleanly separated, in line with the retrieval ordering of Table 1. Several families still resolve into two or three islands, reflecting substituent variation within one core, and a few small cores remain adjacent where their ring systems are genuinely related.

4.5 Limitations

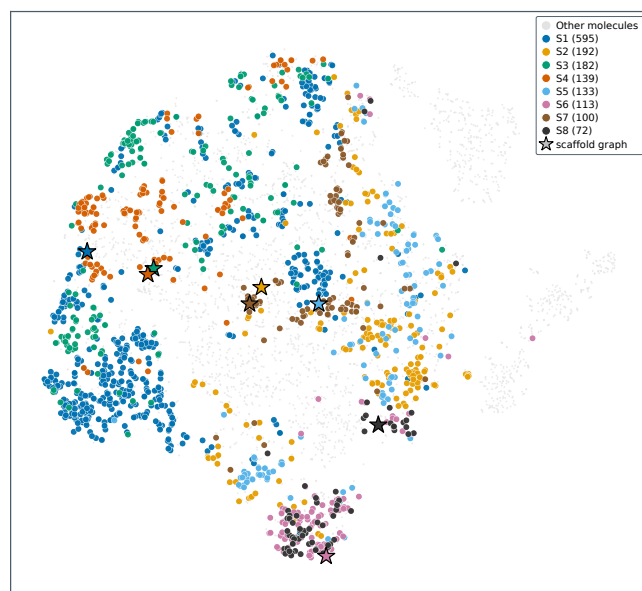
Positioning within structure-aware molecular SSL. This study is a controlled analysis of scaffold supervision as an additional structural signal rather than a comprehensive benchmark of prior structure-aware molecular SSL. Our results establish the effect of BM scaffold supervision relative to matched MolCLR retrains at an equal data and architecture budget, and do not claim superiority over approaches that inject chemistry through the augmentation operator. We consider only Bemis–Murcko scaffolds and supervise a single level of that hierarchy, while alternative scaffold definitions, other structural abstractions, or a chain of successively more generic



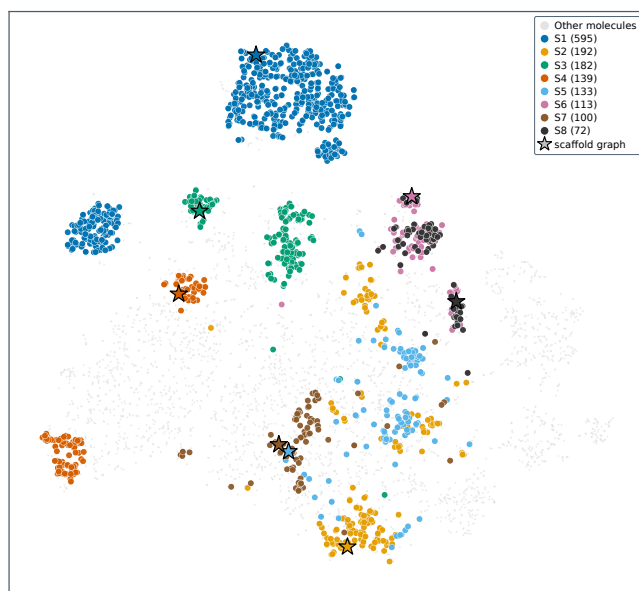
(a) MolCLR — Euclidean.



(b) Scaffold supervision — Euclidean.



(c) MolCLR — Lorentz (bounded).



(d) Scaffold supervision — Lorentz (bounded).

Figure 5: Backbone-space h organization at 15% augmentation, for the Euclidean pair (top) and the bounded Lorentz pair (bottom). Colours identify molecules carrying one of eight frequent, mutually dissimilar Bemis–Murcko scaffolds; grey points are additional molecules from the shared evaluation subset; stars are the embeddings of the corresponding scaffold graphs. All four panels use the same molecules, the same scaffolds and the same t-SNE settings, so differences between panels are differences between models. Under both geometries, scaffold supervision compacts the families and moves most scaffold graphs into the families they define.

cores may induce different organization. Matched comparisons with related approaches remain important future work.

Evaluation protocol. We probe rather than fine-tune, keeping the encoder frozen and training only a linear head, so our numbers are lower than the published MolCLR by construction and are not a comparison with it. Reported standard deviations are over three probing seeds with a single pre-trained encoder per configuration, so variance introduced by pre-training itself is not measured. The retrieval pool is deliberately scaffold-balanced, so its absolute values depend on how the pool was built and only the comparisons between models transfer, and the validation split it draws from was used to select checkpoints. All retrieval and visualization is carried out in the Euclidean backbone space h , so the hyperbolic runs are assessed through the effect of their objective on a Euclidean readout rather than in the hyperbolic space itself; the depth-based hierarchy diagnostics were not recorded at convergence for the bounded runs, so whether a molecule-scaffold depth ordering forms on the hyperboloid remains untested.

Dataset size and Lorentz ablations. Pre-training is restricted to a 1M-molecule subset, so the magnitude of the observed effects may change at larger scale, and no scaffold-frequency balancing or cap is applied. The molecule-scaffold ranking of Appendix A.15 suggests that the unbounded hyperbolic model errs on a minority of molecules rather than on average; whether this tail accounts for its uneven downstream profile is a hypothesis we do not test. The curvature sweep covers only $c \in \{0.5, 1, 2\}$ and predates the bounded formulation, the bounded implementation uses a single $r_{\max}$ that was not independently swept, and the two formulations differ in whether the feature scale is learned or fixed.

5 CONCLUSION

We studied what happens when a molecule’s Bemis–Murcko scaffold is used to supervise molecular contrastive learning directly, and whether the choice of latent geometry changes the outcome. Measured on the graph the encoder consumes, no strength of the inherited stochastic augmentation in the tested range keeps most views connected while still posing a non-trivial pretext task, and connectivity is only a necessary condition for a chemically faithful view; but the augmentation does not have to carry the chemistry, because it can be supplied by a separate term of the objective instead. Supplying the scaffold as an explicit positive reorganizes the embedding space around chemical cores. A molecule’s nearest neighbours share its core far more often, and they remain structurally closer to it even when every molecule carrying the same scaffold is deleted from the candidate set, so the effect is not merely a reproduction of the relation the model was trained on. The same embeddings improve property prediction on several benchmarks, and the pattern of gains is broadly consistent with how much each target depends on the ring system. The seemingly symmetric intervention, withholding same-scaffold molecules from the negative set, does not help; we hypothesize that this is because those pairs are also among the objective’s most informative negatives. The Lorentz objective sharpens scaffold organization only in combination with the scaffold signal and not on its own, and it offers no consistent downstream advantage over Euclidean training; where

it appears to, the advantage belongs to a formulation whose radial coordinate has collapsed, although our two Lorentz formulations differ in more than that one respect. Explicitly teaching a model the relation between a molecule and its structural core is therefore a cheap and generally reliable way to shape a molecular embedding space, while what that organization is worth depends on the task it is asked to serve.

REFERENCES

- [1] Dávid Bajusz, Anita Rácz, and Károly Héberger. 2015. Why is Tanimoto index an appropriate choice for fingerprint-based similarity calculations? *Journal of Cheminformatics* 7 (2015), 20. <https://doi.org/10.1186/s13321-015-0069-3>
- [2] Guy W. Bemis and Mark A. Murcko. 1996. The Properties of Known Drugs. 1. Molecular Frameworks. *Journal of Medicinal Chemistry* 39, 15 (1996), 2887–2893. <https://doi.org/10.1021/jm9602928>
- [3] Regine S. Bohacek, Colin McMartin, and Wayne C. Guida. 1996. The Art and Practice of Structure-Based Drug Design: A Molecular Modeling Perspective. *Medicinal Research Reviews* 16, 1 (1996), 3–50. [https://doi.org/10.1002/\(SICI\)1098-1128\(199601\)16:1<3::AID-MED1>3.0.CO;2-6](https://doi.org/10.1002/(SICI)1098-1128(199601)16:1<3::AID-MED1>3.0.CO;2-6)
- [4] Thomas Breuel. 2020. WebDataset: Efficient PyTorch I/O library for Large Datasets, Many Files, Many GPUs. <https://github.com/webdataset/webdataset>. (2020).
- [5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In *International Conference on Machine Learning (ICML)*. <https://arxiv.org/abs/2002.05709>
- [6] Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. 2020. ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction. *CoRR abs/2010.09885* (2020). [arXiv:2010.09885](https://arxiv.org/abs/2010.09885) <https://arxiv.org/abs/2010.09885>
- [7] Jörg Degen, Christof Wegscheid-Gerlach, Andrea Zaliani, and Matthias Rarey. 2008. On the Art of Compiling and Using “Drug-Like” Chemical Fragment Spaces. *ChemMedChem* 3, 10 (2008), 1503–1507. <https://doi.org/10.1002/cmdc.200800178>
- [8] Delft AI Cluster (DAIC). 2024. The Delft AI Cluster. (2024). https://doi.org/10.4233/rrid:scr_025091
- [9] Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Ramakrishna Vedantam. 2023. Hyperbolic Image-Text Representations. In *International Conference on Machine Learning (ICML)*. <https://arxiv.org/abs/2304.09172>
- [10] Terrance DeVries and Graham W. Taylor. 2017. Improved Regularization of Convolutional Neural Networks with Cutout. *CoRR abs/1708.04552* (2017). <https://arxiv.org/abs/1708.04552>
- [11] Yin Fang, Qiang Zhang, Haihong Yang, Xiang Zhuang, Shumin Deng, Wen Zhang, Ming Qin, Zhuo Chen, Xiaohui Fan, and Huajun Chen. 2022. Molecular Contrastive Learning with Chemical Element Knowledge Graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 3968–3976.
- [12] Yin Fang, Qiang Zhang, Ningyu Zhang, Zhuo Chen, Xiang Zhuang, Xin Shao, Xiaohui Fan, and Huajun Chen. 2023. Knowledge Graph-Enhanced Molecular Contrastive Learning with Functional Prompt. *Nature Machine Intelligence* 5, 5 (2023), 542–553. <https://doi.org/10.1038/s42256-023-00654-0>
- [13] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. 2017. Neural Message Passing for Quantum Chemistry. In *International Conference on Machine Learning (ICML)*. <https://arxiv.org/abs/1704.01212>
- [14] William L. Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*. <https://arxiv.org/abs/1706.02216>
- [15] Kaveh Hassani and Amir Hosein Khasahmadi. 2020. Contrastive Multi-View Representation Learning on Graphs. In *International Conference on Machine Learning (ICML)*. <https://arxiv.org/abs/2006.05582>
- [16] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum Contrast for Unsupervised Visual Representation Learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. <https://arxiv.org/abs/1911.05722>
- [17] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. 2020. Open Graph Benchmark: Datasets for Machine Learning on Graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*. <https://arxiv.org/abs/2005.00687>
- [18] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay S Pande, and Jure Leskovec. 2020. Strategies for Pre-training Graph Neural Networks. In *International Conference on Learning Representations (ICLR)*.
- [19] Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, Leonid Zaslavsky, Jian Zhang, and Evan E Bolton. 2019. PubChem 2019 update: improved access to chemical data. *Nucleic Acids Research* 47, D1 (01 2019), D1102–D1109. <https://doi.org/10.1093/nar/gky1033>

- [20] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations (ICLR)*. <https://arxiv.org/abs/1609.02907>
- [21] Greg Landrum. 2010. RDKit: Open-source cheminformatics. <https://www.rdkit.org/>. (2010).
- [22] Maximilian Nickel and Douwe Kiela. 2017. Poincaré Embeddings for Learning Hierarchical Representations. *CoRR* abs/1705.08039 (2017). [arXiv:1705.08039](https://arxiv.org/abs/1705.08039) <http://arxiv.org/abs/1705.08039>
- [23] Maximilian Nickel and Douwe Kiela. 2018. Learning Continuous Hierarchies in the Lorentz Model of Hyperbolic Geometry. In *International Conference on Machine Learning (ICML)*. <https://arxiv.org/abs/1806.03417>
- [24] Pavel G. Polishchuk, Timur I. Madzhidov, and Alexandre Varnek. 2013. Estimation of the Size of Drug-Like Chemical Space Based on GDB-17 Data. *Journal of Computer-Aided Molecular Design* 27, 8 (2013), 675–679. <https://doi.org/10.1007/s10822-013-9672-4>
- [25] David Rogers and Mathew Hahn. 2010. Extended-Connectivity Fingerprints. *Journal of Chemical Information and Modeling* 50, 5 (2010), 742–754. <https://doi.org/10.1021/ci100050t>
- [26] Yu Rong, Yatao Bian, Tingyang Xu, Weiyang Xie, Ying Wei, Wenbing Huang, and Junzhou Huang. 2020. Self-Supervised Graph Transformer on Large-Scale Molecular Data. In *Advances in Neural Information Processing Systems (NeurIPS)*. <https://arxiv.org/abs/2007.02835>
- [27] Rik Sarkar. 2011. Low Distortion Delaunay Embedding of Trees in Hyperbolic Plane. In *International Symposium on Graph Drawing*. 355–366. https://doi.org/10.1007/978-3-642-25878-7_34
- [28] Kristof T. Schütt, Huziel E. Sauceda, Pieter-Jan Kindermans, Alexandre Tkatchenko, and Klaus-Robert Müller. 2018. SchNet – A Deep Learning Architecture for Molecules and Materials. *The Journal of Chemical Physics* 148, 24 (2018), 241722. <https://doi.org/10.1063/1.5019779>
- [29] Connor Shorten and Taghi M. Khoshgoftaar. 2019. A Survey on Image Data Augmentation for Deep Learning. *Journal of Big Data* 6, 1 (2019), 60. <https://doi.org/10.1186/s40537-019-0197-0>
- [30] Fan-Yun Sun, Jordan Hoffmann, Vikas Verma, and Jian Tang. 2020. InfoGraph: Unsupervised and Semi-supervised Graph-Level Representation Learning via Mutual Information Maximization. In *International Conference on Learning Representations (ICLR)*. <https://arxiv.org/abs/1908.01000>
- [31] Mengying Sun, Jing Xing, Huijun Wang, Bin Chen, and Jiayu Zhou. 2021. MoCL: Data-Driven Molecular Fingerprint via Knowledge-Aware Contrastive Learning from Molecular Graph. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*. 3585–3594. <https://doi.org/10.1145/3447548.3467186>
- [32] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. 2020. What Makes for Good Views for Contrastive Learning?. In *Advances in Neural Information Processing Systems (NeurIPS)*. <https://arxiv.org/abs/2005.10243>
- [33] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation Learning with Contrastive Predictive Coding. *CoRR* abs/1807.03748 (2018). <https://arxiv.org/abs/1807.03748>
- [34] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research* 9, 86 (2008), 2579–2605.
- [35] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *International Conference on Learning Representations (ICLR)*. <https://arxiv.org/abs/1710.10903>
- [36] Petar Veličković, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, and R. Devon Hjelm. 2019. Deep Graph Infomax. In *International Conference on Learning Representations (ICLR)*. <https://arxiv.org/abs/1809.10341>
- [37] Yue Wan, Jialu Wu, Tingjun Hou, Chang-Yu Hsieh, and Xiaowei Jia. 2025. Multi-Channel Learning for Integrating Structural Hierarchies into Context-Dependent Molecular Representation. *Nature Communications* 16 (2025), 413. <https://doi.org/10.1038/s41467-024-55082-4>
- [38] Yuyang Wang, Rishikesh Magar, Chen Liang, and Amir Barati Farimani. 2022. Improving Molecular Contrastive Learning via Faulty Negative Mitigation and Decomposed Fragment Contrast. *Journal of Chemical Information and Modeling* 62, 11 (2022), 2713–2725. <https://doi.org/10.1021/acs.jcim.2c00495>
- [39] Yuyang Wang, Jianren Wang, Zhonglin Cao, and Amir Barati Farimani. 2021. MolCLR: Molecular Contrastive Learning of Representations via Graph Neural Networks. *CoRR* abs/2102.10056 (2021). [arXiv:2102.10056](https://arxiv.org/abs/2102.10056) <https://arxiv.org/abs/2102.10056>
- [40] David Weininger. 1988. SMILES, a Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules. *Journal of Chemical Information and Computer Sciences* 28, 1 (1988), 31–36. <https://doi.org/10.1021/ci00057a005>
- [41] Zhenxing Wu, Dejun Jiang, Chang-Yu Hsieh, Guangyong Chen, Ben Liao, Dongsheng Cao, and Tingjun Hou. 2021. Hyperbolic Relational Graph Convolution Networks Plus: A Simple but Highly Efficient QSAR-Modeling Method. *Briefings in Bioinformatics* 22, 5 (2021), bbab112. <https://doi.org/10.1093/bib/bbab112>
- [42] Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay Pande. 2018. MoleculeNet: A Benchmark for Molecular Machine Learning. *Chemical Science* 9, 2 (2018), 513–530. <https://doi.org/10.1039/C7SC02664A>
- [43] Tete Xiao, Xiaolong Wang, Alexei A. Efros, and Trevor Darrell. 2021. What Should Not Be Contrastive in Contrastive Learning. In *International Conference on Learning Representations (ICLR)*. <https://arxiv.org/abs/2008.05659>
- [44] Zhaoping Xiong, Dingyan Wang, Xiaohong Liu, Feisheng Zhong, Xiaozhe Wan, Xutong Li, Zhaojun Li, Xiaomin Luo, Kaixian Chen, Hualiang Jiang, and Mingyue Zheng. 2020. Pushing the Boundaries of Molecular Representation for Drug Discovery with the Graph Attention Mechanism. *Journal of Medicinal Chemistry* 63, 16 (2020), 8749–8760. <https://doi.org/10.1021/acs.jmedchem.9b00959>
- [45] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2019. How Powerful are Graph Neural Networks?. In *International Conference on Learning Representations (ICLR)*. <https://arxiv.org/abs/1810.00826>
- [46] Kevin Yang, Kyle Swanson, Wengong Jin, Connor W. Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian Kelley, Miriam Mathea, Regina Barzilay, Klavs F. Jensen, William H. Green, and Tommi S. Jaakkola. 2019. Analyzing Learned Molecular Representations for Property Prediction. *Journal of Chemical Information and Modeling* 59, 8 (2019), 3370–3388. <https://doi.org/10.1021/acs.jcim.9b00237>
- [47] Yuning You, Tianlong Chen, Yang Shen, and Zhangyang Wang. 2021. Graph Contrastive Learning Automated. In *International Conference on Machine Learning (ICML)*. <https://arxiv.org/abs/2106.07594>
- [48] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph Contrastive Learning with Augmentations. *CoRR* abs/2010.13902 (2020). [arXiv:2010.13902](https://arxiv.org/abs/2010.13902) <https://arxiv.org/abs/2010.13902>
- [49] Xuan Zang, Xianbing Zhao, and Buzhou Tang. 2023. Hierarchical Molecular Graph Self-Supervised Learning for Property Prediction. *Communications Chemistry* 6 (2023), 34. <https://doi.org/10.1038/s42004-023-00825-5>
- [50] Ziqiao Zhang, Ailin Xie, Jihong Guan, and Shuigeng Zhou. 2023. Molecular Property Prediction by Semantic-Invariant Contrastive Learning. *Bioinformatics* 39, 8 (2023), btad462. <https://doi.org/10.1093/bioinformatics/btad462>
- [51] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. 2020. Random Erasing Data Augmentation. In *AAAI Conference on Artificial Intelligence (AAAI)*. <https://arxiv.org/abs/1708.04896>
- [52] Gengmo Zhou, Zhifeng Gao, Qiankun Ding, Hang Zheng, Hongteng Xu, Zhewei Wei, Linfeng Zhang, and Guolin Ke. 2023. Uni-Mol: A Universal 3D Molecular Representation Learning Framework. In *International Conference on Learning Representations (ICLR)*. <https://openreview.net/forum?id=6K2RM6wVqKu>
- [53] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. 2021. Graph Contrastive Learning with Adaptive Augmentation. In *Proceedings of the Web Conference (WWW)*. <https://arxiv.org/abs/2010.14945>

A APPENDIX

For pre-training and finetuning, the Delft AI Cluster (DAIC) [8] was used.

A.1 Graph encoder details

This appendix gives the encoder formulation summarized in Section 2.4.

Message passing. A GNN [13] represents node v by $h_v^{(k)}$ after k layers, initialized at $h_v^{(0)} = x_v$. Each layer aggregates information from the neighbours $\mathcal{N}(v)$ of a node and combines it with the node’s own state:

$$a_v^{(k)} = \text{AGGREGATE}^{(k)}\left(\left\{h_u^{(k-1)} : u \in \mathcal{N}(v)\right\}\right),$$

$$h_v^{(k)} = \text{COMBINE}^{(k)}\left(h_v^{(k-1)}, a_v^{(k)}\right).$$

After K layers each node embedding summarises a K -hop neighbourhood, and a permutation-invariant READOUT (pooling) produces a single graph-level vector $h_G = \text{READOUT}(\{h_v^{(K)} : v \in V\})$.

GIN and GINE. The expressive power of a GNN, i.e. its ability to tell apart graphs that are not the same up to a relabelling of nodes (non-isomorphic graphs), is bounded by the Weisfeiler–Lehman test and depends critically on the choice of aggregation. The Graph Isomorphism Network (GIN) [45] attains this upper bound with a sum aggregator and a multilayer perceptron (MLP), which together

form an injective function over multisets:

$$h_v^{(k)} = \text{MLP}^{(k)} \left((1 + \epsilon^{(k)}) h_v^{(k-1)} + \sum_{u \in N(v)} h_u^{(k-1)} \right), \quad (13)$$

where $\epsilon^{(k)}$ is a (learnable or fixed) scalar; this makes GIN strictly more discriminative than mean- or max-based architectures such as the Graph Convolutional Network (GCN) [20]. Since bonds carry chemically meaningful information, we use the edge-conditioned variant GINE, which embeds the discrete bond features into the node-feature space and adds them to the neighbor messages before aggregation:

$$h_v^{(k)} = \text{MLP}^{(k)} \left(\sum_{u \in N(v) \cup \{v\}} \text{ReLU}(h_u^{(k-1)} + W_e e_{uv}) \right), \quad (14)$$

where $W_e e_{uv}$ is the learned embedding of the bond feature and self-loops are added so that each atom also passes its own state forward.

Configuration. The encoder stacks $K = 5$ such layers with hidden dimension 300, each followed by batch normalization and a ReLU non-linearity, and uses global mean pooling as the readout. The pooled vector is passed through a linear layer to produce the molecular representation $h \in \mathbb{R}^d$ with $d = 512$, and through a small projection MLP to produce the embedding z used by the contrastive objective. These choices follow MolCLR so that our results remain comparable to it.

A.2 Primer on hyperbolic geometry

This appendix expands the geometric background summarised in Section 2.7.

Riemannian manifolds. A smooth manifold $\mathcal{M}$ generalises the notion of a smooth surface to arbitrary dimension: every point has a local neighbourhood that looks like Euclidean space. A *Riemannian manifold* $(\mathcal{M}, g)$ is a smooth manifold equipped with a Riemannian metric g , a smoothly varying collection of inner products g_x , one for each point $x \in \mathcal{M}$. Each g_x acts on the *tangent space* $T_x \mathcal{M}$, a Euclidean vector space that provides a local linear approximation of the manifold at x . Euclidean space $\mathbb{R}^n$ is itself a Riemannian manifold whose metric is the standard inner product and whose curvature is everywhere zero. An n -dimensional hyperbolic space cannot be embedded in $\mathbb{R}^n$ while preserving both distances and angles; the Poincaré ball model [22] lives inside $\mathbb{R}^n$ at the cost of distorting distances, whereas the Lorentz (hyperboloid) model [23] we adopt represents it as a submanifold of $\mathbb{R}^{n+1}$.

Tangent space and general exponential/logarithmic maps. The tangent space at a point $z \in \mathcal{L}^n$ is the set of vectors orthogonal to z under the Lorentzian inner product, $T_z \mathcal{L}^n = \{v \in \mathbb{R}^{n+1} : \langle z, v \rangle_{\mathcal{L}} = 0\}$. To move between the tangent space and the manifold we use the *exponential map* $\exp_z : T_z \mathcal{L}^n \rightarrow \mathcal{L}^n$ and its inverse, the *logarithmic map* $\log_z : \mathcal{L}^n \rightarrow T_z \mathcal{L}^n$. At a general base point these

are

$$\exp_z(v) = \cosh(\sqrt{c} \|v\|_{\mathcal{L}}) z + \sinh(\sqrt{c} \|v\|_{\mathcal{L}}) \frac{v}{\sqrt{c} \|v\|_{\mathcal{L}}}, \quad (15)$$

$$\log_z(x) = \frac{\text{arccosh}(-c \langle z, x \rangle_{\mathcal{L}})}{\sqrt{(c \langle z, x \rangle_{\mathcal{L}})^2 - 1}} (x + c \langle z, x \rangle_{\mathcal{L}} z), \quad (16)$$

where $\|v\|_{\mathcal{L}} = \sqrt{\langle v, v \rangle_{\mathcal{L}}}$ is the Lorentzian norm and the exponential map describes how the flat tangent space wraps onto the curved manifold. Our method only ever applies these at the origin $z = O$, giving the simplified map of Equation (6).

A.3 Hyperbolic optimisation details

This appendix details the feature scale and curvature of the hyperbolic variant (Section 3.5). In the unbounded formulation we parametrize the feature scale as $\alpha = \exp(a)$ with a clamped so that $\alpha \leq 1$, which prevents it from ever up-scaling the encoder norm, and initialize it at $\alpha \approx d^{-1/2}$ (here $\alpha \approx 0.044$ for $d = 512$). This keeps the lifted points in a numerically stable region of the manifold at initialization, but not thereafter: because a is learned jointly with the encoder, $\|\alpha z\|$ grows during training until the sinh clamp of $\exp_O$ becomes active, which is the saturation documented in Appendix A.4. The bounded formulation therefore holds α fixed at 0.044 and does not learn it, relying on the radial bound $r_{\max} = 5$ of Equation (12) rather than on α to keep the embeddings in range. The single α and curvature c are owned by the contrastive-loss module and passed to the scaffold loss at evaluation time, so that both losses project onto the hyperboloid identically. The curvature is stored as $\log c$ for numerical stability; by default $c = 1$ is fixed, and when it is instead learned jointly with the model weights it is clamped to $[c_0/10, 10 c_0]$ around its initial value c_0 , as in MERU [9]. We train in full 32-bit precision throughout, since the hyperbolic operations are sensitive to the reduced range of half precision.

A.4 Saturation of the unbounded Lorentz projection

This appendix documents the failure that motivates the bounded formulation of Section 3.5. Figure 6 shows the three validation diagnostics over a 50-epoch unbounded run.

Three features of the curves identify the mechanism. First, the two hierarchy diagnostics do not decay towards zero, they *snap* to it at a single epoch and stay pinned there, which is what a hard clamp produces and what a gradual loss of signal does not. Second, they reach exactly zero rather than a small positive value: the entailment rate uses a strict inequality, so equal depths give zero rather than the 0.5 that random depths would give. Third, the similarity gap is unaffected, because the clamp preserves the direction of the space component and the geodesic distance between two points at equal radius still depends on the angle between them. The objective is therefore optimizing happily on what has become, in effect, a sphere of fixed hyperbolic radius.

The practical consequences are twofold. The learned feature scale α of the unbounded runs belongs to the loss module and was not checkpointed with the encoder, so the lifted embeddings of those runs cannot be reconstructed after training; this is why all read-outs in this thesis are taken in the Euclidean backbone

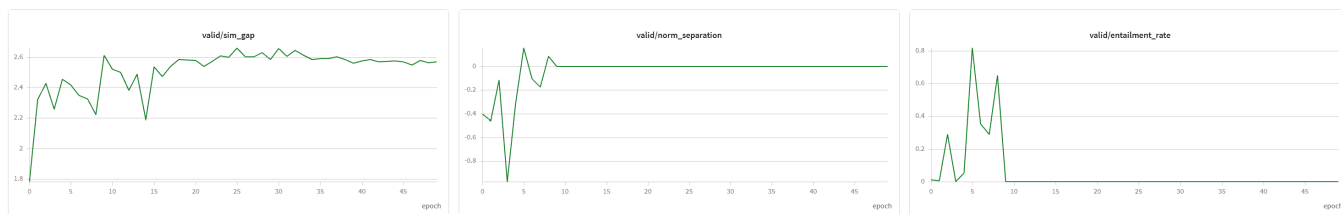


Figure 6: Validation diagnostics of an unbounded Lorentz run. *Left:* the positive-negative similarity gap rises and converges normally, so the contrastive objective is being optimized. *Middle and right:* the norm separation and the entailment rate fluctuate for roughly nine epochs and then become identically zero for the remainder of training. This is the signature of the sinh clamp in $\exp_O$ becoming active for every input: all embeddings are placed at the same geodesic radius, so no scaffold can be strictly closer to the origin than the molecule that contains it, and the mean depth difference is exactly zero rather than merely small. This convergence to 0 of the norm separation and entailment rate happens for all unbounded hyperbolic models.

space h rather than in z . And the norm-separation and entailment-rate curves of the unbounded models carry no information about whether a hierarchy formed, so they are reported here only as the diagnosis of this failure and are not used as evidence in the results.

A.5 Data preprocessing and exploration

A.5.1 Pre-training dataset. Our pre-training data is drawn from the PubChem-10M corpus [19], a collection of roughly ten million unique, unlabeled molecules distributed as SMILES strings [40] and originally curated for the ChemBERTa pre-training study [6]. We pre-train on a one-million-molecule subset of it. Every model in this thesis sees exactly this subset, so all comparisons are at an equal data budget; the consequence for comparability with the published MolCLR, which uses the full corpus, is discussed in the limitations. All statistics reported below are computed on the one million molecules we actually train on, not on the parent corpus.

From each SMILES string we use RDKit [21] to construct the molecular graph, extract its atom and bond features, and derive the corresponding Bemis-Murcko scaffold [2]. Of the 1,000,000 molecules, 971,137 (97.11%) contain at least one ring and therefore yield a non-empty scaffold, while 28,863 (2.89%) are acyclic and possess no Bemis-Murcko scaffold; the latter are excluded from the scaffold-alignment term but retained for the standard contrastive objective. The subset is chemically homogeneous and drug-like in the aggregate: a molecule carries on average 25.2 heavy atoms, 2.79 rings and a molecular weight of 341.9, and its scaffold retains 18.4 heavy atoms, so roughly three quarters of the heavy-atom skeleton survives the decomposition. Every molecule is a single connected component before augmentation, which is the baseline against which the fragmentation measurements are made.

A.5.2 Pre-processing. A practical bottleneck concerns how the molecular graphs are produced during training. The original MolCLR pipeline reconstructs each graph from its SMILES string every time the molecule is sampled, so over a 50-epoch run the same graph, together with all of its features and its scaffold, is parsed and rebuilt 50 times. Because RDKit parsing dominates the per-sample cost, this on-the-fly construction makes data loading the principal throughput limitation. Since none of this computation depends on the model state, we move it into a single offline pre-processing stage: every molecular graph, its features, and its scaffold are computed once and serialised to disk, decoupling graph construction from the training loop. This reduces end-to-end training time by

roughly 73%, a $3.7\times$ speedup. We store the pre-computed examples using the WebDataset library [4], partitioning the subset into four shards of 250,000 molecules each (approximately 1.1 GB per shard); the sharded layout lets the data loader stream examples in parallel by assigning a disjoint set of shards to each worker.

A.5.3 Training and validation split. The split is taken at shard granularity and is fixed for every run: the first shard (250,000 molecules, of which 242,748 carry a scaffold) is held out for validation, and the remaining three (750,000 molecules, 728,389 with a scaffold) are used for training. The shards are statistically interchangeable in their bulk properties: the per-shard fraction of ring-bearing molecules varies only between 97.06% and 97.19%, and the Kolmogorov-Smirnov distance between the training and validation distributions is at most 0.0017 for heavy-atom count, ring count, molecular weight and scaffold size alike.

Their scaffold vocabularies, however, are largely disjoint, and this matters for how the validation split is read. Only 40,330 of the 129,463 scaffolds present in validation (31.2%) also occur in training, although those shared cores are the frequent ones, so 62.1% of validation *molecules* carry a scaffold the model has seen; the Jensen-Shannon divergence between the two scaffold distributions is 0.310 nats against a maximum of 0.693. Most validation scaffold classes are therefore unseen even though most validation molecules carry a seen core, so the split exposes the encoder to a substantial amount of unseen chemistry; since the retrieval pool is built from multi-member classes, which are more likely to be frequent and therefore seen, we do not quantify how much of the probe rests on unseen cores.

A.6 Data discovery and scaffold analysis

Because both the scaffold-alignment loss and the hyperbolic geometry are predicated on a molecule-scaffold hierarchy, in which many compounds share one core and cores themselves range from generic to specific, we characterize the scaffold composition of the pre-training subset directly. Applying Bemis-Murcko decomposition [2] to the 971,137 ring-bearing molecules yields 419,820 distinct scaffolds, an average of only 2.31 molecules per scaffold. This average is highly misleading: the distribution is extremely heavy-tailed, with a Gini coefficient of 0.55 over scaffold frequencies. A single scaffold, the benzene ring, is shared by 69,592 molecules, or 7.2% of everything with a scaffold at all, whereas 346,933 scaffolds, fully

82.6% of all distinct cores, are *singletons* occurring in exactly one molecule.

Figure 7 makes this concrete by plotting the scaffold rank–frequency relationship on logarithmic axes. The near-linear decay over four decades is the signature of an approximately power-law distribution, and integrating it over scaffold rank reveals how concentrated the subset is: the 4 most frequent scaffolds (the top 0.001%) already account for 9.1% of all molecules, the top 42 (0.01%) for 16.5%, the top 420 (0.1%) for 27.1%, the top 4,198 (1%) for 40.2%, and the top 41,982 (10%) for 57.9%. In other words, a vanishingly small fraction of scaffolds covers close to half of the data, while the long tail of hundreds of thousands of rare and singleton scaffolds covers the remainder.

The identity of the most frequent scaffolds is equally telling. Figure 8 shows the 50 most common cores. They are small, generic ring systems, benzene, pyridine, cyclohexane, pyrazole, thiophene and other simple mono- and bicyclic frameworks, of the kind that recur across vast numbers of otherwise unrelated compounds.

First, the frequency imbalance is why the scaffold-alignment loss draws its candidate pool from the *unique* scaffolds in a batch. The alternative is to include one scaffold entry per molecule, so that the denominator of Equation (10) runs over all N molecules rather than the $U \leq N$ distinct cores. A core carried by k molecules in the batch would then contribute k terms to the normalization instead of one, so the most frequent cores would dominate the softmax in proportion to their corpus frequency, and separating a molecule from a handful of ubiquitous cores such as benzene would account for most of the gradient regardless of which core it actually carries. Drawing on unique scaffolds gives every core exactly one term, whatever its multiplicity.

Second, the same imbalance is why the retrieval pool cannot be a uniform sample. Only 17,045 scaffolds, 4.1% of the vocabulary, have five or more members; a uniformly drawn pool would consist overwhelmingly of molecules that are the sole representative of their core, leaving nothing to retrieve, which is why the pool is balanced over multi-member scaffold classes and capped per class.

A.7 Training details

For pre-training, we optimize with Adam (initial learning rate 5×10^{-4} , weight decay 10^{-5}) and a batch size of 512 for 50 epochs, holding the learning rate constant for 10 warm-up epochs and then annealing it to zero with a cosine schedule. We set both temperatures to $\tau = \tau_s = 0.1$, matching the value used by MolCLR, and weight the scaffold term by $\lambda_s = 0.1$ in Equation (7). All hyperbolic runs fix the curvature at $c = 1$. The unbounded runs train the feature scale α jointly with the encoder at the same learning rate; the bounded runs hold it fixed at $\alpha = 0.044$ and set $r_{\max} = 5$ with the tanh squashing of Equation (12). The selection of λ_s (a grid search over $\{0.01, 0.05, 0.1, 0.2, 0.5\}$), of the augmentation strength, and of the curvature are reported in Appendix A.9, A.10, and A.12.

After pre-training, we evaluate the representations with linear probing: the backbone is frozen and only a single linear head is trained on top of the pooled 512-dimensional embedding, on an 80/10/10 scaffold split. Regression targets are z-score normalized only for QM7. The head is optimized with Adam (learning rate 10^{-3} , weight decay 10^{-6} , no learning-rate schedule) for 100 epochs with

a batch size of 32, using cross-entropy loss for classification and MSE (L1 for QM7) for regression. We select the checkpoint with the best validation metric and report its test-set performance.

A.8 Downstream benchmarks

This appendix expands the benchmark panel summarized in Section 3.7.

- **BBBP** (2,039 molecules, 1 task) labels whether a compound crosses the blood–brain barrier. For passively transported drugs this is governed by bulk physicochemical properties rather than by a specific interaction: lipophilicity, molecular weight, polar surface area, and hydrogen-bond donor count. It is thus a whole-molecule property in which the peripheral substituents, which modulate polarity, matter at least as much as the core.
- **BACE** (1,513 molecules, 1 task) labels binding activity against human β -secretase 1, a protease target in Alzheimer’s disease. Unlike BBBP this is a molecular-recognition task: activity is determined by how the compound fits a specific binding pocket. The dataset is assembled from a small number of congeneric series built around shared cores, so the label depends jointly on the scaffold, which fixes the binding mode, and on the substituents that tune affinity. It is therefore the benchmark most directly aligned with a scaffold-based prior.
- **ClinTox** (1,478 molecules, 2 tasks) contrasts FDA-approved drugs with compounds that failed clinical trials for toxicity. Toxicity of this kind is often traced to localized structural alerts, that is, reactive or metabolically liable functional groups, rather than to the overall shape of the molecule. The signal is therefore substructural and local, and the label distribution is strongly imbalanced, which makes the ROC-AUC noisy across seeds.
- **ESOL** (1,128 molecules, 1 task) is a regression of aqueous solubility. Solubility is driven by the balance between polar, hydrogen-bonding groups that favour solvation and large, rigid aromatic systems that favour the solid state. Rings enter mainly through their number and size, so a representation organized by ring system carries useful, if coarse, information.
- **FreeSolv** (642 molecules, 1 task) is a regression of the hydration free energy of small neutral organic molecules in water. Physically it is the cleanest of the property tasks, being a single solute–solvent interaction energy dominated by polar and hydrogen-bonding surface. It is also by far the smallest benchmark and contains mostly small, often acyclic compounds, so results on it are expected to be the least stable and are the least constrained by scaffold structure.
- **Lipo** (lipophilicity, 4,200 molecules, 1 task) is a regression of the octanol–water distribution coefficient at physiological pH. It is again a whole-molecule property, but one that depends on ionization state as well as hydrophobic surface, and it is the largest and most chemically diverse of the regression sets, which makes it the most stable regression signal in the panel.
- **QM7** (6,830 molecules, 1 task) is a regression of the atomization energy of small organic molecules with up to seven

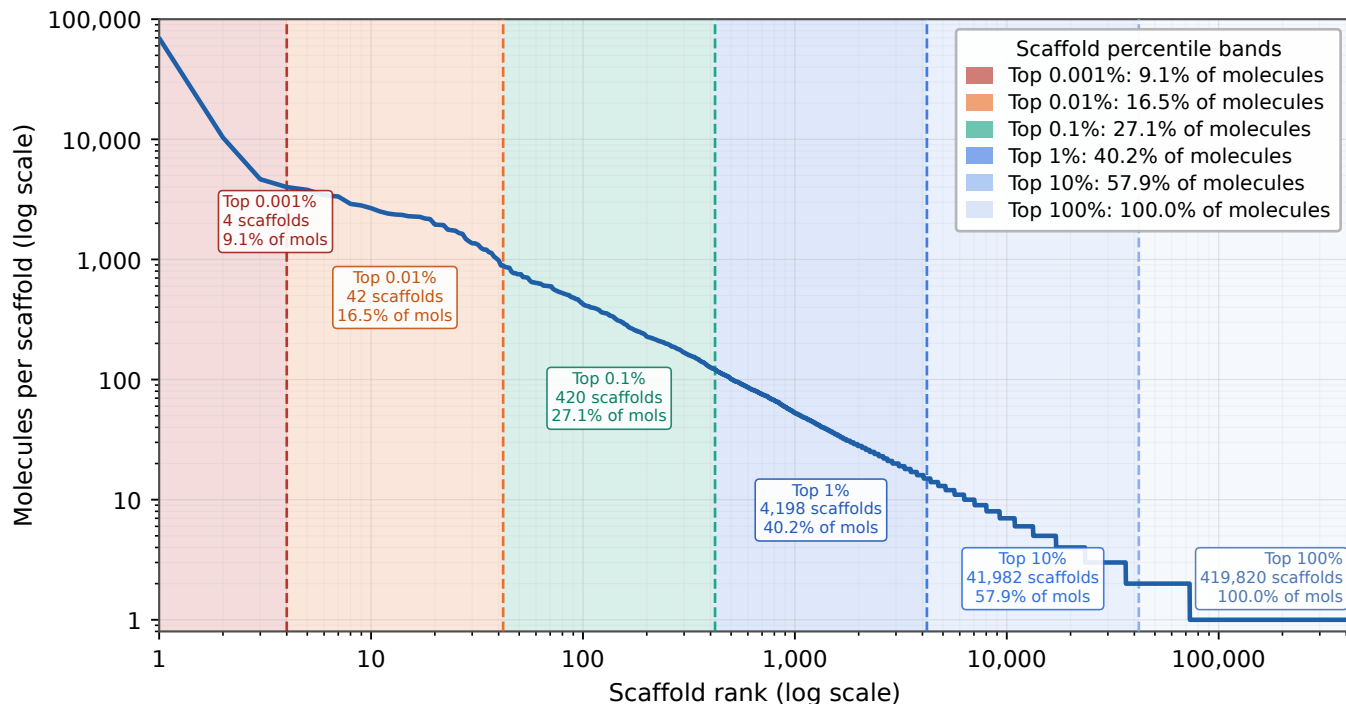


Figure 7: Scaffold rank–frequency distribution of the one-million-molecule pre-training subset on logarithmic axes, with shaded bands marking the molecule coverage of the most frequent scaffolds. The approximately linear decay indicates a power-law distribution: the top 1% of scaffolds (4,198 cores) already account for 40.2% of all molecules, while 82.6% of scaffolds occur in a single molecule.

heavy atoms, computed by density functional theory. This is a quantum-mechanical rather than a pharmacological property: it is determined by elemental composition and three-dimensional geometry and is close to additive in the atom and bond counts. A two-dimensional Bemis–Murcko scaffold abstracts away precisely the substituent-level composition that the label depends on, and many QM7 molecules are too small to possess a meaningful ring system at all, so QM7 acts as a stress test in which the scaffold prior is expected to be least helpful.

These are the four groups used in the main text. Scaffold splitting interacts with all four, since placing molecules with different cores in different folds additionally asks the representation to generalize to unseen chemistry rather than to interpolate within a congeneric series.

A.9 Training ablation for λ_s (scaffold-loss weight)

The scaffold weight λ_s in Equation (7) controls how strongly the scaffold-alignment term contributes relative to the contrastive objective. Because no labels are available at pre-training time, we select λ_s from the self-supervised training curves alone, sweeping $\lambda_s \in \{0.01, 0.05, 0.1, 0.2, 0.5\}$ on the scaffold model. All ablations performed here are in Euclidean geometry, with an augmentation strength of 25%. Figure 9 reports, per epoch, the fraction of the total validation loss contributed by the scaffold term, the validation contrastive accuracy, and the total validation loss.

Two observations bound the usable range of λ_s . First, the scaffold fraction scales directly and predictably with λ_s (left panel): the scaffold term accounts for roughly 1.5% of the loss at $\lambda_s = 0.01$, 6% at 0.05, 14% at 0.1, 28% at 0.2, and almost half ($\sim 48\%$) at 0.5. Second, the contrastive accuracy converges to essentially the same value (≈ 0.73) for every setting (middle panel), showing that the primary contrastive task is learned robustly regardless of λ_s , at least until the scaffold term begins to dominate. Together these indicate that λ_s should be large enough for the scaffold signal to meaningfully shape the representation, yet small enough that it remains a secondary, regularizing term rather than overriding the contrastive objective. At $\lambda_s = 0.1$ the scaffold term contributes a clear but minority share of the loss ($\sim 14\%$); larger values (0.2, 0.5) push it toward parity with the contrastive loss and we hypothesize it risks over-regularizing the embedding toward scaffolds, while smaller values (0.01, 0.05) render it nearly negligible. We therefore adopt $\lambda_s = 0.1$ throughout.

A.10 Training ablation for the augmentation percentage

Section 3.2 bracketed the augmentation strength by measuring how far each perturbation fragments the molecular graph; here we examine its effect on the contrastive task itself. We sweep the masking-and-deletion fraction over $\{5\%, 10\%, 15\%, 25\%\}$ for both the vanilla MolCLR model (Figure 10) and the scaffold model (Figure 11), tracking the validation contrastive accuracy and the total validation loss.

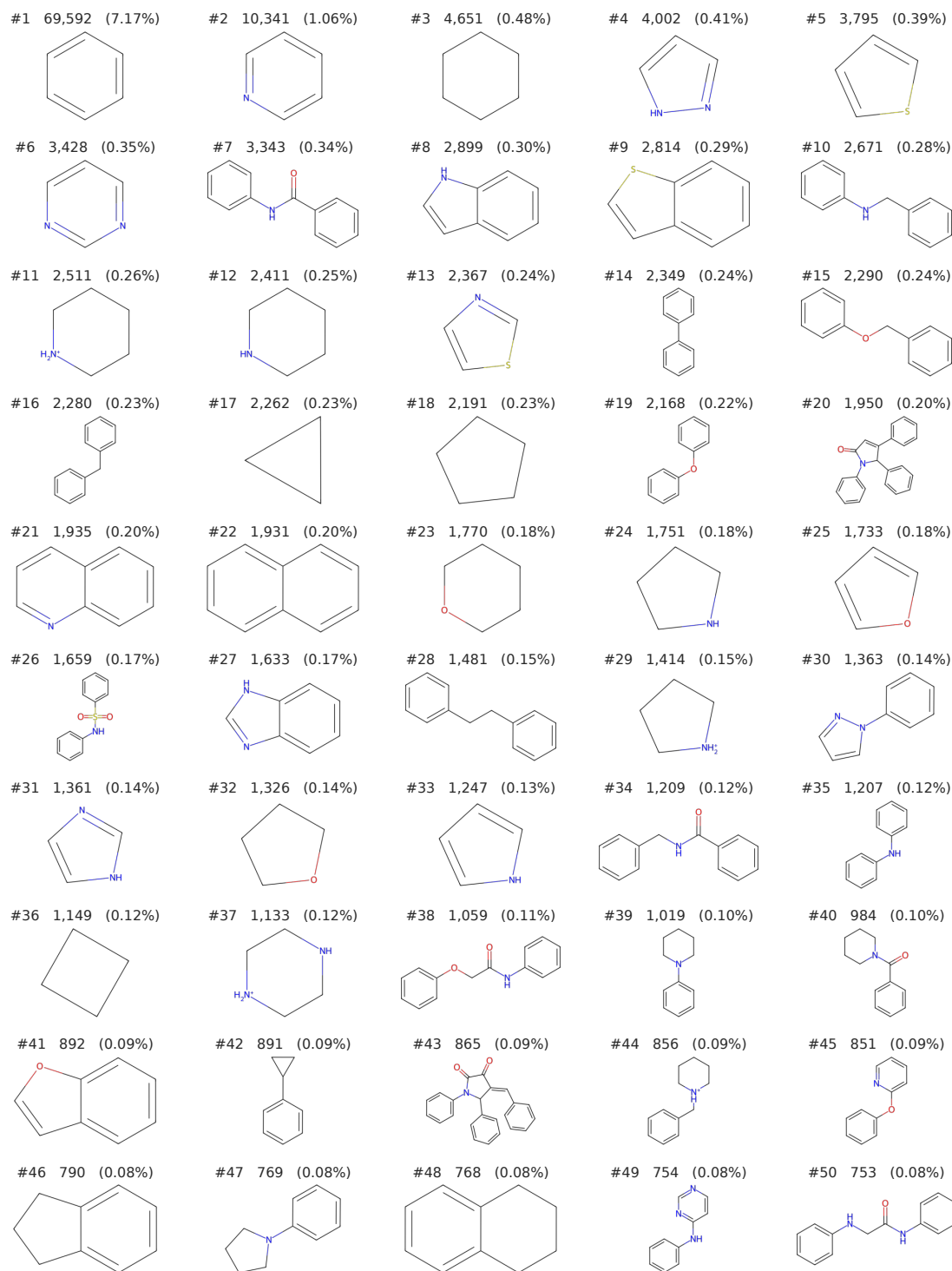


Figure 8: The 50 most frequent Bemis–Murcko scaffolds in the one-million-molecule pre-training subset, ordered by the number of molecules in which they occur, with that count and its share of all ring-bearing molecules above each structure. The most common cores are small, generic ring systems: benzene alone accounts for 7.2% of the subset, and no other scaffold reaches 1.1%.

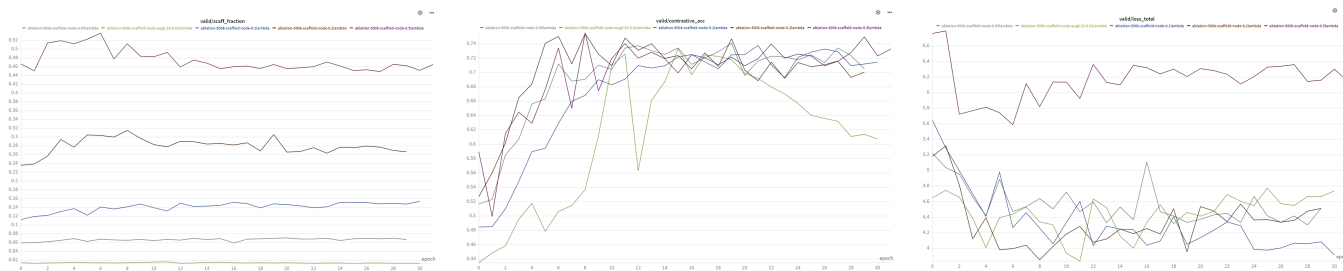


Figure 9: Scaffold-weight (λ_s) ablation for the scaffold model, $\lambda_s \in \{0.01, 0.05, 0.1, 0.2, 0.5\}$. *Left:* fraction of the total validation loss contributed by the scaffold term, which scales directly with λ_s . *Middle:* validation contrastive accuracy, which converges to ≈ 0.73 for all settings. *Right:* total validation loss. We select $\lambda_s = 0.1$, for which the scaffold term contributes a meaningful but minority ($\sim 14\%$) share of the loss.

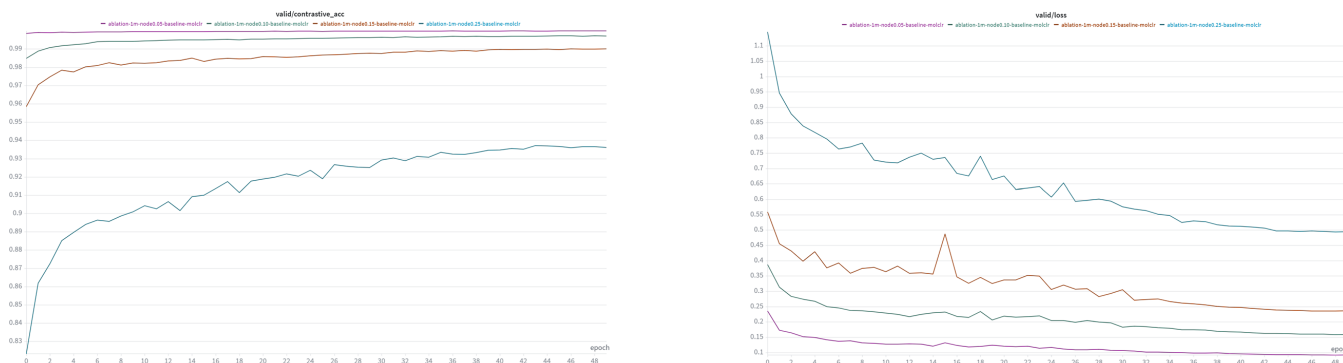


Figure 10: Augmentation-strength ablation for the vanilla MolCLR model, at 5%, 10%, 15%, 25%. *Left:* validation contrastive accuracy, which falls monotonically as augmentation strengthens (the task becomes harder). *Right:* total validation loss.

The contrastive accuracy decreases monotonically with augmentation strength. For the vanilla model it sits at ≈ 0.997 at 5%, ≈ 0.995 at 10%, ≈ 0.99 at 15%, and falls to ≈ 0.93 at 25%; the scaffold model behaves the same way. This is precisely the signature of task difficulty: under weak augmentation the two views of a molecule are almost identical, so matching them is trivial and the encoder is barely challenged, whereas stronger augmentation produces genuinely different views and a harder, more informative pretext task. A near-saturated contrastive accuracy ($\gtrsim 0.99$, as at 5–10%) is therefore a warning that the task is too easy to drive useful representation learning. Connectivity alone brackets rather than fixes the choice, since it is necessary but not sufficient for a usable view; these curves therefore carry the selection within that bracket: 25% is the strength at which the contrastive task becomes substantially harder, while 15% sits at the edge of saturation and is retained as the milder operating point.

A.11 Alternative augmentation operator: bond masking

Section 3.2 introduces bond masking as the operator that removes fragmentation: the edge is retained and its bond-type feature is replaced by a mask token, so every view keeps the molecule’s full heavy-atom skeleton. This appendix reports the corresponding pre-training runs. Only the augmentation changes; the encoder, losses, temperatures, λ_s , curvature, the one-million-molecule pre-training corpus, and the probing protocol are as in the main experiments, so Table 4 isolates the operator.

The split is by task, not by model. All four models improve on ClinTox and three of four on BBBP, while on BACE three of the four move the other way. Every model is worse on every regression benchmark, QM7 most sharply (75–85 to 109–392). That direction is the surprising one, since masking preserves every edge deletion would remove and is therefore the gentler operator: the losses cannot be attributed to heavier corruption. One possibility these runs do not test is that retaining every edge makes the pretext task easier at the same p , and weaker perturbations are held to yield less transferable representations [32, 43].

The ranking of the learning signals largely survives the change of operator. Under masking the Euclidean scaffold model is best of the four on BBBP, BACE, ESOL and Lipo and beats the baseline on five of seven benchmarks, echoing the main results; at a single strength this is consistent with the scaffold advantage being independent of the augmentation without establishing it. The hyperbolic models remain the most variable under either operator, giving both the largest QM7 error in this work (391.9) and the lowest BACE score (0.412).

A.12 Training ablation for the curvature of the hyperbolic model

We sweep the Lorentz curvature $c \in \{0.5, 1, 2\}$ and monitor the self-supervised diagnostics: the total validation loss, the contrastive accuracy, the scaffold-loss fraction, and the three hyperbolic hierarchy diagnostics (norm separation, entailment rate, and similarity gap). Figure 12 collects these curves. The ablations here are performed for

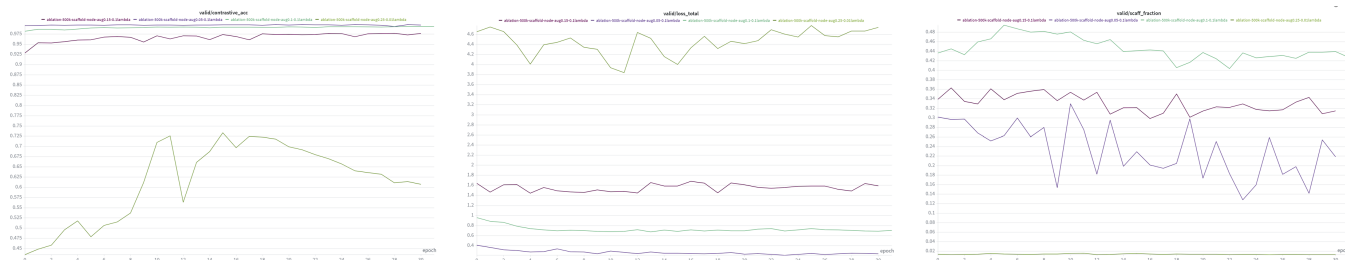


Figure 11: Augmentation-strength ablation for the scaffold model. *Left:* validation contrastive accuracy. *Middle:* total validation loss. *Right:* scaffold-loss fraction. As with the vanilla model, stronger augmentation yields a harder contrastive task (lower accuracy), supporting the choice of the 15% and 25% operating points.

Table 4: Effect of replacing bond deletion with bond masking. Learning signal (MolCLR vs. scaffold alignment) crossed with two geometries (Euclidean and Lorentz-unbounded, the latter labelled “Hyp.”), at 25% augmentation, under each operator. The upper block is bond deletion, reproduced from Table 3; the lower block is bond masking. Both blocks pre-train on the same one million molecules. Classification is ROC-AUC ($\uparrow$), regression is RMSE ($\downarrow$) except QM7 (MAE, $\downarrow$); mean $\pm$ std over three seeds. The best value per column is bolded within each block, so that the strongest model under each operator stays visible.

Model	BBBP ROC-AUC $\uparrow$	BACE ROC-AUC $\uparrow$	ClinTox ROC-AUC $\uparrow$	ESOL RMSE $\downarrow$	FreeSolv RMSE $\downarrow$	Lipo RMSE $\downarrow$	QM7 MAE $\downarrow$
<i>Bond deletion (main setting, 1M molecules)</i>							
MolCLR	0.651 $\pm$ 0.002	0.606 $\pm$ 0.022	0.521 $\pm$ 0.033	1.652 $\pm$ 0.002	3.374 $\pm$ 0.044	1.009 $\pm$ 0.002	76.148 $\pm$ 0.095
Hyp. MolCLR	0.656 $\pm$ 0.001	0.690 $\pm$ 0.002	0.489 $\pm$ 0.004	1.541 $\pm$ 0.001	2.896 $\pm$ 0.036	1.022 $\pm$ 0.004	81.901 $\pm$ 0.972
Scaffold	0.636 $\pm$ 0.002	0.619 $\pm$ 0.014	0.566 $\pm$ 0.006	1.577 $\pm$ 0.002	3.194 $\pm$ 0.011	0.984 $\pm$ 0.001	75.011 $\pm$ 1.117
Hyp. Scaffold	0.611 $\pm$ 0.001	0.742 $\pm$ 0.003	0.568 $\pm$ 0.010	1.682 $\pm$ 0.006	2.787 $\pm$ 0.060	0.992 $\pm$ 0.004	84.965 $\pm$ 3.803
<i>Bond masking (1M molecules)</i>							
MolCLR	0.635 $\pm$ 0.001	0.585 $\pm$ 0.037	0.689 $\pm$ 0.001	1.789 $\pm$ 0.003	5.158 $\pm$ 0.002	1.084 $\pm$ 0.001	150.621 $\pm$ 1.252
Hyp. MolCLR	0.673 $\pm$ 0.008	0.412 $\pm$ 0.010	0.543 $\pm$ 0.006	1.842 $\pm$ 0.003	3.892 $\pm$ 0.117	1.065 $\pm$ 0.012	108.512 $\pm$ 0.950
Scaffold	0.675 $\pm$ 0.001	0.746 $\pm$ 0.017	0.568 $\pm$ 0.002	1.762 $\pm$ 0.002	4.794 $\pm$ 0.011	1.043 $\pm$ 0.002	176.882 $\pm$ 7.592
Hyp. Scaffold	0.665 $\pm$ 0.004	0.599 $\pm$ 0.019	0.649 $\pm$ 0.003	1.807 $\pm$ 0.022	5.454 $\pm$ 0.066	1.077 $\pm$ 0.003	391.899 $\pm$ 14.029

the unbounded hyperbolic variant, with an augmentation strength of 25% and 15%.

The clearest signal concerns optimization stability. At $c = 2$ training is numerically fragile: the total validation loss spikes sharply (to ~ 8.5) early on and drifts upward again in later epochs, and the contrastive accuracy oscillates and degrades over time (falling below 0.7). By contrast, $c = 0.5$ and $c = 1$ train stably, with a smoothly decreasing loss and a monotonically rising accuracy. This rules out $c = 2$; a plausible cause is that the stronger curvature pushes embeddings to radii where the exponential growth of geodesic distance amplifies small perturbations. Between the two stable settings, the pre-training diagnostics actually favour $c = 0.5$: it reaches the highest contrastive accuracy (≈ 0.97 versus ≈ 0.85 for $c = 1$) and the largest similarity gap. This is expected rather than encouraging: a smaller curvature flattens the manifold toward the Euclidean regime, which makes the contrastive task easier, and for precisely that reason we would not expect it to exploit the hierarchical structure the hyperbolic geometry is meant to provide. The remaining hierarchy diagnostics, norm separation and entailment rate, are uninformative in this sweep rather than merely noisy. It predates the bounded formulation, so in all three curvatures the sinh clamp of $\exp_O$ becomes active within the first ten epochs and pins both quantities at exactly zero for the rest of training (Appendix A.4); they therefore cannot separate the settings even in principle. We therefore fix $c = 1$ as the setting that trains stably while retaining a clearly non-Euclidean curvature.

A.13 Molecule-molecule retrieval, full metric set

Table 5 gives every column of the retrieval probe of Section 4.1 for all twelve pre-trained encoders, together with an untrained encoder of the same architecture (random weights, same mean pooling) as a reference for what the inductive bias of the GIN alone provides. The 95% cluster-bootstrap intervals quoted in the main text are computed on this pool with 2,000 resamples of whole scaffold classes. Exact MAP@10 is the objective-proximal check; the four graded columns first delete every candidate carrying the query’s exact BM scaffold and then score the remaining top ten by ECFP4-Tanimoto similarity, discounted by rank and normalized by the best achievable ordering for the nDCG columns. Full-molecule ECFP4 retrieval attains Mol-nDCG@10 = 1 by construction, since it ranks by exactly the similarity that column grades.

A.14 Controlled scaffold identification

This appendix gives the full results of the controlled setting introduced in Section 3.6 and quoted in Section 4.1. Because it is the most direct test in this thesis of whether an embedding space encodes the scaffold *specifically*, we describe it in full.

What one trial looks like. Take a query molecule from the retrieval pool whose BM scaffold is shared by at least one other pool

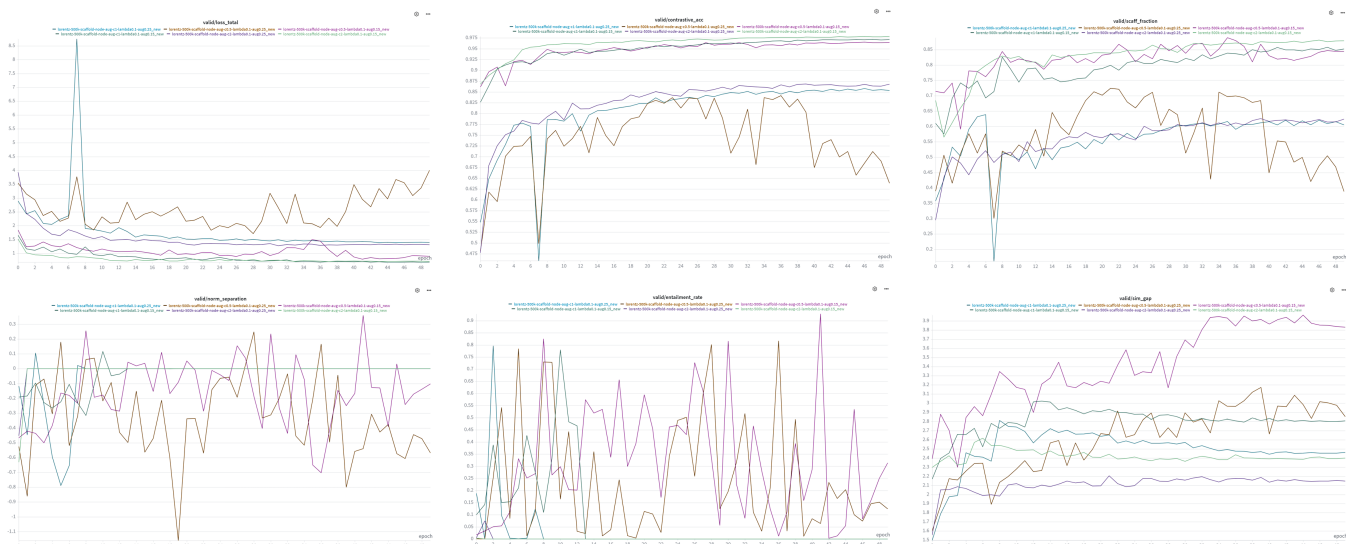


Figure 12: Curvature (c) ablation for the hyperbolic scaffold model, $c \in \{0.5, 1, 2\}$ (at 15% and 25% augmentation). *Top row:* total validation loss, contrastive accuracy, and scaffold-loss fraction. *Bottom row:* the three hyperbolic hierarchy diagnostics, norm separation, entailment rate, and similarity gap. Note the instability of $c = 2$ in the loss and accuracy panels, and that the near-Euclidean $c = 0.5$ attains the highest contrastive accuracy and similarity gap during pre-training, for the reason given in the text.

Table 5: Retrieval in the backbone space h , full metric set. Exact MAP@10 is an objective-proximal check; the graded columns exclude every candidate carrying the query’s exact BM scaffold. Full-molecule ECFP4 retrieval is a deterministic chemical-similarity reference. Boldface marks the best learned model in each column.

Aug.	Model	Exact MAP@10 $\uparrow$	BM-Tan@10 $\uparrow$	BM-nDCG@10 $\uparrow$	Mol-Tan@10 $\uparrow$	Mol-nDCG@10 $\uparrow$
15%	MolCLR (Euclidean)	0.0988	0.3276	0.5111	0.2346	0.6274
15%	Scaffold (Euclidean)	0.1880	0.3715	0.5790	0.2461	0.6565
15%	MolCLR (Lorentz-unbounded)	0.0582	0.2932	0.4534	0.2140	0.5695
15%	Scaffold (Lorentz-unbounded)	0.2564	0.3901	0.6047	0.2471	0.6580
15%	MolCLR (Lorentz-bounded)	0.0694	0.3030	0.4691	0.2185	0.5812
15%	Scaffold (Lorentz-bounded)	0.2627	0.3937	0.6097	0.2479	0.6597
25%	MolCLR (Euclidean)	0.0880	0.3213	0.5012	0.2314	0.6186
25%	Scaffold (Euclidean)	0.1182	0.3386	0.5278	0.2391	0.6380
25%	MolCLR (Lorentz-unbounded)	0.0710	0.3029	0.4746	0.2330	0.6233
25%	Scaffold (Lorentz-unbounded)	0.1542	0.3522	0.5491	0.2471	0.6592
25%	MolCLR (Lorentz-bounded)	0.0763	0.3077	0.4804	0.2328	0.6224
25%	Scaffold (Lorentz-bounded)	0.1308	0.3401	0.5293	0.2358	0.6290
–	Untrained encoder	0.0465	0.2720	0.4260	0.1962	0.5267
–	Full-molecule ECFP4/Tanimoto	0.2424	0.3991	0.6233	0.3665	1.0000

molecule. Build a candidate set of exactly ten molecules: one *positive*, drawn at random from the other molecules that carry the query’s exact scaffold, and nine *distractors*, all of which carry a different scaffold. Rank the ten candidates by cosine similarity to the query in the frozen backbone space h . The trial is a success if the positive ranks first. Accuracy is the fraction of successful trials over all 18,000 queries. Because every candidate set has exactly one correct answer among ten, a retriever that ranks at random succeeds 10% of the time, so chance is exactly 10% for every model and does not depend on how the pool was built. The candidate sets are drawn once with a fixed seed and are identical for every retriever, so the comparison is paired: two models are scored on exactly the same questions.

Random versus hard distractors. The nine distractors are drawn in one of two ways, and the difference between the two is the point of the experiment.

- **Random distractors** are sampled uniformly from the pool, subject only to having a different scaffold from the query. Most of them are chemically unrelated to the query: different size, different rings, different functional groups. A model can rank the positive first simply by encoding that it *resembles the query at all*, since the positive shares the query’s core and typically much of its periphery, while the random distractors share little. This column therefore mostly measures generic whole-molecule similarity.
- **Hard distractors** are sampled from the 200 molecules that are most similar to the query by full-molecule ECFP4–Tanimoto

Table 6: Controlled scaffold identification. For each query, one candidate shares its BM scaffold and nine do not; the table gives the percentage of queries for which the same-scaffold candidate is ranked first by cosine similarity in h . Random distractors are drawn uniformly from the pool; hard distractors are the query’s ECFP4-nearest molecules with a different scaffold. Chance is 10% in both columns. The untrained encoder has random weights; the ECFP4 row under hard distractors is scored against candidates chosen adversarially to it. Boldface marks the best learned model per column.

Aug.	Model	Random distractors $\uparrow$	Hard distractors $\uparrow$
15%	MolCLR (Euclidean)	72.6%	34.7%
15%	Scaffold (Euclidean)	84.8%	48.2%
15%	MolCLR (Lorentz-unbounded)	67.5%	30.1%
15%	Scaffold (Lorentz-unbounded)	92.8%	59.1%
15%	MolCLR (Lorentz-bounded)	70.1%	31.9%
15%	Scaffold (Lorentz-bounded)	93.4%	60.5%
25%	MolCLR (Euclidean)	76.4%	35.2%
25%	Scaffold (Euclidean)	77.8%	37.7%
25%	MolCLR (Lorentz-unbounded)	60.6%	24.4%
25%	Scaffold (Lorentz-unbounded)	82.4%	42.2%
25%	MolCLR (Lorentz-bounded)	65.0%	27.1%
25%	Scaffold (Lorentz-bounded)	79.5%	39.8%
–	Untrained encoder	61.0%	28.6%
–	Full-molecule ECFP4/Tanimoto	86.1%	38.8%
–	Chance	10.0%	10.0%

but that carry a *different* BM scaffold. They look like the query: similar size, similar substituents, similar local atom environments, only built on a different core. Whole-molecule fingerprint similarity now points at the distractors as strongly as at the positive, so ranking the positive first requires encoding something beyond it, in particular the scaffold. This column measures scaffold-specific organization.

The *drop* from the random column to the hard column is the diagnostic. A model whose neighbourhoods are organized by whole-molecule resemblance loses most of its margin over chance when the distractors are made to resemble the query; a model whose neighbourhoods are organized by scaffold keeps more of it.

Reference rows. Two model-free references are scored on the same candidate sets. The *untrained encoder* is the GIN backbone with random weights and the same mean pooling: it shows how much of any score is due to the architecture’s inductive bias rather than to pre-training. The *ECFP4* row ranks candidates by Tanimoto similarity of full-molecule fingerprints. Under hard distractors this row must be read with care: the distractors were selected as ECFP4’s own nearest neighbours, so they are adversarial to it by construction and its drop overstates how much a fingerprint degrades on such candidates. This does not affect comparisons between the learned models, which all face the same candidates.

Reading the table. Table 6 reports both columns for all twelve encoders. Three observations follow. First, the random column is close to saturated and separates the models poorly: every learned model scores 61–93%, and the untrained encoder already reaches 61%, so whole-molecule resemblance is largely available without training. Second, the hard column spreads the models out. The six MolCLR baselines fall to 24–35%, within a few points of the untrained encoder’s 29%, meaning that ordinary contrastive pre-training adds little scaffold-specific organization beyond what the untrained architecture already provides. The scaffold-supervised models retain far more: 38–42% at 25% augmentation and 48–60% at 15%. Third, within the scaffold-supervised models the ordering matches the graded retrieval of Table 1: the two Lorentz variants at

15% are best (59–60%), Euclidean at 15% next (48%), and all three are lower at 25%.

Statistical support. Queries that share a scaffold see nearly the same candidates, so they are not independent observations. All intervals are therefore 95% percentile intervals from 2,000 bootstrap resamples that draw whole scaffold classes rather than individual queries. The six paired scaffold-minus-MolCLR differences in hard-distractor accuracy are +0.135 [+0.129, +0.141] (Euclidean, 15%), +0.290 [+0.281, +0.299] (Lorentz-unbounded, 15%), +0.286 [+0.277, +0.295] (Lorentz-bounded, 15%), +0.025 [+0.020, +0.030] (Euclidean, 25%), +0.178 [+0.171, +0.185] (Lorentz-unbounded, 25%) and +0.127 [+0.120, +0.134] (Lorentz-bounded, 25%); all exclude zero. The same protocol was run at $k \in \{1, 4, 19, 49, 99\}$ distractors with the same ordering of models at every k ; we report $k = 9$ as the representative setting.

A.15 Molecule–scaffold ranking

This is the secondary molecule–scaffold diagnostic: for each molecule we rank all unique scaffolds in the batch by their similarity to the molecule’s embedding and record the rank of its own scaffold. Both models compared here are trained with the scaffold-alignment loss, and molecule–scaffold alignment is the relation that loss states directly, so this table measures how faithfully each geometry satisfies the objective it was given rather than establishing emergent structure; the molecule–molecule retrieval results are the substantive measurement. We evaluate over 500 validation batches, which gives an average pool of about 435 unique scaffolds per batch, so that a random ranking corresponds to $\text{MRR} \approx 0.002$ and a mean rank of $\approx 50\%$ of the pool. Both models are taken at 25% augmentation and the hyperbolic model is the unbounded Lorentz variant; Table 7 reports the ranking obtained from the original (un-augmented) molecule embedding.

Both models sit far above the random baseline ($\text{MRR} \approx 0.002$), so neither has collapsed and both encode scaffold membership to a meaningful degree. What separates them is concentrated at the very top of the ranking. The unbounded hyperbolic model attains the better MRR (0.599 against 0.528), ranks the correct scaffold

Table 7: Molecule–scaffold ranking from the original-molecule embedding, over 500 validation batches (pool $\approx$ 435 unique scaffolds). For each molecule we rank all pool scaffolds by similarity and record the rank of its own scaffold. MRR and Hits@ k are higher-is-better; median rank and mean rank (as a percentage of pool size) are lower-is-better. A random ranking gives MRR $\approx$ 0.002 and mean rank $\approx$ 50%. Both models are trained with the scaffold-alignment loss at 25% augmentation; the hyperbolic model is the unbounded Lorentz variant.

Model	MRR $\uparrow$	Hits@1 $\uparrow$	Hits@5 $\uparrow$	Hits@10 $\uparrow$	Med. rank $\downarrow$	Mean rank% $\downarrow$
Euclid. Scaffold	0.528	0.433	0.630	0.723	2	5.49
Hyp. Scaffold (unbounded)	0.599	0.532	0.674	0.724	1	11.65

first for over half of all molecules (Hits@1 = 0.532 against 0.433), and reduces the median rank from 2 to 1. That advantage narrows monotonically as k grows: the gap in Hits@ k falls from 9.9 points at $k = 1$ to 4.4 points at $k = 5$ and to 0.1 points at $k = 10$, where the two are effectively tied (0.724 against 0.723). What the unbounded Lorentz objective buys is therefore a sharpening of the top of the ranking rather than a broad improvement in how often the correct scaffold is recovered at all.

The mean rank inverts this ordering, and the inversion is informative rather than contradictory. Despite its better median and top- k rates, the unbounded hyperbolic model has a mean rank of 11.65% of the pool against 5.49% for the Euclidean model, roughly a factor of two worse. Both remain far below the 50% of a random ranking, so this is not a failure of the objective; it signals a heavy-tailed rank distribution. The hyperbolic model ranks the typical molecule’s own scaffold first, but a minority of molecules are ranked very far from their scaffold, and these outliers dominate the mean. We do not attribute this to the exponential volume growth of hyperbolic space: the model compared here is the unbounded variant, whose radial coordinate has collapsed to a single value (Appendix A.4), so no point can sit at a disproportionate hyperbolic radius. The tail is instead a property of how the collapsed angular objective distributes its errors, and we leave its cause open. The Euclidean model is the more balanced of the two: a strong median (2) together with the lowest mean rank.

These ranking results refine rather than repeat the molecule–molecule retrieval comparison: under the scaffold-alignment loss the two geometries reach comparable coverage by $k = 10$ but distribute their errors differently, the Euclidean form most consistently across the whole pool and the unbounded hyperbolic form most sharply at the top of the ranking, at the cost of a heavier tail.