POLYBC: A GRAPH-AUGMENTED CHEMICAL LANGUAGE MODEL FOR COMPLEX COPOLYMER LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Copolymers are macromolecules assembled from repeating structural units, arranged in random, alternating, or block architectures. Predicting and designing copolymer properties from molecular structure is of broad interest across energy storage, construction, medicine, and aerospace, yet remains fundamentally challenging because (1) there is no effective parameterization for the structural units or for the complex architectures connecting them, and (2) experimental measurements for any particular application are typically scarce and unbalanced. To resolve these challenges, we propose PolyBC, a physics-guided, graph-augmented chemical language model. In particular, we introduce a graph neural network over structural units at two levels of granularity — repeat units and sub-monomer fragments — whose nodes are encoded by Poly4mer, a chemical language model that supplies transferable latent embeddings of homopolymer segments. At either resolution, the GNN reasons over the polymer topology and closes the finite chain with architecture-matched boundary conditions: periodic for alternating copolymers, Neumann for block copolymers, and mean-field for random copolymers. This construction imposes a theoretically guaranteed, consistent bulk physics: training on one copolymer type and boundary condition generalizes to the others, and predictions are invariant to how many repeats are drawn. On both synthetic and experimental benchmarks, our model is repetition-invariant, data-efficient, and transferable across copolymer types at both granularities, outperforming state-of-the-art GNN and LLM baselines on prediction and design tasks.

1 INTRODUCTION

Polymers are built from many copies of a few structural units. A copolymer combines two or more monomers, and the *same* monomers at the *same* composition can be arranged as alternating, block, or random chains whose properties differ markedly: block copolymers microphase-separate, whereas random copolymers mix at the segment level (Bates & Fredrickson, 1990). While polymer informatics learns such structure-property relationships from data to accelerate materials discovery (Ramprasad et al., 2017; Chen et al., 2021), two obstacles make copolymers modeling much harder than small molecules:

Lack of generalizable parameterization. A copolymer is not a single molecule but an ensemble of long chains, and any finite graph or string drawn from it is an artifact of how many repeat units R were serialized. Atom-level graphs grow linearly in R, and standard message-passing networks produce predictions that drift with R (Zhu et al., 2025). Chemical language models (LMs) that read a copolymer as one string face the opposite problem: a single-repeat string barely encodes architecture, because for the 28.6% of rows in the benchmark of Aldeghi & Coley (2022) that are alternating or block at f_A=0.5, the R=1 monomer sequence is AB in both architectures, while a 20-repeat string (median ≈1,600 characters) exceeds the 512-token context of polyBERT and TransPolymer (Kuenneth & Ramprasad, 2023; Xu et al., 2023).

Experimental measurements are scarce and unbalanced. Compared with monomers and homopolymers, copolymers have longer and more complex molecular structures, which require larger AI/ML models to capture. Larger models, in turn, need more training data, yet experimental measurements for copolymers are often scarce. Moreover, the available data are unevenly distributed

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

across copolymer types, leaving some types with even fewer measurements. Taking the glass-transition records in the PoLyInfo dataset (Otsuka et al., 2011) for instance: Here's the polished version, with your LaTeX kept: Copolymers have only $\sim 1/4$ as many records as homopolymers. Among the 2,976 binary copolymer records with a measured architecture label, only 427 are block copolymers, and only 258 of these fall in the training split (see Appendix B). Such limited data can lead to severe overfitting when training copolymer models.

Existing methods address pieces of the problem. Atom-level GNNs learn repetition invariance from repeat-unit augmentation (Zhu et al., 2025) or target shifts in graph size and environment (Wu et al., 2022; Liu et al., 2022; Chen et al., 2022; Buffelli et al., 2022; Huang et al., 2024) rather than the pattern of node types that distinguishes architectures; pretrained chemical LMs (Ross et al., 2022; Kuenneth & Ramprasad, 2023; Xu et al., 2023; Yin et al., 2025) transfer chemistry but read a copolymer as one string. None keeps the chemistry of the structural units fixed while letting the architecture change only how the chain is closed, which is what zero-shot transfer across architectures requires.

We make that separation explicit: a chemical LM supplies the chemistry of each structural unit, and a GNN over the units supplies their arrangement (Figure 1). Nodes are either whole monomers or ring/linker fragments of a monomer, and each node feature is a fixed, differentiable, label-free map of the frozen Poly4mer latent (Yin et al., 2025). Message passing on the unit chain is read as a forward-Euler discretization of a nonlocal operator whose horizon is one node; in this view the finite chain is a computational window, and its closure is a *boundary condition* in the sense of nonlocal volume constraints (Du et al., 2012). We propose to alter boundary conditions from different copolymer types: periodic for alternating, Neumann (zero-flux) for block, and mean-field for random chains, so that the copolymer architecture enters through how the window is closed rather than through learned embeddings. **Our key contributions:**

(i) We introduce PolyBC, a graph-augmented chemical LM for copolymers that operates at both monomer and fragment granularity. Its node features depend solely on LM latents, which makes the property model differentiable with respect to the unit latents, as inverse design requires.

(ii) We formulate copolymer architecture as a boundary condition of a nonlocal message-passing operator and provide theoretical guarantees of repetition invariance and generalization across copolymer types (Section 3.5). Specifically, we show that for alternating and block copolymers, the predicted property is invariant to the number of repeat units used to represent the same copolymer. For random copolymers, this invariance holds asymptotically as the number of repeat units grows. We further prove that all three copolymer types (alternating, block, and random) can be parameterized by a shared interior operator, which enables zero-shot transfer across types.

(iii) We evaluate PolyBC on two benchmark datasets: *Coley*, a synthetic dataset of electron affinity (EA) and ionization potential (IP) properties (Aldeghi & Coley, 2022); and *PoLyInfo*, an experimental glass-transition temperature dataset (Otsuka et al., 2011). On both datasets, we cover all three copolymer types and compare against 12 GNN and chemical-LM baselines. The results confirm our theoretical guarantees. Trained only on single-repeat chains, PolyBC predicts properties both for the copolymer type seen in training and for other types, without additional tuning. Moreover, on test copolymers with $20\times$ repeat units, PolyBC remains stable while most baselines collapse.

2 Related Work

Physics-guided learning and nonlocal operators. Physics-informed learning builds known structure into models to extrapolate with less data (Karniadakis et al., 2021). Neural operators, nonlocal kernel networks, and PDE-inspired GNNs interpret layers as integral operators or diffusion (Li et al., 2020; Lu et al., 2021; Kovachki et al., 2023; You et al., 2022; Chamberlain et al., 2021; Eliasof et al., 2021). In nonlocal calculus and peridynamics, boundary conditions are *volume constraints* on a collar outside the domain (Silling, 2000; Du et al., 2012; 2013). We use volume constraints to encode copolymer architecture, obtaining repetition invariance from the boundary condition rather than from augmentation or algorithmic alignment (Xu et al., 2020; Veličković & Blundell, 2021).

Graph neural networks for polymers. Polymer GNNs operate on monomer, repeat-unit, or periodic graphs (St. John et al., 2019; Park et al., 2022; Gurnani et al., 2023; Antoniuk et al., 2022); copolymer models add composition or sequence descriptors (Kuenneth et al., 2021; Patel et al., 2022) or weight the bonds of a stochastic monomer ensemble (Aldeghi & Coley, 2022). GRIN

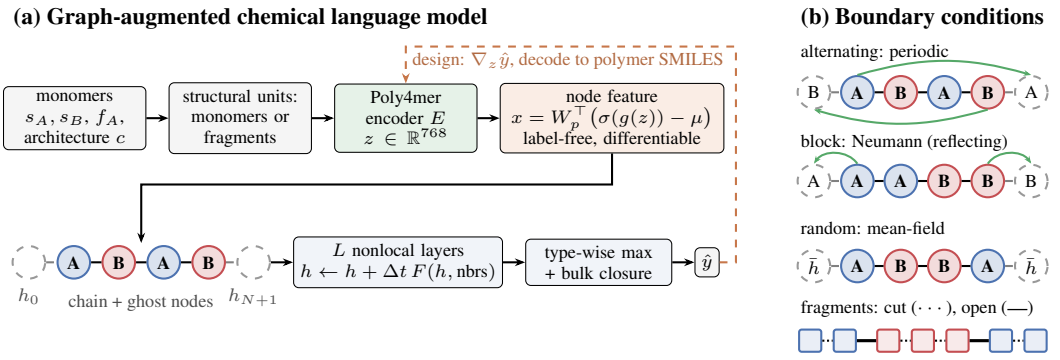

Figure 1: **Overview.** (a) Each structural unit, a monomer or a ring/linker fragment of a monomer, is encoded by the frozen Poly4mer encoder; a label-free distillation head g and a fixed PCA map the latent z to the node feature x. A nonlocal message-passing operator acts on the unit chain, whose two ends are closed by ghost nodes h_0, h_{N+1}, and a type-wise max readout produces the property. Because x is a differentiable function of z, gradients reach the latent for inverse design. (b) Architecture enters only through the type sequence and the ghost values (periodic images for alternating chains, reflection for block chains, and the chain mean $\bar{h} = \frac{1}{N}\sum_j h_j$ for random chains), so one set of weights serves every architecture. Fragment edges carry an open/cut bit that separates bonds between monomer occurrences from intra-monomer cuts.

learns representations invariant to the number of repeat units through max aggregation, maximum-spanning-tree alignment, and repeat-unit augmentation (Zhu et al., 2025). Invariant-rationale and size-generalization GNNs (Arjovsky et al., 2019; Wu et al., 2022; Liu et al., 2022; Chen et al., 2022; Miao et al., 2022; Buffelli et al., 2022; Murphy et al., 2019; Huang et al., 2024; Yehudai et al., 2021; Bevilacqua et al., 2021) address shifts in graph size or environment, not in the sequence of node types. Motif-based graphs coarsen molecules into substructures (Degen et al., 2008; Jin et al., 2020; Zhang et al., 2021); our fragment chain is such a coarsening with LM-embedded substructures.

Chemical language models. SMILES and polymer LMs (Chithrananda et al., 2020; Irwin et al., 2022; Ross et al., 2022; Soares et al., 2025; Kuenneth & Ramprasad, 2023; Xu et al., 2023; Liu et al., 2025) provide transferable embeddings, graph transformers attend over molecular graphs (Ying et al., 2021), graph-language models couple graph encoders with LMs (Zhao et al., 2023; Liu et al., 2023), and Poly4mer (Yin et al., 2025) adds polymer tokens and an autoencoding latent that supports generation. We use the LM to encode structural units rather than the whole copolymer, which removes the context limit, lets a graph carry the architecture, and keeps the property model differentiable in the LM latents for inverse design (Gómez-Bombarelli et al., 2018; Jin et al., 2018; Kadurin et al., 2017; Batra et al., 2020; Gurnani et al., 2021).

3 Methodology

3.1 Problem Definition

We present PolyBC, a graph-augmented chemical LM with a monomer-level (PolyBC-M) and a fragment-level (PolyBC-F) variant. A binary copolymer is $P = (s_A, s_B, f_A, c)$ with monomers s_A, s_B given as polymer SMILES (PSMILES) whose attachment points are marked by $*$, molar fraction $f_A \in [0, 1]$ of A, and architecture $c \in \{\text{alt}, \text{blk}, \text{ran}\}$. Let $n_A/(n_A{+}n_B)$ be the closest fraction to f_A with $n_A{+}n_B \leq 20$. The *realization* of P with R repeat units is a type sequence $w_R^c \in \{A, B\}^{(n_A+n_B)R}$:

$$w_R^{\text{alt}} = (AB)^{n_A R}\ (n_A{=}n_B), \qquad w_R^{\text{blk}} = A^{n_A R} B^{n_B R}, \qquad w_R^{\text{ran}} \in \text{Arr}(n_A R, n_B R), \quad (1)$$

where $\text{Arr}(a, b)$ is the set of arrangements of a A's and b B's, from which w_R^{ran} is chosen per copolymer by a seeded rule and redrawn for every R (Appendix B, also for the soft-alternating sequence of PoLyInfo alternating records with $n_A \neq n_B$). All realizations share one label y. We seek f_θ, trained on $R{=}1$ only, with $f_\theta(w_R^c) \approx y$ for all R (*repetition invariance*), accurate on $c' \neq c$ when trained on c (*architecture transfer*), and differentiable in the unit latents (*inverse design*).

3.2 Structural Units and Latent Node Features

Two granularities. In PolyBC-M every monomer occurrence in w_R^c is one node. In PolyBC-F every monomer is decomposed into ring/linker fragments: fused ring systems and acyclic linker components form groups, all bonds between groups are cut, and each cut end is capped with $*$ so that every fragment is itself a valid PSMILES. Each occurrence is expanded into its fragments in a fixed, deterministic order (by lowest atom index in the input PSMILES), which requires no backbone tracing and applies to any valid PSMILES, and an open/cut edge bit marks where one occurrence ends and the next begins. The 683 Coley and 956 PoLyInfo monomers map onto 307 and 739 fragments (2.41 and 4.90 per monomer); a few fragments are shared widely (`*c1ccc(*)cc1` occurs in 435 PoLyInfo monomers), whereas 70% of PoLyInfo fragments occur in a single monomer (Appendix B).

Latent node features. Let E denote the frozen Poly4mer encoder (Yin et al., 2025), which maps a PSMILES u to $z = E(u) \in \mathbb{R}^{768}$ through a SMI-TED backbone (Soares et al., 2025) and an autoencoder bottleneck (Appendix C). Raw latents are poorly conditioned under repetition, and on fragment nodes and PoLyInfo even their principal components carry little property signal (Section 4.5). We therefore re-express z through a label-free distillation head $g : \mathbb{R}^{768} \to \mathbb{R}^{2048}$ (MLP $768 \to 1024 \to 1024 \to 2048$, GELU) trained with binary cross-entropy to predict the Morgan fingerprint (radius 2) (Rogers & Hahn, 2010) of the unit, followed by a fixed PCA:

$$x(u) \;=\; W_p^\top \big(\sigma(g(E(u))) - \mu\big) \in \mathbb{R}^p, \tag{2}$$

with p=512 on Coley and p=256 on PoLyInfo. Neither g nor the PCA sees a property label. For PolyBC-M on Coley the node unit is the monomer; on PoLyInfo a monomer's node feature is the mean of its fragment features, which places both granularities on the same basis. Because Eq. (2) composes fixed differentiable maps, f_θ is differentiable in the latent of every node unit (monomer latents for PolyBC-M on Coley; fragment latents otherwise), and no node feature is a fingerprint or quantum-chemical descriptor that would stay frozen while z moves during design.

3.3 Message Passing as a Nonlocal Operator

Order the N nodes of a realization by monomer occurrence along the chain and, within an occurrence, by this fixed fragment order, $i = 1, \dots, N$, with unit type $\tau(i) \in \{A, B\}$ and occurrence index $\omega(i)$. Node i exchanges messages with its left and right neighbors through edge features $e_i^- = [-1, o_i^-]$ and $e_i^+ = [+1, o_i^+]$: the sign encodes direction, and the open bit $o_i^- = \mathbf{1}[i{=}1 \;\vee\; \omega(i{-}1) \neq \omega(i)]$ (analogously o_i^+) separates *open* bonds between consecutive monomer occurrences, including the ghost bonds below, from *cut* bonds inside a monomer. On monomer graphs every bond is open and the bit is dropped. With $h_i^{(0)} = \mathrm{ELU}(\mathrm{LN}_0(W_{\mathrm{in}} x_i + b_{\mathrm{in}}))$ (layer normalization LN; dropout omitted), layer $k = 1, \dots, L$ updates

$$m_i^{(k)} = \textstyle\sum_{\pm} \phi_k\big([h_{i\pm1}^{(k-1)} \| e_i^{\pm}]\big), \quad h_i^{(k)} = h_i^{(k-1)} + \Delta t\, \mathrm{ELU}\big(\mathrm{LN}_k(\psi_k([h_i^{(k-1)} \| m_i^{(k)}]))\big), \tag{3}$$

where ϕ_k and ψ_k are Linear-LN-ELU-Linear maps shared by all nodes and all architectures and $\|$ denotes concatenation. Assigning a pseudo-grid $\xi_i = i\Delta\xi$ and pseudo-time $t_k = k\Delta t$, Eq. (3) is the forward-Euler discretization of

$$\partial_t h(\xi, t) \;=\; F_t\Big(h(\xi,t),\; \int_{B_\delta(\xi)\setminus\{\xi\}} \phi_t\big(h(\eta,t),\, e(\xi,\eta)\big)\, d\nu(\eta)\Big), \qquad \delta = \Delta\xi, \tag{4}$$

a nonlocal evolution with horizon one node, counting measure ν on the grid, and layer-specific F_t, ϕ_t (Du et al., 2012; You et al., 2022). For ξ_1 and ξ_N the ball B_δ reaches the interaction collar $\{\xi_0, \xi_{N+1}\}$ outside the chain, where h must be prescribed; this is where architecture enters.

3.4 Architecture as a Boundary Condition

We prescribe the collar values $h_0^{(k)}, h_{N+1}^{(k)}$ (*ghost nodes*) at every layer from the current states, according to the architecture of the chain:

$$\textbf{alternating (periodic):} \quad h_0 = h_N, \qquad h_{N+1} = h_1; \tag{5}$$

$$\textbf{block (Neumann):} \quad h_0 = h_1, \qquad h_{N+1} = h_N \quad \big(\text{zero flux at both ends}\big); \tag{6}$$

$$\textbf{random (mean-field):} \quad h_0 = h_{N+1} = \bar{h} := \tfrac{1}{N}\textstyle\sum_{j=1}^{N} h_j. \tag{7}$$

A periodic boundary condition continues the $ABAB$ tiling of an alternating chain; a Neumann boundary condition lets each chain end see a copy of itself, reproducing exactly the environment inside a long homogeneous run on monomer graphs; and a mean-field boundary condition, in the spirit of mean-field theory for disordered systems, supplies the composition-typical environment of a random chain. Ghost bonds are open. No architecture embedding enters Eq. (3): the architecture changes only the sequence (1) and the ghost rule, so a model trained under one boundary condition is applied to another by rebuilding the test graph with its own boundary condition.

Readout and bulk closure. Let $M_t = \max_{i:\tau(i)=t} h_i^{(L)}$ be the element-wise maximum over nodes of type t ($M_t{:=}0$ for an absent type under a bulk closure; without one, the second branch below is the maximum over all nodes), and let α be the composition weight of A: the fraction of A nodes on Coley (a fragment-count ratio on fragment graphs) and the molar fraction f_A on PoLyInfo, where the output is additionally rescaled by the training-label mean and standard deviation. The prediction is

$$\hat{y} = \mathrm{MLP}\big([\,\alpha M_A + (1-\alpha)M_B \parallel \max(M_A, M_B)\,]\big). \tag{8}$$

A single-repeat training chain can lack the interior of a homogeneous run (both runs of AB, the B-run of $AAAB$), which appears only at larger R. An optional *bulk closure* supplies it: for each type t we propagate a phantom node carrying the mean input feature of type t through the same L layers, across open bonds, with both neighbors equal to itself (Neumann bulk) or to the composition-weighted mixture of the two phantoms (mean-field bulk), and replace $M_t \leftarrow \max(M_t, b_t^{(L)})$ for each type present. On monomer graphs at $L{=}1$ the Neumann phantom is exactly the interior state of a long block (Proposition 1b); on fragment graphs it is a mean-fragment pseudo-monomer and the closure is approximate. On PoLyInfo we match the phantom to the training architecture (mean-field for random, none for alternating, whose periodic boundary condition supplies the bulk of 1:1 chains, Neumann for block); on Coley we use the Neumann bulk throughout. A *canonical chain* closure, used only in ablations, replaces the realization by a fixed chain $A^{L+1}B^{L+1}$ of type-mean features and so ignores the sequence; the design of Section 4.6 uses the Neumann bulk closure of the reported Coley model, not this one. Max-type readouts are duplication invariant; mean readouts would drift as $c_0 + c_1/R$.

3.5 Theoretical Guarantees

Define the depth-k environment of node i recursively by $\mathcal{E}_0(i) = x_i$ and $\mathcal{E}_k(i) = \big(\mathcal{E}_{k-1}(i), o_i^-, o_i^+, \mathcal{E}_{k-1}(\ell(i)), \mathcal{E}_{k-1}(r(i))\big)$ (the o bits are omitted on monomer graphs), where $\ell(i), r(i)$ are the left and right neighbors of i after applying the boundary condition (periodic: $\ell(1) = N$; Neumann: $\ell(1) = 1$). For monomer graphs at $L{=}1$ we write $(t; a, b)$ for a node of type t with left neighbor of type a and right neighbor of type b. Proofs are in Appendix A.

Lemma 1 (Locality)**.** *Under the periodic or Neumann boundary condition there is a function Φ_L, determined by θ alone, with $h_i^{(L)} = \Phi_L(\mathcal{E}_L(i))$ for every chain and repeat number; under any boundary condition the same holds for every node with $L < i \le N - L$, whose recursion never reaches a ghost point. Hence an L-layer model resolves the order of at most $2L{+}1$ consecutive nodes in a single node state: with $L{=}1$ a node encodes a three-node window, whereas PoLyInfo monomers have 4.9 fragments on average (up to 28), so longer within-monomer order reaches the prediction only through the readout.*

Theorem 1 (Exact repetition invariance under periodic boundary condition)**.** *Let u be the node word of one repeat unit (monomers, or fragments after expansion) and G_R the chain of u^R with the periodic boundary condition (5). For every θ, depth L, and $R \ge 1$ (in evaluation mode), $h_i^{(L)}(G_R) = h_{\pi(i)}^{(L)}(G_1)$ with $\pi(i) = 1 + ((i-1) \bmod |u|)$, and consequently $f_\theta(G_R) = f_\theta(G_1)$, with or without bulk closure.*

Theorem 1 covers every 1:1 alternating copolymer at both granularities and needs no max aggregation: the boundary condition gives every node, including the chain ends, the two neighbors it has in the infinite chain. Soft-alternating PoLyInfo records with $n_A \neq n_B$ are not tilings; for them the stability of accuracy under repetition is established empirically (Section 4.4). Block chains are not tilings either, but their environment set stabilizes:

Proposition 1 (Block chains under Neumann boundary condition)**.** *Let G_R be the block chain $A^{n_A R}B^{n_B R}$ with $n_A, n_B \ge 1$ and boundary condition (6). (a) For monomer or fragment*

nodes and any L, the set of node states of each type, and hence $f_\theta(G_R)$, is identical for all $R \geq R_0 := \lceil (2L+1)/\min(n_A, n_B) \rceil$. (b) For monomer nodes, $L = 1$, and the Neumann bulk closure, $f_\theta(G_R) = f_\theta(G_1)$ for all $R \geq 1$.

Proposition 2 (Random chains under mean-field boundary condition)**.** *For monomer or fragment nodes and $L = 1$, the mean-field ghost $\bar{h}$ is independent of R, and for $n_A, n_B \geq 1$ every interior environment occurs in w_R^{ran} with probability tending to one as $R \to \infty$. Asymptotically, the prediction therefore depends on R only through the environments of the two chain ends.*

Proposition 3 (Shared interior operator)**.** *For architectures c, c', parameters θ, and nodes $i \in G^c$, $j \in G^{c'}$ with $L < i \leq N^c - L$ and $L < j \leq N^{c'} - L$ whose depth-L environments coincide, $h_i^{(L)} = h_j^{(L)}$; if neither chain uses the mean-field boundary condition, the position condition can be dropped.*

Corollary 1 (Environment generalization)**.** *For monomer graphs at L=1, alternating chains realize only the environments $(A; B, B)$ and $(B; A, A)$, block chains a subset of $\{(A; A, A), (A; A, B), (B; A, B), (B; B, B)\}$, and random chains with $n_A, n_B \geq 1$ all eight interior environments with probability $\to 1$ as $R \to \infty$. Single-repeat random chains with $f_A \neq \frac{1}{2}$ can already contain alternating or block environments (e.g. $(A; B, B)$ in $BBAB$, $(A; A, A)$ in $BAAA$).*

Proposition 3 guarantees that every environment shared with the training chains is processed exactly as in training, which makes zero-shot transfer well defined. Unseen environments are handled by the same shared maps ϕ_k, ψ_k, whose generalization Section 4 confirms empirically. For 1:1 alternating chains, as on Coley dataset we tested below, Corollary 1 predicts that transfers between alternating and block chains, whose chain environments are disjoint, are the hardest. It does not apply to PoLyInfo, where 263 of 330 alternating test records contain block environments. Propositions 1 and 2 also predict weaker invariance for deeper models, fragment graphs, and random chains.

4 Experiments

We ask five questions. **Q1 (Effectiveness):** is PolyBC more accurate than GNN and chemical-LM baselines on the architecture it was trained on? **Q2 (Architecture transfer):** does a model trained on one architecture transfer to the others? **Q3 (Small data):** how does accuracy degrade as labels become scarce? **Q4 (Repetition invariance):** does accuracy survive a $20\times$ increase of the repeat number without multi-repeat training? **Q5 (Design):** can the latent-only model drive inverse design? Training cost is reported in Appendix H.

4.1 Experimental Settings

Datasets. *Coley* (Table 3) contains 42,966 copolymers of 9 A-monomers and 682 B-monomers in three architectures and three compositions ($f_A \in \{0.25, 0.5, 0.75\}$; alternating only at 1:1) with xTB-computed, DFT-calibrated EA and IP (Aldeghi & Coley, 2022; Bannwarth et al., 2019). We use the full set, $14\times$ larger than the 3,000-graph subsample of prior work (Zhu et al., 2025), split pair-level 60/10/30 (seed 42), so that Coley measures generalization to unseen monomer pairs; unseen monomers are tested on PoLyInfo. *PoLyInfo* contains experimental T_g records of binary copolymers with a measured architecture label (Otsuka et al., 2011), split record-disjointly (only 3 unordered monomer pairs recur, reversed, in 7 test rows); This makes PoLyInfo a test of generalization to new chemistry: 210 of the 333 test monomers (63%) never appear in a labeled training row. The feature maps use no labels and are therefore a label-free setting in which no test label is ever seen (Appendix C). Most PoLyInfo alternating records (263 of 330 alternating test rows) are alternating-tendency condensation copolymers with $n_A \neq n_B$; we represent 254 of them faithfully as soft-alternating sequences (Appendix B), so this split tests the model on realistic rather than idealized alternation. *PoLyInfo-Small* keeps a per-architecture 12.5% subsample of the training rows (n_{train}=460, of which 230 are copolymers) with identical validation and test rows.

Protocol. For each training architecture c we train on R=1 realizations of c, select the checkpoint on validation rows of c, and evaluate it on test rows of all three architectures, rebuilding each test graph with its own architecture and boundary condition; rows of the 3×3 matrix are training and columns test architectures. We report means over the three diagonal (*in-domain*) and six off-diagonal (*transfer*) cells; full matrices with standard deviations are in Appendix D. Test graphs use $R \in \{1, 5, 10, 20\}$ for our models on Coley and PoLyInfo and for all baselines (Test$R$).

Table 1: **Coley EA and IP, 3×3 cross-architecture results.** Test R^2 averaged over the three in-domain and six transfer cells at R=1 (Test1) and R=20 (Test20); mean over 5 seeds. ‡Trained on $R \in \{1, 3\}$; all others on R=1. †Input independent of R. "<0": negative mean; "n/a": not run. Bold: best per column. Per-cell results: Tables 7 and 8.

	EA				IP			
	In-domain		Transfer		In-domain		Transfer	
Model	Test1	Test20	Test1	Test20	Test1	Test20	Test1	Test20
Atom-level graph models								
GRIN (Zhu et al., 2025)	$0.9542_{\pm.0164}$	$0.9367_{\pm.0217}$	$0.8930_{\pm.0505}$	$0.8770_{\pm.0439}$	$0.9611_{\pm.0297}$	$0.9581_{\pm.0314}$	$0.8738_{\pm.0610}$	$0.8676_{\pm.0577}$
CIGA (Chen et al., 2022)	$0.9065_{\pm.0109}$	$0.8383_{\pm.0458}$	$0.8558_{\pm.0158}$	$0.8255_{\pm.0352}$	$0.9156_{\pm.0127}$	$0.7746_{\pm.0787}$	$0.8416_{\pm.0211}$	$0.7835_{\pm.0673}$
DISGEN (Huang et al., 2024)	$0.8678_{\pm.0096}$	$0.8008_{\pm.0342}$	$0.8208_{\pm.0108}$	$0.7716_{\pm.0276}$	$0.8090_{\pm.0215}$	$0.5268_{\pm.0868}$	$0.7338_{\pm.0273}$	$0.5791_{\pm.0709}$
GREA (Liu et al., 2022)	$0.9462_{\pm.0046}$	<0	$0.7352_{\pm.1056}$	<0	$0.9494_{\pm.0063}$	<0	$0.5784_{\pm.4094}$	<0
SSR (Buffelli et al., 2022)	$0.8952_{\pm.0197}$	<0	$0.4892_{\pm.2708}$	<0	$0.9494_{\pm.0057}$	<0	$0.7015_{\pm.1439}$	<0
GSAT (Miao et al., 2022)	$0.8920_{\pm.0012}$	<0	<0	<0	$0.9240_{\pm.0035}$	<0	$0.1634_{\pm.1028}$	<0
RP-GIN (Murphy et al., 2019)	$0.9092_{\pm.0033}$	<0	$0.1448_{\pm.0685}$	<0	$0.9235_{\pm.0051}$	<0	<0	<0
DIR (Wu et al., 2022)	$0.7707_{\pm.0609}$	<0	<0	<0	$0.8499_{\pm.0270}$	<0	<0	<0
Chemical language models								
polyBERT (Kuenneth & Ramprasad, 2023)	$0.9824_{\pm.0004}$	<0	$0.8626_{\pm.0065}$	<0	$0.9805_{\pm.0009}$	<0	$0.8852_{\pm.0014}$	<0
TransPolymer (Xu et al., 2023)	$0.9720_{\pm.0005}$	<0	$0.7910_{\pm.0126}$	<0	$0.9597_{\pm.0007}$	<0	$0.8332_{\pm.0043}$	<0
MoLFormer (Ross et al., 2022)	$0.9700_{\pm.0007}$	<0	$0.8759_{\pm.0028}$	<0	$0.9644_{\pm.0009}$	$0.2583_{\pm.0331}$	$0.8571_{\pm.0032}$	$0.2083_{\pm.0371}$
TrinityLLM (Liu et al., 2025)	$0.9709_{\pm.0006}$	<0	$0.8755_{\pm.0051}$	<0	$0.9672_{\pm.0008}$	$0.2006_{\pm.0474}$	$0.8641_{\pm.0033}$	$0.1766_{\pm.0534}$
Architecture-blind reference								
Morgan + GBM	0.9700	0.9700	0.9167	0.9167	0.9633	0.9633	0.8861	0.8861
Ours								
PolyBC-M	$\mathbf{0.9955}_{\pm.0001}$	$\mathbf{0.9941}_{\pm.0003}$	$0.9293_{\pm.0014}$	$0.9259_{\pm.0022}$	$\mathbf{0.9955}_{\pm.0001}$	$\mathbf{0.9943}_{\pm.0002}$	$0.9071_{\pm.0016}$	$0.9062_{\pm.0024}$
PolyBC-F	$0.9900_{\pm.0001}$	$0.9885_{\pm.0002}$	$\mathbf{0.9387}_{\pm.0015}$	$\mathbf{0.9389}_{\pm.0013}$	$0.9903_{\pm.0003}$	$0.9886_{\pm.0003}$	$\mathbf{0.9259}_{\pm.0012}$	$\mathbf{0.9235}_{\pm.0009}$

Baselines. *Atom-level graph models*: GRIN with $\{1, 3\}$-repeat augmentation (Zhu et al., 2025) and CIGA, DISGEN, GREA, SSR, GSAT, RP-GIN, and DIR (Chen et al., 2022; Huang et al., 2024; Liu et al., 2022; Buffelli et al., 2022; Miao et al., 2022; Murphy et al., 2019; Wu et al., 2022) on the atom graph of the R=1 realization. *Chemical LMs*: frozen polyBERT, TransPolymer, and MoLFormer encoders with an MLP head (Kuenneth & Ramprasad, 2023; Xu et al., 2023; Ross et al., 2022), and TrinityLLM, a MoLFormer head pre-trained on physics-simulated labels (Liu et al., 2025), on the realization's PSMILES. *Architecture-blind reference*: gradient-boosted trees (GBM) on the two monomers and f_A (Friedman, 2001), which measures what chemistry and composition alone explain. All baselines are validated on the training architecture and, except GRIN, trained on R=1 (Appendix C.1).

Implementation. Both variants use L=1, width 300, Δt=0.5, and 2.6 to 2.7M parameters, trained with Adam (learning rate 10^{-3}, batch 64) and early stopping on validation RMSE (squared error on Coley; absolute error with dropout and an ℓ_1 penalty on PoLyInfo; Appendix C).

4.2 Effectiveness and Architecture Transfer (Q1, Q2)

Obs. 1: PolyBC is the most accurate model in-domain on Coley. PolyBC-M attains R^2=0.9955 on EA and IP, and PolyBC-F 0.9900 and 0.9903 (Table 1). PolyBC-M reduces $1 - R^2$ by 74% (EA) and 87% (IP) relative to the strongest LM (polyBERT) and by 90% and 88% relative to the augmentation-trained GRIN; the other GNNs reach 0.77 to 0.95.

Obs. 2: Only PolyBC transfers better than an architecture-blind reference. A Morgan+GBM that never sees the sequence reaches 0.917 (EA) and 0.886 (IP) transfer, because architecture moves EA and IP far less than chemistry does; every baseline stays below it on average (GRIN 0.893 / 0.874; best LM 0.876 on EA and 0.885 on IP), whereas both of our variants exceed it, so reading the sequence does not by itself yield transfer. PolyBC-F transfers best on both properties (0.9387 on EA, 0.9259 on IP) and exceeds the reference in all six cells of each, whereas PolyBC-M (0.9293 / 0.9071) falls below it in one cell per property at Test1 (block→alternating on EA, alternating→block on IP). Consistent with Corollary 1, alternating→block and block→alternating are the two hardest transfers of our models, as for GRIN; for every other sequence-aware model the hardest also starts from alternating data.

Obs. 3: On experimental T_g, PolyBC-F is best in-domain and in transfer. PolyBC-F reaches an all-cell R^2 of 0.815 (Table 2), against 0.719 for the strongest GNN (GRIN, trained with augmentation) and 0.689 for the strongest LM (polyBERT); it leads in transfer by a wide margin (0.811 vs. 0.688 and 0.633) and has the highest in-domain mean (0.823 vs. 0.800 for polyBERT, a gap comparable to cross-process variation). It beats polyBERT in all nine cells, by less than 0.03 in four of them and most when trained on alternating data ($+0.345$ on alternating→block, $+0.316$ on

Table 2: **PoLyInfo experimental T_g, 3×3 results, Test1.** Full: n_{train}=3,690; Small: 12.5% per-architecture subsample (n_{train}=460), same validation and test rows. Mean test R^2 (5 seeds) over in-domain, transfer, and all cells. LM inputs follow each model's published context window, which truncates some long PoLyInfo strings at R=1 (Appendix C.1); our models encode each structural unit separately and are not bound by chain length.

	PoLyInfo-Full			PoLyInfo-Small		
Model	In-domain	Transfer	All	In-domain	Transfer	All
Atom-level graph models						
GRIN (Zhu et al., 2025)	$0.779_{\pm.023}$	$0.688_{\pm.038}$	$0.719_{\pm.033}$	$0.601_{\pm.052}$	$0.533_{\pm.061}$	$0.555_{\pm.058}$
CIGA (Chen et al., 2022)	$0.660_{\pm.079}$	$0.531_{\pm.067}$	$0.574_{\pm.068}$	$0.342_{\pm.110}$	$0.285_{\pm.072}$	$0.304_{\pm.076}$
DISGEN (Huang et al., 2024)	$0.596_{\pm.018}$	$0.462_{\pm.017}$	$0.506_{\pm.017}$	$0.318_{\pm.079}$	$0.225_{\pm.103}$	$0.256_{\pm.095}$
RP-GIN (Murphy et al., 2019)	$0.595_{\pm.036}$	$0.334_{\pm.049}$	$0.421_{\pm.044}$	$0.450_{\pm.042}$	$0.063_{\pm.312}$	$0.192_{\pm.217}$
GREA (Liu et al., 2022)	$0.711_{\pm.017}$	$0.273_{\pm.259}$	$0.419_{\pm.177}$	$0.180_{\pm.191}$	<0	<0
GSAT (Miao et al., 2022)	$0.150_{\pm.209}$	<0	$0.005_{\pm.263}$	$0.126_{\pm.240}$	<0	<0
SSR (Buffelli et al., 2022)	<0	<0	<0	0.167	<0	<0
DIR (Wu et al., 2022)	<0	<0	<0	<0	<0	<0
Chemical language models						
polyBERT (Kuenneth & Ramprasad, 2023)	$0.800_{\pm.012}$	$0.633_{\pm.016}$	$0.689_{\pm.012}$	$0.664_{\pm.011}$	$0.494_{\pm.029}$	$0.550_{\pm.022}$
MoLFormer (Ross et al., 2022)	$0.747_{\pm.012}$	$0.603_{\pm.013}$	$0.651_{\pm.008}$	$0.621_{\pm.014}$	$0.480_{\pm.019}$	$0.527_{\pm.016}$
TransPolymer (Xu et al., 2023)	$0.662_{\pm.018}$	$0.523_{\pm.015}$	$0.569_{\pm.015}$	$0.575_{\pm.016}$	$0.485_{\pm.024}$	$0.515_{\pm.020}$
TrinityLLM (Liu et al., 2025)	$0.663_{\pm.010}$	$0.496_{\pm.030}$	$0.551_{\pm.023}$	$0.555_{\pm.014}$	$0.402_{\pm.011}$	$0.453_{\pm.012}$
Ours						
PolyBC-M	$0.759_{\pm.025}$	$0.777_{\pm.013}$	$0.771_{\pm.013}$	$0.709_{\pm.017}$	$\mathbf{0.736}_{\pm.019}$	$\mathbf{0.727}_{\pm.016}$
PolyBC-F	$\mathbf{0.823}_{\pm.002}$	$\mathbf{0.811}_{\pm.012}$	$\mathbf{0.815}_{\pm.008}$	$\mathbf{0.757}_{\pm.019}$	$0.682_{\pm.022}$	$0.707_{\pm.017}$

alternating→random). It also beats GRIN in eight of nine cells (not random→alternating, 0.758 vs. 0.824). PolyBC-M is below polyBERT and GRIN in-domain (0.759 vs. 0.800 and 0.779) but transfers better (0.777 vs. 0.633 and 0.688). GREA, GSAT, SSR, DIR, and RP-GIN produce negative cells (DISGEN one, on PoLyInfo-Small), whereas CIGA does not. Every cell of ours, in both regimes, exceeds the weak blind reference of its test architecture (raw-latent features, diagonal only; none was run on our features).

4.3 Small Data (Q3)

Obs. 4: PolyBC stays the most accurate with $8\times$ fewer labels. With 460 PoLyInfo training records (230 copolymers, only 32 of them block), PolyBC-M (0.727) and PolyBC-F (0.707) remain far ahead of GRIN (0.555), polyBERT (0.550), and all other baselines (≤ 0.53). From Full to Small, PolyBC-M loses only 0.044 all-cell R^2 and PolyBC-F 0.108, against 0.124 to 0.164 for MoLFormer, polyBERT, and GRIN and 0.23 to 0.27 for CIGA, DISGEN, and RP-GIN; TransPolymer and TrinityLLM lose 0.054 and 0.098 from lower levels (0.569, 0.551). With few labels the monomer graph transfers better (0.736 vs. 0.682) while PolyBC-F stays better in-domain (0.757 vs. 0.709). On Coley (Table 10), with 300 or 1,000 labels per architecture both our models are more accurate than every baseline in-domain and in transfer, on EA and IP. The closest are the chemical LMs (EA, n_{train}=300: TrinityLLM 0.807 vs. 0.815 in-domain and 0.732 vs. 0.773 in transfer), which collapse at R=20; GRIN trails PolyBC-F by 0.07 to 0.13 in-domain, and CIGA fails with few labels. The fragment graph leads the monomer graph in-domain at both sizes (IP: by 0.028 and 0.010), whereas with all labels the monomer graph is more accurate (0.996 vs. 0.990). With few labels the fragment graph is thus more accurate in-domain on both datasets.

4.4 Repetition Invariance (Q4)

Obs. 5: Invariance is exact where the theory predicts it. On Coley, every cell tested on 1:1 alternating chains returns identical R^2, RMSE, and MAE to four decimals at $R \in \{1, 5, 10, 20\}$ for every seed, for both models on EA and IP (Theorem 1; Table 9). With the Neumann bulk closure, PolyBC-M is exactly invariant on every block test cell (block→block 0.9960 at all R; Proposition 1b), whereas PolyBC-F on block chains changes once from R=1 to R=5 (block→block 0.9904→0.9888 on EA, 0.9913→0.9882 on IP) and is constant afterwards (Proposition 1a). Random test cells move mostly between R=1 and R=5 (PolyBC-F EA random→random: 0.9923, 0.9897, 0.9894, 0.9893; Proposition 2); a few drift slowly afterwards (e.g. PolyBC-M EA block→random 0.9652→0.9593). Matrix averages change by at most 0.0016 (in-domain) and 0.0033 (transfer).

Obs. 6: Baselines collapse, and aggregate stability is not per-row invariance. Five of the seven invariance-oriented GNNs reach negative R^2 at R=20 and CIGA and DISGEN lose 0.07 to 0.28 in-

domain; chemical LMs collapse through truncation and the shift to 20-repeat strings (MoLFormer IP in-domain 0.964→0.258). GRIN, trained with augmentation, is stable but starts lower on Coley. On PoLyInfo no cell of ours changes by more than 0.0073 R^2 from Test1 to Test20, whereas polyBERT loses 0.263 (Full) and 0.157 (Small) all-cell (Table 12). We therefore claim exact invariance where it is proven and stable aggregate accuracy elsewhere; per-row RMS drift for PolyBC-F on PoLyInfo-Full is 0.29 to 3.96 °C, and the largest single-row changes (up to 80.7 °C) are reported in Table 13.

4.5 Ablations

Five ablations are summarized here and given in full, with their tables, in Appendix E. **Latent features.** The distilled map $\sigma(g(z))$ is what makes fragment nodes work (it adds 0.107 at n_{train}=300, where raw latents collapse to < 0); on Coley monomer nodes it is not needed, $\text{PCA}_{64}(z)$ being marginally better in transfer and marginally worse in-domain. **Open/cut bit.** Gains are obtained at larger data, $+0.0012$ on Coley to -0.004 at n_{train}=300. **Closure.** All closures are accurate (within 0.008 on Coley EA), and the choice matters for repetition behavior: the Neumann bulk drifts least and the sequence-blind canonical chain is least accurate for PolyBC-F. **Boundary condition.** Intervening on the ghost rule alone, with the chain, the weights and the readout held fixed, the architecture-matched rule is the most accurate. **Depth.** Test20 R^2 falls with depth while Test1 is flat, and max aggregation is less stable than the sum.

4.6 Latent-Space Inverse Design (Q5)

Obs. 7: on the corpus this task actually has, the generative baselines collapse. Latent-only node features make the property model differentiable in the monomer latent. Optimizing PolyBC-M itself from 28 training copolymers and decoding every iterate yields 736 designs spanning 163 repeat units absent from the 683 Coley monomers (Appendix G). Trained on those same 683 monomers with the official MOSES trainers (Polykovskiy et al., 2020), a VAE emits 3 distinct repeat units per 1,000 samples and an AAE exactly 1, and every molecule the VAE emits is already in its training set; G2S-VAE (Vogel & Weber, 2025), trained on the 42,966 Coley copolymers, is posterior-collapsed and returns 5 memorized strings. Our loop returns 31.2 ± 5.8 distinct units per 1,000 iterates at $65.1 \pm 6.7\%$ novelty (Table 19).

Obs. 8: An independent oracle confirms the two objectives reachable by substituent edits. GFN2-xTB (Bannwarth et al., 2019) agrees with the intended direction for 88% of EA-maximizing and 73% of IP-minimizing designs ($+0.49$ and -0.53 eV), through chemically sensible substituent swaps (Appendix G); for the other two objectives it agrees 45% and 64% of the time, with changes of at most 0.04 eV. The designed molecules are real and oracle-validated, and the analysis isolates the one component to improve: the gain predicted at the optimized latent ($+0.78$ eV) is partly retained once the decode is re-encoded ($+0.14$ eV overall, $+0.02$ eV on EA maximization; Table 20), so the encoder–decoder round trip, bounds latent-space design.

5 Conclusion

We introduced PolyBC, a graph-augmented chemical language model that separates the chemistry of copolymer structural units from their arrangement. The chemistry is supplied by label-free latent features from a frozen chemical LM. The arrangement is handled by a nonlocal message-passing operator whose boundary condition encodes the copolymer architecture. The construction comes with theoretical guarantees on repetition invariance and generalizability: predictions are exactly invariant to the number of repeat units for 1:1 alternating chains at any depth and for monomer-level block chains at the depth we use, become invariant beyond a finite repeat number for all block chains, and depend on the repeat number only through the two chain ends for random chains. One interior operator is shared by all architectures, which is what makes zero-shot transfer between them well defined. Trained on single-repeat chains only, one of the two PolyBC variants is the most accurate model in-domain and in transfer on both benchmarks, both variants remain the most accurate with $8\times$ fewer labels, and accuracy holds at 20 repeats, where most baselines collapse. Because the node features are differentiable in the LM latents, the same trained model drives latent-space inverse design: an independent xTB oracle confirms the direction of the designed changes for the two objectives reachable by substituent edits, and the decode–re-encode round trip, not the optimizer or the property model, is the remaining bottleneck. Extending design to the fragment variant and validating designs experimentally are the natural next steps.

AI USE STATEMENT

In this work, we used generative AI tools only for writing and polishing the paper, namely drafting and editing parts of the text to improve readability. We did not use generative AI tools for any of the other tasks listed in the ICLR 2027 AI Policy for Authors. We have reviewed all AI-assisted text, and we take responsibility for the final content of this work, including text and claims produced with the aid of generative AI.

ETHICS STATEMENT

This work concerns property prediction and design for polymer materials from public benchmark data (the copolymer set of Aldeghi & Coley, 2022, and PoLyInfo (Otsuka et al., 2011)). It involves no human subjects or personal data. Generated candidate structures are hypotheses that require synthesis, safety, and environmental assessment before any use.

REPRODUCIBILITY STATEMENT

Section 3 and Appendix C specify the model, features, and training protocol; Appendix A contains complete proofs; Appendix B documents the datasets, repeat-unit construction, fragmentation, and splits; Appendix D reports per-cell results with standard deviations at every evaluated repeat number; and Appendices E to G give the ablations, the enumerated T_g benchmark, and design details, including the exact optimizer, decoder, oracle and generative-baseline protocols. We will release code, splits, feature bundles, trained checkpoints, and per-row predictions.

REFERENCES

Matteo Aldeghi and Connor W. Coley. A graph representation of molecular ensembles for polymer property prediction. *Chemical Science*, 13(35):10486–10498, 2022. doi: 10.1039/D2SC02839E.

Evan R. Antoniuk, Peggy Li, Bhavya Kailkhura, and Anna M. Hiszpanski. Representing polymers as periodic graphs with learned descriptors for accurate polymer property predictions. *Journal of Chemical Information and Modeling*, 62(22):5435–5445, 2022. doi: 10.1021/acs.jcim.2c00875.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

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

Frank S. Bates and Glenn H. Fredrickson. Block copolymer thermodynamics: Theory and experiment. *Annual Review of Physical Chemistry*, 41(1):525–557, 1990. doi: 10.1146/annurev.pc.41.100190.002521.

Rohit Batra, Hanjun Dai, Tran Doan Huan, Lihua Chen, Chiho Kim, Will R. Gutekunst, Le Song, and Rampi Ramprasad. Polymers for extreme conditions designed using syntax-directed variational autoencoders. *Chemistry of Materials*, 32(24):10489–10500, 2020. doi: 10.1021/acs.chemmater.0c03332.

Beatrice Bevilacqua, Yangze Zhou, and Bruno Ribeiro. Size-invariant graph representations for graph classification extrapolations. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 837–851, 2021.

Manav Bhati, Mohammad Atif Faiz Afzal, Alex K. Chew, Andrea R. Browning, and Mathew D. Halls. Glass transition prediction of binary copolymers across large chemical spaces using machine learning and physics-based modeling. *Polymers*, 18(14):1727, 2026. doi: 10.3390/polym18141727.

Davide Buffelli, Pietro Liò, and Fabio Vandin. SizeShiftReg: A regularization method for improving size-generalization in graph neural networks. In *Advances in Neural Information Processing Systems*, volume 35, pp. 31871–31885, 2022.

Ben Chamberlain, James Rowbottom, Maria I. Gorinova, Michael Bronstein, Stefan Webb, and Emanuele Rossi. GRAND: Graph neural diffusion. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 1407–1418, 2021.

Lihua Chen, Ghanshyam Pilania, Rohit Batra, Tran Doan Huan, Chiho Kim, Christopher Kuenneth, and Rampi Ramprasad. Polymer informatics: Current status and critical next steps. *Materials Science and Engineering: R: Reports*, 144:100595, 2021. doi: 10.1016/j.mser.2020.100595.

Yongqiang Chen, Yonggang Zhang, Yatao Bian, Han Yang, Kaili Ma, Binghui Xie, Tongliang Liu, Bo Han, and James Cheng. Learning causally invariant representations for out-of-distribution generalization on graphs. In *Advances in Neural Information Processing Systems*, volume 35, pp. 22131–22148, 2022.

Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. ChemBERTa: Large-scale self-supervised pretraining for molecular property prediction. *arXiv preprint arXiv:2010.09885*, 2020.

Jörg Degen, Christof Wegscheid-Gerlach, Andrea Zaliani, and Matthias Rarey. On the art of compiling and using 'drug-like' chemical fragment spaces. *ChemMedChem*, 3(10):1503–1507, 2008. doi: 10.1002/cmdc.200800178.

Qiang Du, Max Gunzburger, R. B. Lehoucq, and Kun Zhou. Analysis and approximation of nonlocal diffusion problems with volume constraints. *SIAM Review*, 54(4):667–696, 2012. doi: 10.1137/110833294.

Qiang Du, Max Gunzburger, R. B. Lehoucq, and Kun Zhou. A nonlocal vector calculus, nonlocal volume-constrained problems, and nonlocal balance laws. *Mathematical Models and Methods in Applied Sciences*, 23(3):493–540, 2013. doi: 10.1142/S0218202512500546.

Moshe Eliasof, Eldad Haber, and Eran Treister. PDE-GCN: Novel architectures for graph neural networks motivated by partial differential equations. In *Advances in Neural Information Processing Systems*, volume 34, pp. 3836–3849, 2021.

Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001. doi: 10.1214/aos/1013203451.

Rafael Gómez-Bombarelli, Jennifer N. Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D. Hirzel, Ryan P. Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276, 2018. doi: 10.1021/acscentsci.7b00572.

Rishi Gurnani, Deepak Kamal, Huan Tran, Harikrishna Sahu, Kenny Scharm, Usman Ashraf, and Rampi Ramprasad. polyG2G: A novel machine learning algorithm applied to the generative design of polymer dielectrics. *Chemistry of Materials*, 33(17):7008–7016, 2021. doi: 10.1021/acs.chemmater.1c02061.

Rishi Gurnani, Christopher Kuenneth, Aubrey Toland, and Rampi Ramprasad. Polymer informatics at scale with multitask graph neural networks. *Chemistry of Materials*, 35(4):1560–1567, 2023. doi: 10.1021/acs.chemmater.2c02991.

Zheng Huang, Qihui Yang, Dawei Zhou, and Yujun Yan. Enhancing size generalization in graph neural networks through disentangled representation learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 20365–20381, 2024.

Ross Irwin, Spyridon Dimitriadis, Jiazhen He, and Esben Jannik Bjerrum. Chemformer: A pre-trained transformer for computational chemistry. *Machine Learning: Science and Technology*, 3 (1):015022, 2022. doi: 10.1088/2632-2153/ac3ffb.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2323–2332, 2018.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Hierarchical generation of molecular graphs using structural motifs. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 4839–4848, 2020.

Artur Kadurin, Alexander Aliper, Andrey Kazennov, Polina Mamoshina, Quentin Vanhaelen, Kuzma Khrabrov, and Alex Zhavoronkov. The cornucopia of meaningful leads: Applying deep adversarial autoencoders for new molecule development in oncology. *Oncotarget*, 8(7):10883–10890, 2017. doi: 10.18632/oncotarget.14073.

George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021. doi: 10.1038/s42254-021-00314-5.

Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to PDEs. *Journal of Machine Learning Research*, 24(89):1–97, 2023.

Christopher Kuenneth and Rampi Ramprasad. polyBERT: A chemical language model to enable fully machine-driven ultrafast polymer informatics. *Nature Communications*, 14(1):4099, 2023. doi: 10.1038/s41467-023-39868-6.

Christopher Kuenneth, William Schertzer, and Rampi Ramprasad. Copolymer informatics with multitask deep neural networks. *Macromolecules*, 54(13):5957–5961, 2021. doi: 10.1021/acs.macromol.1c00728.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Graph kernel network for partial differential equations. *arXiv preprint arXiv:2003.03485*, 2020.

Gang Liu, Tong Zhao, Jiaxin Xu, Tengfei Luo, and Meng Jiang. Graph rationalization with environment-based augmentations. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1069–1078. ACM, 2022. doi: 10.1145/3534678.3539347.

Ning Liu, Siavash Jafarzadeh, Brian Y. Lattimer, Shuna Ni, Jim Lua, and Yue Yu. Harnessing large language models for data-scarce learning of polymer properties. *Nature Computational Science*, 5(3):245–254, 2025. doi: 10.1038/s43588-025-00768-y.

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

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021. doi: 10.1038/s42256-021-00302-5.

Siqi Miao, Mia Liu, and Pan Li. Interpretable and generalizable graph learning via stochastic attention mechanism. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 15524–15543, 2022.

Ryan Murphy, Balasubramaniam Srinivasan, Vinayak Rao, and Bruno Ribeiro. Relational pooling for graph representations. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4663–4673, 2019.

Shingo Otsuka, Isao Kuwajima, Junko Hosoya, Yibin Xu, and Masayoshi Yamazaki. PoLyInfo: Polymer database for polymeric materials design. In *2011 International Conference on Emerging Intelligent Data and Web Technologies (EIDWT)*, pp. 22–29. IEEE, 2011. doi: 10.1109/EIDWT.2011.13.

Jaehong Park, Youngseon Shim, Franklin Lee, Aravind Rammohan, Sushmit Goyal, Munbo Shim, Changwook Jeong, and Dae Sin Kim. Prediction and interpretation of polymer properties using the graph convolutional network. *ACS Polymers Au*, 2(4):213–222, 2022. doi: 10.1021/acspolymersau.1c00050.

Roshan A. Patel, Carlos H. Borca, and Michael A. Webb. Featurization strategies for polymer sequence or composition design by machine learning. *Molecular Systems Design & Engineering*, 7(6):661–676, 2022. doi: 10.1039/D1ME00160D.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12(85): 2825–2830, 2011.

Daniil Polykovskiy, Alexander Zhebrak, Benjamin Sanchez-Lengeling, Sergey Golovanov, Oktai Tatanov, Stanislav Belyaev, Rauf Kurbanov, Aleksey Artamonov, Vladimir Aladinskiy, Mark Veselov, Artur Kadurin, Simon Johansson, Hongming Chen, Sergey Nikolenko, Alán Aspuru-Guzik, and Alex Zhavoronkov. Molecular sets (MOSES): A benchmarking platform for molecular generation models. *Frontiers in Pharmacology*, 11:565644, 2020. doi: 10.3389/fphar.2020.565644.

Rampi Ramprasad, Rohit Batra, Ghanshyam Pilania, Arun Mannodi-Kanakkithodi, and Chiho Kim. Machine learning in materials informatics: Recent applications and prospects. *npj Computational Materials*, 3(1):54, 2017. doi: 10.1038/s41524-017-0056-5.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, 2010. doi: 10.1021/ci100050t.

Jerret Ross, Brian Belgodere, Vijil Chenthamarakshan, Inkit Padhi, Youssef Mroueh, and Payel Das. Large-scale chemical language representations capture molecular structure and properties. *Nature Machine Intelligence*, 4(12):1256–1264, 2022. doi: 10.1038/s42256-022-00580-7.

S. A. Silling. Reformulation of elasticity theory for discontinuities and long-range forces. *Journal of the Mechanics and Physics of Solids*, 48(1):175–209, 2000. doi: 10.1016/S0022-5096(99)00029-0.

Eduardo Soares, Emilio Vital Brazil, Victor Shirasuna, Dmitry Zubarev, Renato Cerqueira, and Kristin Schmidt. An open-source family of large encoder-decoder foundation models for chemistry. *Communications Chemistry*, 8(1):193, 2025. doi: 10.1038/s42004-025-01585-0.

Peter C. St. John, Caleb Phillips, Travis W. Kemper, A. Nolan Wilson, Yanfei Guan, Michael F. Crowley, Mark R. Nimlos, and Ross E. Larsen. Message-passing neural networks for high-throughput polymer screening. *The Journal of Chemical Physics*, 150(23):234111, 2019. doi: 10.1063/1.5099132.

Petar Veličković and Charles Blundell. Neural algorithmic reasoning. *Patterns*, 2(7):100273, 2021. doi: 10.1016/j.patter.2021.100273.

Gabriel Vogel and Jana M. Weber. Inverse design of copolymers including stoichiometry and chain architecture. *Chemical Science*, 16(3):1161–1178, 2025. doi: 10.1039/D4SC05900J.

Yingxin Wu, Xiang Wang, An Zhang, Xiangnan He, and Tat-Seng Chua. Discovering invariant rationales for graph neural networks. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=hGXij5rfiHw.

Changwen Xu, Yuyang Wang, and Amir Barati Farimani. TransPolymer: A transformer-based language model for polymer property predictions. *npj Computational Materials*, 9(1):64, 2023. doi: 10.1038/s41524-023-01016-5.

Keyulu Xu, Jingling Li, Mozhi Zhang, Simon S. Du, Ken-ichi Kawarabayashi, and Stefanie Jegelka. What can neural networks reason about? In *International Conference on Learning Representations (ICLR)*, 2020. URL https://openreview.net/forum?id=rJxbJeHFPS.

Gilad Yehudai, Ethan Fetaya, Eli Meirom, Gal Chechik, and Haggai Maron. From local structures to size generalization in graph neural networks. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 11975–11986, 2021.

Naiyu Yin, Ning Liu, Jiuzhou Chen, Brian Y. Lattimer, Jim Lua, and Yue Yu. Fake it till you make it: Multi-physics synthesis breaks the data barrier in chemical language models. In *NeurIPS 2025 Workshop on Machine Learning and the Physical Sciences (ML4PS)*, 2025. Paper 260.

Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform badly for graph representation? In *Advances in Neural Information Processing Systems*, volume 34, pp. 28877–28888, 2021.

Huaiqian You, Yue Yu, Marta D'Elia, Tian Gao, and Stewart Silling. Nonlocal kernel network (NKN): A stable and resolution-independent deep neural network. *Journal of Computational Physics*, 469:111536, 2022. doi: 10.1016/j.jcp.2022.111536.

Zaixi Zhang, Qi Liu, Hao Wang, Chengqiang Lu, and Chee-Kong Lee. Motif-based graph self-supervised learning for molecular property prediction. In *Advances in Neural Information Processing Systems*, volume 34, pp. 15870–15882, 2021.

Haiteng Zhao, Shengchao Liu, Chang Ma, Hannan Xu, Jie Fu, Zhi-Hong Deng, Lingpeng Kong, and Qi Liu. GIMLET: A unified graph-text model for instruction-based molecule zero-shot learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 5850–5887, 2023.

Yihan Zhu, Gang Liu, Eric Inae, Tengfei Luo, and Meng Jiang. Learning repetition-invariant representations for polymer informatics. In *Advances in Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=uO0EHn4EmE.

A PROOFS

Throughout, the ghost-augmented neighbor maps are $\ell(i) = i - 1$, $r(i) = i + 1$ for interior nodes, and at the ends $\ell(1) = N$, $r(N) = 1$ (periodic) or $\ell(1) = 1$, $r(N) = N$ (Neumann). The ghost edge features are $e_1^- = [-1, 1]$ and $e_N^+ = [+1, 1]$ on fragment graphs (ghost bonds are open) and $[\mp 1]$ on monomer graphs. Under the mean-field boundary condition the ghost is $\bar{h}^{(k-1)}$ rather than a node state. All statements refer to evaluation mode (no dropout).

Proof of Lemma 1. By induction on k. For $k = 0$, $h_i^{(0)} = \mathrm{ELU}(\mathrm{LN}_0(W_{\mathrm{in}} x_i + b_{\mathrm{in}}))$ is a function of $x_i = \mathcal{E}_0(i)$. Assume $h_j^{(k-1)} = \Phi_{k-1}(\mathcal{E}_{k-1}(j))$ for all j. By Eq. (3) and the boundary condition, $h_i^{(k)}$ is a fixed function of $\big(h_i^{(k-1)}, o_i^-, o_i^+, h_{\ell(i)}^{(k-1)}, h_{r(i)}^{(k-1)}\big)$, because the edge features $e_i^{\pm}$ are determined by $o_i^{\pm}$. Substituting the induction hypothesis gives $h_i^{(k)} = \Phi_k(\mathcal{E}_k(i))$. The maps $W_{\mathrm{in}}, \phi_k, \psi_k$ are shared by all nodes, chains, and architectures, so Φ_k depends on θ only. Unfolding the recursion, $\mathcal{E}_L(i)$ is determined by the inputs of the nodes reachable in at most L steps of ℓ, r and by the bonds among them; reordering nodes outside this window leaves $h_i^{(L)}$ unchanged. For a node with $L < i \le N - L$ the recursion never evaluates $\ell(1)$ or $r(N)$, so the induction is independent of the ghost rule and holds under any boundary condition, including the mean-field one. □

Proof of Theorem 1. Let $m = |u|$ and $N = mR$. The node inputs of G_R are periodic, $x_i(G_R) = x_{\pi(i)}(G_1)$. The open bits are periodic as well, because occurrence boundaries repeat with period m and, under the periodic boundary condition, the bonds $N \to 1$ of G_R and $m \to 1$ of G_1 are both open bonds between consecutive occurrences. The periodic neighbor maps commute with π: $\pi(\ell_{G_R}(i)) = \ell_{G_1}(\pi(i))$ and $\pi(r_{G_R}(i)) = r_{G_1}(\pi(i))$ for all i, including $i \in \{1, N\}$. By induction on k, $\mathcal{E}_k(i; G_R) = \mathcal{E}_k(\pi(i); G_1)$, hence $h_i^{(L)}(G_R) = h_{\pi(i)}^{(L)}(G_1)$ by Lemma 1. The node states of G_R are therefore R copies of those of G_1, with types $\tau(i) = \tau(\pi(i))$. The element-wise maximum over a multiset equals the maximum over its support, so M_A, M_B coincide on G_R and G_1; the fraction of A nodes is invariant under R-fold duplication, and f_A is a property of P. Bulk phantoms depend only on the per-type mean input, the (open) ghost edge features, and, for the mean-field bulk, the composition weight, all of which are invariant under duplication. Hence all arguments of Eq. (8) coincide and $f_\theta(G_R) = f_\theta(G_1)$. The argument uses no property of the aggregation beyond its being a function of the two neighbor messages, so it holds for sum and max aggregation alike. □

Proof of Proposition 1. (a) Write $\lambda_A = n_A R$ and $\lambda_B = n_B R$ for the run lengths in monomer occurrences, and $k_A, k_B \ge 1$ for the number of nodes per occurrence (k_t=1 on monomer graphs); distances are counted in nodes. By Lemma 1, a node's state is determined by $\mathcal{E}_L(i)$, which is fixed by (i) the unit-internal position of i, (ii) its distance d_ℓ to the left chain end if $d_\ell < L$, and (iii) its distance d_j to the $A|B$ junction if $d_j < L$; if both distances are at least L, i is a bulk node whose environment is a window of the periodic A-run pattern. If $\lambda_A \ge 2L + 1$, the A-run has $k_A \lambda_A \ge 2L + 1$ nodes, so no window touches both the chain end and the junction, and its bulk region of $k_A \lambda_A - 2L \ge k_A$ nodes contains every unit-internal phase; every class is realized, and increasing R only adds copies of bulk nodes. The same holds for the B-run. Hence, for $R \ge R_0$, the set of distinct states of each type is independent of R, the maxima M_A, M_B are constant, the composition weight is R-invariant, and so is $f_\theta(G_R)$.

(b) For monomer nodes and $L = 1$, $\mathcal{E}_1(i)$ is determined by $(\tau(i); \tau(\ell(i)), \tau(r(i)))$, because all monomer edges carry $[\mp 1]$ and a monomer's input depends only on its type. Under the Neumann boundary condition the left end of the A-run has $\ell(1) = 1$ and hence the environment $(A; A, A)$ of a bulk A node whenever $\lambda_A \ge 2$; the last A node has $(A; A, B)$; symmetrically the B-run realizes $(B; A, B)$ and, if $\lambda_B \ge 2$, $(B; B, B)$. If $\lambda_A = 1$ the single A node has $(A; A, B)$ because its left neighbor is itself. Hence, for every $R \ge 1$, the set of A-states is $\{(A; A, B)\}$ or $\{(A; A, B), (A; A, A)\}$, and likewise for B. The Neumann bulk phantom b_A carries the mean input of type A, which equals x_A on a monomer graph, and has both neighbors equal to itself with edge features $[\mp 1]$; its layer-1 state is exactly that of environment $(A; A, A)$. After the closure M_A is the maximum over $\{(A; A, B), (A; A, A)\}$ for every $R \ge 1$, and likewise M_B. With the R-invariant composition weight, $f_\theta(G_R) = f_\theta(G_1)$. □

Proof of Proposition 2. At $L = 1$ the mean-field ghost is $\bar{h}^{(0)} = \rho\, \bar{h}_A^{(0)} + (1 - \rho)\, \bar{h}_B^{(0)}$ with node composition $\rho = n_A k_A / (n_A k_A + n_B k_B)$ and $\bar{h}_t^{(0)}$ the mean layer-0 state of the k_t nodes of one t

occurrence (k_t=1 on monomer graphs), because layer-0 states depend only on the node's own input; ρ and $\bar{h}_t^{(0)}$ do not depend on R. On monomer graphs, fix an interior environment $(t; a, b)$, i.e. the consecutive pattern $a\,t\,b$, let j be the number of A's in it, $T = \lfloor N/3 \rfloor$, and X the number of the T disjoint consecutive triples equal to the pattern. By exchangeability, $p_N := \Pr(\text{triple} = a\,t\,b) = (n_A R)_j (n_B R)_{3-j}/(N)_3 \to q^j(1-q)^{3-j} > 0$ with $q = n_A/(n_A{+}n_B) \in (0,1)$ and $(n)_j$ the falling factorial, and for two distinct triples $\Pr(\text{both}) = (n_A R)_{2j}(n_B R)_{6-2j}/(N)_6 = p_N^2 + O(1/N)$. Hence $\mathbb{E}[X] = Tp_N = \Theta(N)$ and $\mathrm{Var}(X) \le Tp_N + T^2\,O(1/N) = O(N)$ (the covariances need not be negative: for $n_A R$=2, $n_B R$=4, i.e. unit $(1,2)$ at R=2, and pattern BAB, $\Pr(\text{both}) = 1/15 > (1/5)^2$). Chebyshev's inequality gives $\Pr(X=0) \le \mathrm{Var}(X)/\mathbb{E}[X]^2 = O(1/N) \to 0$, and a union bound over the eight patterns shows that all interior environments occur with probability tending to one. On fragment graphs the within-occurrence environments are the same for every R, and the same argument applies to the finitely many occurrence-junction environments. The only remaining R-dependence of the type-wise maxima comes from the two end nodes, whose environments involve $\bar{h}$ and range over a finite, R-independent set. □

Proof of Proposition 3. The parameters $(W_{\text{in}}, \phi_k, \psi_k, \text{MLP})$ and the edge features contain no architecture-dependent component. Both nodes satisfy the position condition, so the ghost-free part of Lemma 1 (or, without a mean-field boundary condition, its first part) gives $h_i^{(L)} = \Phi_L(\mathcal{E}_L(i)) = \Phi_L(\mathcal{E}_L(j)) = h_j^{(L)}$. □

Proof of Corollary 1. In $(AB)^n$ with the periodic boundary condition every A node has two B neighbors and every B node two A neighbors, including the ends. The block case is part (b) of the proof of Proposition 1. For random chains the claim is Proposition 2; the examples follow by inspection ($BBAB$: third node $(A; B, B)$; $BAAA$: third node $(A; A, A)$). Each such chain contains environments of only one other architecture, and the R=1 chain BA at $f_A{=}\frac{1}{2}$ has no interior node. □

B Datasets and Preprocessing

Table 3: **Datasets.** Unseen test monomers never occur in training. [a]Copolymer records (1,451 random, 1,098 alternating, 427 block); training also contains 617 homopolymer records replicated under each architecture label (1,851 rows; 4,827 in total).

Dataset	Property	Copolymers	Monomers	Fragments	Train / Valid / Test rows	Unseen test mon.	Architectures
Coley (Aldeghi & Coley, 2022)	EA, IP (eV)	42,966	683	307	25,774 / 4,298 / 12,894	0 of 664	alt., block, random
PoLyInfo (Otsuka et al., 2011)	T_g (°C)	2,976[a]	956	739	3,690 / 284 / 853	210 of 333	alt., block, random

Repeat-unit construction. The unit (n_A, n_B) is the closest fraction to f_A with $n_A{+}n_B \le 20$ (on PoLyInfo the realized fraction differs from f_A by at most 0.02). Coley has $f_A \in \{0.25, 0.5, 0.75\}$, giving $(1,3), (1,1), (3,1)$. Block chains are a single macro-block $A^{n_A R}B^{n_B R}$, not a tiling of $A^{n_A}B^{n_B}$. Random chains are resampled independently at every R from the target composition, with one fixed sequence per copolymer so that all models see the same chains. At small R the random label is close to degenerate: every Coley R=1 random chain at f_A=0.5 is the two-node chain BA, identical to an alternating unit, so architecture-sensitive evaluation of random chains comes from the multi-repeat tests up to R=20, where our models remain stable. A unit with n_A=0 or n_B=0 gives a homopolymer chain. PoLyInfo alternating records with $0 < n_A \ne n_B$ (alternating-tendency condensation copolymers whose chain-level unit order is effectively random) use the soft-alternating sequence $(AB)^{mR}X^{|n_A - n_B|R}$ with $m = \min(n_A, n_B)$ and X the majority type, under the periodic boundary condition. Of the 330 alternating test rows, 254 are soft-alternating, 67 are strict 1:1, and 9 round to a homopolymer unit.

Fragmentation. Fused ring systems (smallest-set-of-smallest-rings rings merged while they share atoms) and connected acyclic components of non-dummy atoms each form one group; every polymer $*$ is folded into the group of its neighbor; all inter-group bonds are cut with dummy atoms, so each fragment is a PSMILES with at least one $*$. We prefer this rule to BRICS (Degen et al., 2008), which leaves 420 of the 683 Coley monomers unsplit. The ordering of the fragment list is canonical but not chemical: consecutive fragments are bonded in 49% (Coley) and 51% (PoLyInfo) of multi-fragment monomers, and the list follows the backbone in only 1.9% and 32.1%. The fragment view

is therefore close to an unordered description of the monomer, which bounds what it can resolve. On Coley, positional isomers can map to the same fragment list (104 monomers share 51 lists), an irreducible error floor for PolyBC-F on that benchmark. Table 4 summarizes the vocabularies.

Table 4: **Fragment vocabularies** (ring/linker fragmentation, counting each unique monomer once).

Dataset	Monomers	Fragments	Frag./monomer (max)	Unsplit monomers	Top-10 share
Coley	683	307	2.41 (10)	105	46.1%
PoLyInfo	956	739	4.90 (28)	248	55.9%
Enumerated T_g	315	255	3.32 (14)	130	54.8%

Coley splits. Pair-level 60/10/30 on the ordered pair $s_A \| s_B$ with seed 42: 25,774 / 4,298 / 12,894 rows, i.e. 11,046 / 1,842 / 5,526 for random and block and 3,682 / 614 / 1,842 for alternating. The test split contains 664 monomers, all seen in other pairings, so Coley numbers measure generalization to new monomer *pairs*; generalization to unseen monomers is measured on PoLyInfo (210 of 333 test monomers unseen).

PoLyInfo. Starting from 6,629 PoLyInfo T_g records of strictly binary copolymers with a measured architecture label, canonicalization and deduplication give 2,976 copolymer rows (random 1,451, alternating 1,098, block 427) over 798 unordered monomer pairs (803 ordered). We add 617 homopolymer records to the training split, each replicated under all three architecture labels, giving 4,827 rows over 956 monomers; per-architecture counts are in Table 3. A homopolymer record is admitted only when the same measurement does not also appear in validation or test: composition endpoints of a copolymer series pass a pair-level split unnoticed while carrying the test row's own measurement, and admitting them would leak the label. Homopolymer records are admitted only if their record identifier occurs in no validation or test row. Train/valid/test contain 3,690/284/853 rows (random 1,531/160/377, alternating 1,284/101/330, block 875/23/146); the 1,851 homopolymer rows all lie in training, leaving 914/667/258 copolymer training rows. No record or sample identifier is shared between train and test, and only 3 unordered monomer pairs are (7 test rows: 5 alternating, 2 block). PoLyInfo-Small keeps 191/160/109 training rows (115/83/32 copolymers). Table 5 reports graph sizes; one-node graphs are homopolymer units.

Table 5: **PoLyInfo graph sizes at** $R{=}1$ (nodes per graph and share of one-node graphs; F: fragment, M: monomer).

Arch.	Split	Graphs	Frag. nodes (mean / max)	Mon. nodes (mean / max)	1-node (F / M)
random	train	1,531	12.65 / 100	8.06 / 20	21.2% / 38.7%
random	test	377	19.16 / 99	11.94 / 20	2.7% / 3.7%
alternating	train	1,284	23.36 / 366	5.88 / 20	24.2% / 44.9%
alternating	test	330	38.48 / 281	8.83 / 20	0.6% / 2.7%
block	train	875	9.16 / 100	4.78 / 20	35.7% / 65.4%
block	test	146	22.39 / 94	13.12 / 20	0.0% / 0.7%

C Implementation Details

Poly4mer. The encoder is the frozen SMI-TED-Light backbone (12 layers, 12 heads, width 768, rotary linear attention, 202-token context, pretrained on PubChem) (Soares et al., 2025). Its token embeddings, with a fixed (untrained) embedding for the polymer token $*$, are compressed by an MLP autoencoder bottleneck ($202{\times}768 \to 3072 \to 1536 \to 768$) into $z \in \mathbb{R}^{768}$; unlike the sum-pooled latent of the published model (Yin et al., 2025), this bottleneck latent is decoded directly by a mirrored decoder and a token classifier over the 2,393-token vocabulary, trained with token cross-entropy plus a reconstruction loss on token embeddings. For Coley and PoLyInfo we fine-tune Poly4mer label-free for reconstruction on PSMILES, which needs no labels and reconstructs 92.8% of the 683 Coley and 92.4% of 1,433 PoLyInfo monomers exactly.

Distillation head and PCA. g is trained with AdamW (learning rate 10^{-3}, weight decay 10^{-4}, cosine schedule, 12,000 epochs) on the unlabeled units of each corpus: the 885 unique Coley strings

(683 monomers and 307 fragments; in-sample Tanimoto 0.914 after thresholding at 0.5) and the 739 PoLyInfo fragments (0.973), so no label enters the node features. The node input is the continuous $\sigma(g(z))$, centered and projected on the leading p principal components (not whitened; 98.0% of variance for p=512 on Coley, 95.2% for p=256 on PoLyInfo). Every node feature is therefore a deterministic function of z alone: no feature column is binary, and no global or edge input carries fingerprint information. No feature column is binary, and no global or edge input carries fingerprint information.

Table 6: **Hyperparameters of the main models.**

	Coley EA / IP	PoLyInfo T_g
Node feature (PolyBC-F)	$\mathrm{PCA}_{512}(\sigma(g(z_{\text{frag}})))$	$\mathrm{PCA}_{256}(\sigma(g(z_{\text{frag}})))$
Node feature (PolyBC-M)	$\mathrm{PCA}_{512}(\sigma(g(z_{\text{mono}})))$	mean of fragment features
Edge feature	$[\pm 1]$ (+ open/cut bit for PolyBC-F)	$[\pm 1]$ (+ open/cut bit for PolyBC-F)
Layers L / width / Δt	1 / 300 / 0.5	1 / 300 / 0.5
Aggregation / readout	sum / type-wise max, Eq. (8)	sum / type-wise max, Eq. (8)
Composition weight α	fraction of A nodes	molar fraction f_A
Closure	Neumann bulk	mean-field bulk / none / Neumann bulk (random / alt. / block training)
Loss	mean squared error	absolute error (°C) $+ \lambda(e)\|\theta\|_1$ (schedule[a])
Dropout / weight decay	0 / 0	0.2 / 10^{-5}
Optimizer / batch	Adam, 10^{-3} / 64	Adam, 10^{-3} / 64
Epochs / patience	3,000 / 300	400 / 100
Output scaling	none	$\mu_y + \sigma_y \hat{y}$ (training labels)
Parameters	2.69M (F) / 2.69M (M)	2.61M (F) / 2.61M (M)
Seeds / train R / test R	5 / 1 / 1, 5, 10, 20	5 / 1 / 1, 20

[a] $\lambda = 0$ for epochs $e < 50$, rising linearly to 10^{-3} at $e = 100$.

Training. Table 6 lists all hyperparameters. Batches containing a single graph or node are skipped. Metrics are computed with scikit-learn (Pedregosa et al., 2011) on the original scale; standard deviations are population standard deviations over the 5 seeds of a single process.

C.1 Baselines

All baselines use the same splits, 5 seeds (GRIN: 3), R=1 training (GRIN: $\{1, 3\}$), and validation on the training architecture.

GRIN: the authors' configuration with a GCN encoder, max aggregation and an ℓ_1 weight penalty (the MST-aligned setting of Zhu et al., 2025), 5 layers, width 300, and a mean-pooled readout concatenated with a whole-chain Morgan fingerprint.

CIGA, DISGEN, GREA, SSR, GSAT, RP-GIN, DIR: official implementations adapted to regression (squared-error loss; CIGA with a 10-bin contrastive target), with hyperparameters selected on validation of the training architecture where the original method prescribes a search.

Chemical LMs: frozen encoders with an MLP head; for MoLFormer the learning rate and head shape are selected on validation of the training architecture, whereas polyBERT and TransPolymer use their published head and learning rate (10^{-3}) with the epoch selected on validation. Inputs are the PSMILES of the realization, built with an order-preserving stitcher (126 Coley rows whose monomers carry no attachment point and 3 PoLyInfo alternating training rows cannot be stitched and are dropped). Each LM uses its published context window. On Coley, 20-repeat strings are encoded with a 4,096-token window for MoLFormer and truncated at 512 tokens for polyBERT and TransPolymer. On PoLyInfo, long repeat units exceed the default windows (202 tokens for MoLFormer, 512 for polyBERT and TransPolymer) already at R=1 for 1,313, 331, and 297 of the 2,907 unique strings; our models, which encode each structural unit separately, are not bound by chain length.

TrinityLLM (Liu et al., 2025): frozen MoLFormer-XL with a two-stage head that is first trained on physics-simulated labels and then fine-tuned on the benchmark; the authors' fire-property corpus

has no counterpart here, so stage 1 fits the xTB ionization potential or electron affinity of each Coley monomer (681 of 683; the two monomers without attachment points are skipped) and, on PoLyInfo, the Van Krevelen group-contribution T_g of each monomer in the training rows. Each monomer is presented as its homopolymer dimer chain and encoded like the copolymer chains, so no benchmark label enters stage 1. Stage 2 uses the MoLFormer inputs, with learning rate $\{3, 10, 30\} \times 10^{-5}$ and head shape selected on validation on Coley and TrinityLLM's stage-2 learning rate 10^{-4} on PoLyInfo.

Architecture-blind GBM: on Coley, Morgan fingerprints (radius 2, 2048 bits) of both monomers reduced by SVD to 64 dimensions, their composition-weighted mix and f_A, and histogram gradient boosting (deterministic).

D Full Results

Tables 7 to 13 report every cell of the 3×3 matrices with standard deviations, our models at every repeat number, the Coley data-efficiency runs, and the PoLyInfo per-row drift.

Table 7: **Coley EA, all** 3×3 **cells** (train→test; r/a/b = random/alternating/block). Test R^2, mean$_{\pm sd}$ over seeds. "<0": negative mean R^2 (values range from -0.006 to below -10^6). ‡GRIN runs of Appendix C.1 (3 seeds, $\{1,3\}$ augmentation).

Model	r→r	r→a	r→b	a→r	a→a	a→b	b→r	b→a	b→b
Test1									
GRIN‡ ({1,3} repeat-unit aug.) (Zhu et al., 2025)	$0.9358_{\pm.0002}$	$0.9453_{\pm.0007}$	$0.9355_{\pm.0032}$	$0.8752_{\pm.0020}$	$0.9594_{\pm.0001}$	$0.8097_{\pm.0040}$	$0.9175_{\pm.0009}$	$0.8746_{\pm.0055}$	$0.9673_{\pm.0019}$
CIGA (Chen et al., 2022)	$0.9050_{\pm.0275}$	$0.8981_{\pm.0376}$	$0.9055_{\pm.0229}$	$0.8125_{\pm.0213}$	$0.9006_{\pm.0085}$	$0.7924_{\pm.0313}$	$0.8832_{\pm.0236}$	$0.8429_{\pm.0259}$	$0.9138_{\pm.0250}$
DISGEN (Huang et al., 2024)	$0.8735_{\pm.0066}$	$0.8813_{\pm.0094}$	$0.8646_{\pm.0093}$	$0.7436_{\pm.0230}$	$0.8446_{\pm.0171}$	$0.7221_{\pm.0254}$	$0.8668_{\pm.0045}$	$0.8463_{\pm.0086}$	$0.8854_{\pm.0060}$
GREA (Liu et al., 2022)	$0.9383_{\pm.0026}$	$0.9403_{\pm.0086}$	$0.9369_{\pm.0050}$	$0.4029_{\pm.3012}$	$0.9360_{\pm.0106}$	$0.3570_{\pm.3266}$	$0.9115_{\pm.0082}$	$0.8627_{\pm.0135}$	$0.9643_{\pm.0060}$
SSR (Buffelli et al., 2022)	$0.8962_{\pm.0057}$	$0.8975_{\pm.0051}$	$0.8872_{\pm.0110}$	<0	$0.8974_{\pm.0179}$	<0	$0.8365_{\pm.0323}$	$0.7524_{\pm.0557}$	$0.8920_{\pm.0448}$
GSAT (Miao et al., 2022)	$0.8999_{\pm.0037}$	$0.8956_{\pm.0086}$	$0.8996_{\pm.0039}$	<0	$0.8623_{\pm.0033}$	<0	$0.8434_{\pm.0140}$	$0.8355_{\pm.0213}$	$0.9137_{\pm.0027}$
RP-GIN (Murphy et al., 2019)	$0.8855_{\pm.0099}$	$0.8983_{\pm.0177}$	$0.8824_{\pm.0111}$	<0	$0.9079_{\pm.0080}$	<0	$0.8322_{\pm.0209}$	$0.8428_{\pm.0183}$	$0.9342_{\pm.0106}$
DIR (Wu et al., 2022)	$0.7025_{\pm.2159}$	$0.7140_{\pm.1695}$	$0.7215_{\pm.2038}$	<0	$0.7741_{\pm.0648}$	<0	$0.8160_{\pm.0234}$	$0.7843_{\pm.0475}$	$0.8355_{\pm.0178}$
polyBERT (Kuenneth & Ramprasad, 2023)	$0.9813_{\pm.0005}$	$0.9360_{\pm.0057}$	$0.9397_{\pm.0024}$	$0.7274_{\pm.0220}$	$0.9836_{\pm.0010}$	$0.7647_{\pm.0166}$	$0.9305_{\pm.0012}$	$0.8770_{\pm.0042}$	$0.9824_{\pm.0006}$
TransPolymer (Xu et al., 2023)	$0.9697_{\pm.0017}$	$0.9282_{\pm.0016}$	$0.9256_{\pm.0028}$	$0.5344_{\pm.0439}$	$0.9753_{\pm.0006}$	$0.5646_{\pm.0391}$	$0.9174_{\pm.0043}$	$0.8755_{\pm.0067}$	$0.9709_{\pm.0006}$
MoLFormer (Ross et al., 2022)	$0.9732_{\pm.0015}$	$0.9357_{\pm.0030}$	$0.9408_{\pm.0013}$	$0.7899_{\pm.0062}$	$0.9649_{\pm.0004}$	$0.7759_{\pm.0070}$	$0.9364_{\pm.0019}$	$0.8766_{\pm.0048}$	$0.9718_{\pm.0004}$
TrinityLLM (Liu et al., 2025)	$0.9726_{\pm.0008}$	$0.9364_{\pm.0019}$	$0.9403_{\pm.0007}$	$0.7903_{\pm.0136}$	$0.9658_{\pm.0014}$	$0.7712_{\pm.0127}$	$0.9384_{\pm.0004}$	$0.8766_{\pm.0038}$	$0.9742_{\pm.0008}$
Morgan + GBM	0.9736	0.9489	0.9504	0.9081	0.9642	0.8495	0.9611	0.8824	0.9722
PolyBC-M	$0.9972_{\pm.0001}$	$0.9705_{\pm.0011}$	$0.9753_{\pm.0006}$	$0.9241_{\pm.0027}$	$0.9932_{\pm.0001}$	$0.8543_{\pm.0032}$	$0.9754_{\pm.0013}$	$0.8761_{\pm.0093}$	$0.9960_{\pm.0002}$
PolyBC-F	$0.9923_{\pm.0002}$	$0.9672_{\pm.0018}$	$0.9690_{\pm.0014}$	$0.9336_{\pm.0026}$	$0.9874_{\pm.0001}$	$0.8799_{\pm.0046}$	$0.9744_{\pm.0013}$	$0.9082_{\pm.0022}$	$0.9904_{\pm.0002}$
Test20									
GRIN‡ ({1,3} repeat-unit aug.) (Zhu et al., 2025)	$0.9133_{\pm.0003}$	$0.9246_{\pm.0032}$	$0.9079_{\pm.0029}$	$0.8722_{\pm.0020}$	$0.9562_{\pm.0000}$	$0.8035_{\pm.0045}$	$0.8990_{\pm.0008}$	$0.8546_{\pm.0074}$	$0.9406_{\pm.0010}$
CIGA (Chen et al., 2022)	$0.9003_{\pm.0212}$	$0.9041_{\pm.0243}$	$0.8939_{\pm.0191}$	$0.7041_{\pm.0990}$	$0.7180_{\pm.1185}$	$0.7228_{\pm.0878}$	$0.8771_{\pm.0240}$	$0.8513_{\pm.0416}$	$0.8965_{\pm.0166}$
DISGEN (Huang et al., 2024)	$0.8550_{\pm.0080}$	$0.8631_{\pm.0116}$	$0.8469_{\pm.0155}$	$0.6103_{\pm.0856}$	$0.6748_{\pm.0940}$	$0.6161_{\pm.0607}$	$0.8547_{\pm.0078}$	$0.8385_{\pm.0116}$	$0.8727_{\pm.0084}$
GREA (Liu et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0
SSR (Buffelli et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0
GSAT (Miao et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0
RP-GIN (Murphy et al., 2019)	<0	<0	<0	<0	<0	<0	<0	<0	<0
DIR (Wu et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0
polyBERT (Kuenneth & Ramprasad, 2023)	<0	<0	<0	<0	<0	<0	<0	<0	<0
TransPolymer (Xu et al., 2023)	<0	<0	<0	<0	<0	<0	<0	<0	<0
MoLFormer (Ross et al., 2022)	<0	<0	<0	<0	<0	<0	<0	$0.0290_{\pm.1574}$	<0
TrinityLLM (Liu et al., 2025)	<0	$0.1116_{\pm.0415}$	<0	<0	<0	<0	<0	$0.2397_{\pm.1120}$	<0
Morgan + GBM	0.9736	0.9489	0.9504	0.9081	0.9642	0.8495	0.9611	0.8824	0.9722
PolyBC-M	$0.9932_{\pm.0006}$	$0.9705_{\pm.0011}$	$0.9753_{\pm.0006}$	$0.9202_{\pm.0024}$	$0.9932_{\pm.0001}$	$0.8543_{\pm.0032}$	$0.9593_{\pm.0065}$	$0.8761_{\pm.0093}$	$0.9960_{\pm.0002}$
PolyBC-F	$0.9893_{\pm.0005}$	$0.9672_{\pm.0018}$	$0.9696_{\pm.0017}$	$0.9328_{\pm.0019}$	$0.9874_{\pm.0001}$	$0.8795_{\pm.0037}$	$0.9760_{\pm.0014}$	$0.9082_{\pm.0022}$	$0.9888_{\pm.0004}$

Table 8: **Coley IP, all 3×3 cells.** Same format as Table 7.

Model	r→r	r→a	r→b	a→r	a→a	a→b	b→r	b→a	b→b
Test1									
GRIN‡ ({1,3} repeat-unit aug.) (Zhu et al., 2025)	$0.9270_{\pm.0018}$	$0.9388_{\pm.0044}$	$0.9391_{\pm.0040}$	$0.8468_{\pm.0036}$	$0.9753_{\pm.0014}$	$0.7991_{\pm.0063}$	$0.9008_{\pm.0031}$	$0.8182_{\pm.0166}$	$0.9810_{\pm.0010}$
CIGA (Chen et al., 2022)	$0.8870_{\pm.0221}$	$0.8900_{\pm.0132}$	$0.8939_{\pm.0287}$	$0.7933_{\pm.0451}$	$0.8956_{\pm.0311}$	$0.7753_{\pm.0499}$	$0.8924_{\pm.0144}$	$0.8046_{\pm.0131}$	$0.9641_{\pm.0218}$
DISGEN (Huang et al., 2024)	$0.8310_{\pm.0215}$	$0.8360_{\pm.0262}$	$0.8284_{\pm.0264}$	$0.5851_{\pm.0635}$	$0.7342_{\pm.0379}$	$0.5997_{\pm.0655}$	$0.8078_{\pm.0240}$	$0.7455_{\pm.0279}$	$0.8617_{\pm.0273}$
GREA (Liu et al., 2022)	$0.9303_{\pm.0026}$	$0.9302_{\pm.0144}$	$0.9426_{\pm.0119}$	<0	$0.9410_{\pm.0188}$	<0	$0.9039_{\pm.0045}$	$0.8171_{\pm.0088}$	$0.9769_{\pm.0024}$
SSR (Buffelli et al., 2022)	$0.9283_{\pm.0043}$	$0.9309_{\pm.0084}$	$0.9428_{\pm.0028}$	$0.3171_{\pm.4619}$	$0.9479_{\pm.0163}$	$0.3524_{\pm.4042}$	$0.8807_{\pm.0114}$	$0.7848_{\pm.0272}$	$0.9719_{\pm.0104}$
GSAT (Miao et al., 2022)	$0.9132_{\pm.0027}$	$0.8977_{\pm.0081}$	$0.9297_{\pm.0052}$	<0	$0.9060_{\pm.0098}$	<0	$0.8623_{\pm.0109}$	$0.7790_{\pm.0240}$	$0.9529_{\pm.0028}$
RP-GIN (Murphy et al., 2019)	$0.8968_{\pm.0086}$	$0.9135_{\pm.0059}$	$0.8921_{\pm.0135}$	<0	$0.9155_{\pm.0069}$	<0	$0.8484_{\pm.0244}$	$0.8071_{\pm.0504}$	$0.9582_{\pm.0046}$
DIR (Wu et al., 2022)	$0.8815_{\pm.0349}$	$0.8878_{\pm.0386}$	$0.8907_{\pm.0432}$	<0	$0.7586_{\pm.1185}$	<0	$0.8613_{\pm.0287}$	$0.8029_{\pm.0339}$	$0.9096_{\pm.0346}$
polyBERT (Kuenneth & Ramprasad, 2023)	$0.9798_{\pm.0011}$	$0.9123_{\pm.0031}$	$0.9450_{\pm.0014}$	$0.8656_{\pm.0047}$	$0.9783_{\pm.0013}$	$0.8349_{\pm.0052}$	$0.9414_{\pm.0017}$	$0.8121_{\pm.0043}$	$0.9833_{\pm.0007}$
TransPolymer (Xu et al., 2023)	$0.9616_{\pm.0009}$	$0.8884_{\pm.0025}$	$0.9233_{\pm.0015}$	$0.7496_{\pm.0115}$	$0.9530_{\pm.0016}$	$0.7391_{\pm.0110}$	$0.9074_{\pm.0032}$	$0.7911_{\pm.0074}$	$0.9644_{\pm.0011}$
MoLFormer (Ross et al., 2022)	$0.9691_{\pm.0024}$	$0.9071_{\pm.0028}$	$0.9296_{\pm.0046}$	$0.8037_{\pm.0088}$	$0.9527_{\pm.0022}$	$0.7619_{\pm.0104}$	$0.9332_{\pm.0026}$	$0.8072_{\pm.0052}$	$0.9713_{\pm.0014}$
TrinityLLM (Liu et al., 2025)	$0.9719_{\pm.0007}$	$0.9083_{\pm.0020}$	$0.9342_{\pm.0024}$	$0.8199_{\pm.0118}$	$0.9578_{\pm.0015}$	$0.7785_{\pm.0090}$	$0.9364_{\pm.0007}$	$0.8073_{\pm.0033}$	$0.9718_{\pm.0013}$
Morgan + GBM	0.9691	0.9200	0.9452	0.8862	0.9552	0.8295	0.9378	0.7981	0.9657
PolyBC-M	$0.9971_{\pm.0001}$	$0.9563_{\pm.0011}$	$0.9702_{\pm.0005}$	$0.9043_{\pm.0052}$	$0.9932_{\pm.0003}$	$0.8102_{\pm.0096}$	$0.9657_{\pm.0005}$	$0.8357_{\pm.0075}$	$0.9962_{\pm.0002}$
PolyBC-F	$0.9923_{\pm.0002}$	$0.9542_{\pm.0028}$	$0.9674_{\pm.0012}$	$0.9324_{\pm.0006}$	$0.9872_{\pm.0008}$	$0.8757_{\pm.0026}$	$0.9714_{\pm.0008}$	$0.8546_{\pm.0030}$	$0.9913_{\pm.0003}$
Test20									
GRIN‡ ({1,3} repeat-unit aug.) (Zhu et al., 2025)	$0.9219_{\pm.0021}$	$0.9131_{\pm.0168}$	$0.9394_{\pm.0049}$	$0.8430_{\pm.0034}$	$0.9734_{\pm.0011}$	$0.7979_{\pm.0044}$	$0.8989_{\pm.0035}$	$0.8132_{\pm.0129}$	$0.9789_{\pm.0016}$
CIGA (Chen et al., 2022)	$0.8833_{\pm.0218}$	$0.8604_{\pm.0292}$	$0.9014_{\pm.0207}$	$0.5734_{\pm.1856}$	$0.4845_{\pm.2307}$	$0.6562_{\pm.1689}$	$0.8934_{\pm.0139}$	$0.8163_{\pm.0194}$	$0.9559_{\pm.0164}$
DISGEN (Huang et al., 2024)	$0.8274_{\pm.0220}$	$0.8334_{\pm.0300}$	$0.8295_{\pm.0281}$	$0.0451_{\pm.2112}$	<0	$0.1939_{\pm.1859}$	$0.8082_{\pm.0298}$	$0.7646_{\pm.0490}$	$0.8498_{\pm.0335}$
GREA (Liu et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0
SSR (Buffelli et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0
GSAT (Miao et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0
RP-GIN (Murphy et al., 2019)	<0	<0	<0	<0	<0	<0	<0	<0	<0
DIR (Wu et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0
polyBERT (Kuenneth & Ramprasad, 2023)	$0.1173_{\pm.0829}$	$0.0572_{\pm.1221}$	<0	<0	<0	<0	<0	<0	<0
TransPolymer (Xu et al., 2023)	<0	<0	<0	<0	<0	<0	<0	<0	<0
MoLFormer (Ross et al., 2022)	$0.3775_{\pm.0265}$	$0.3939_{\pm.0276}$	$0.2504_{\pm.0407}$	$0.0554_{\pm.1052}$	$0.1366_{\pm.1113}$	<0	$0.3506_{\pm.0426}$	$0.3798_{\pm.0322}$	$0.2608_{\pm.0665}$
TrinityLLM (Liu et al., 2025)	$0.2292_{\pm.0764}$	$0.2629_{\pm.0624}$	$0.0479_{\pm.1006}$	$0.1599_{\pm.1141}$	$0.2352_{\pm.1104}$	<0	$0.2783_{\pm.0794}$	$0.3544_{\pm.0566}$	$0.1373_{\pm.1067}$
Morgan + GBM	0.9691	0.9200	0.9452	0.8862	0.9552	0.8295	0.9378	0.7981	0.9657
PolyBC-M	$0.9936_{\pm.0003}$	$0.9563_{\pm.0011}$	$0.9702_{\pm.0005}$	$0.8967_{\pm.0038}$	$0.9932_{\pm.0003}$	$0.8102_{\pm.0096}$	$0.9683_{\pm.0019}$	$0.8357_{\pm.0075}$	$0.9962_{\pm.0002}$
PolyBC-F	$0.9905_{\pm.0005}$	$0.9542_{\pm.0028}$	$0.9654_{\pm.0010}$	$0.9254_{\pm.0019}$	$0.9872_{\pm.0008}$	$0.8696_{\pm.0020}$	$0.9720_{\pm.0006}$	$0.8546_{\pm.0030}$	$0.9882_{\pm.0005}$

Table 9: **PolyBC on Coley at every repeat number** (Neumann bulk closure). Test R^2 at $R \in \{1, 5, 10, 20\}$ and RMSE (eV) at $R \in \{1, 20\}$, mean$_{\pm \text{sd}}$ over 5 seeds.

Train→test	R^2 Test1	Test5	Test10	Test20	RMSE Test1	RMSE Test20
EA, PolyBC-F						
random→random	$0.9923_{\pm.0002}$	$0.9897_{\pm.0004}$	$0.9894_{\pm.0005}$	$0.9893_{\pm.0005}$	$0.0574_{\pm.0008}$	$0.0677_{\pm.0014}$
random→alt.	$0.9672_{\pm.0018}$	$0.9672_{\pm.0018}$	$0.9672_{\pm.0018}$	$0.9672_{\pm.0018}$	$0.1176_{\pm.0032}$	$0.1176_{\pm.0032}$
random→block	$0.9690_{\pm.0014}$	$0.9696_{\pm.0017}$	$0.9696_{\pm.0017}$	$0.9696_{\pm.0017}$	$0.1162_{\pm.0027}$	$0.1150_{\pm.0032}$
alt.→random	$0.9336_{\pm.0026}$	$0.9334_{\pm.0018}$	$0.9329_{\pm.0018}$	$0.9328_{\pm.0019}$	$0.1685_{\pm.0034}$	$0.1694_{\pm.0024}$
alt.→alt.	$0.9874_{\pm.0001}$	$0.9874_{\pm.0001}$	$0.9874_{\pm.0001}$	$0.9874_{\pm.0001}$	$0.0729_{\pm.0004}$	$0.0729_{\pm.0004}$
alt.→block	$0.8799_{\pm.0046}$	$0.8795_{\pm.0037}$	$0.8795_{\pm.0037}$	$0.8795_{\pm.0037}$	$0.2286_{\pm.0044}$	$0.2290_{\pm.0036}$
block→random	$0.9744_{\pm.0013}$	$0.9761_{\pm.0014}$	$0.9760_{\pm.0014}$	$0.9760_{\pm.0014}$	$0.1045_{\pm.0027}$	$0.1012_{\pm.0029}$
block→alt.	$0.9082_{\pm.0022}$	$0.9082_{\pm.0022}$	$0.9082_{\pm.0022}$	$0.9082_{\pm.0022}$	$0.1967_{\pm.0024}$	$0.1967_{\pm.0024}$
block→block	$0.9904_{\pm.0002}$	$0.9888_{\pm.0004}$	$0.9888_{\pm.0004}$	$0.9888_{\pm.0004}$	$0.0645_{\pm.0005}$	$0.0698_{\pm.0013}$
EA, PolyBC-M						
random→random	$0.9972_{\pm.0001}$	$0.9943_{\pm.0004}$	$0.9937_{\pm.0005}$	$0.9932_{\pm.0006}$	$0.0344_{\pm.0004}$	$0.0537_{\pm.0024}$
random→alt.	$0.9705_{\pm.0011}$	$0.9705_{\pm.0011}$	$0.9705_{\pm.0011}$	$0.9705_{\pm.0011}$	$0.1116_{\pm.0021}$	$0.1116_{\pm.0021}$
random→block	$0.9753_{\pm.0006}$	$0.9753_{\pm.0006}$	$0.9753_{\pm.0006}$	$0.9753_{\pm.0006}$	$0.1038_{\pm.0012}$	$0.1038_{\pm.0012}$
alt.→random	$0.9241_{\pm.0027}$	$0.9206_{\pm.0023}$	$0.9206_{\pm.0024}$	$0.9202_{\pm.0024}$	$0.1800_{\pm.0033}$	$0.1846_{\pm.0028}$
alt.→alt.	$0.9932_{\pm.0001}$	$0.9932_{\pm.0001}$	$0.9932_{\pm.0001}$	$0.9932_{\pm.0001}$	$0.0535_{\pm.0003}$	$0.0535_{\pm.0003}$
alt.→block	$0.8543_{\pm.0032}$	$0.8543_{\pm.0032}$	$0.8543_{\pm.0032}$	$0.8543_{\pm.0032}$	$0.2518_{\pm.0028}$	$0.2518_{\pm.0028}$
block→random	$0.9754_{\pm.0013}$	$0.9652_{\pm.0050}$	$0.9618_{\pm.0058}$	$0.9593_{\pm.0065}$	$0.1026_{\pm.0027}$	$0.1315_{\pm.0102}$
block→alt.	$0.8761_{\pm.0093}$	$0.8761_{\pm.0093}$	$0.8761_{\pm.0093}$	$0.8761_{\pm.0093}$	$0.2285_{\pm.0086}$	$0.2285_{\pm.0086}$
block→block	$0.9960_{\pm.0002}$	$0.9960_{\pm.0002}$	$0.9960_{\pm.0002}$	$0.9960_{\pm.0002}$	$0.0418_{\pm.0013}$	$0.0418_{\pm.0013}$
IP, PolyBC-F						
random→random	$0.9923_{\pm.0002}$	$0.9908_{\pm.0004}$	$0.9906_{\pm.0005}$	$0.9905_{\pm.0005}$	$0.0462_{\pm.0007}$	$0.0512_{\pm.0013}$
random→alt.	$0.9542_{\pm.0028}$	$0.9542_{\pm.0028}$	$0.9542_{\pm.0028}$	$0.9542_{\pm.0028}$	$0.1067_{\pm.0033}$	$0.1067_{\pm.0033}$
random→block	$0.9674_{\pm.0012}$	$0.9654_{\pm.0010}$	$0.9654_{\pm.0010}$	$0.9654_{\pm.0010}$	$0.0959_{\pm.0017}$	$0.0988_{\pm.0014}$
alt.→random	$0.9324_{\pm.0006}$	$0.9268_{\pm.0015}$	$0.9258_{\pm.0018}$	$0.9254_{\pm.0019}$	$0.1369_{\pm.0006}$	$0.1437_{\pm.0019}$
alt.→alt.	$0.9872_{\pm.0008}$	$0.9872_{\pm.0008}$	$0.9872_{\pm.0008}$	$0.9872_{\pm.0008}$	$0.0565_{\pm.0017}$	$0.0565_{\pm.0017}$
alt.→block	$0.8757_{\pm.0026}$	$0.8696_{\pm.0020}$	$0.8696_{\pm.0020}$	$0.8696_{\pm.0020}$	$0.1873_{\pm.0020}$	$0.1918_{\pm.0015}$
block→random	$0.9714_{\pm.0008}$	$0.9719_{\pm.0007}$	$0.9721_{\pm.0006}$	$0.9720_{\pm.0006}$	$0.0890_{\pm.0012}$	$0.0880_{\pm.0009}$
block→alt.	$0.8546_{\pm.0030}$	$0.8546_{\pm.0030}$	$0.8546_{\pm.0030}$	$0.8546_{\pm.0030}$	$0.1904_{\pm.0019}$	$0.1904_{\pm.0019}$
block→block	$0.9913_{\pm.0003}$	$0.9882_{\pm.0005}$	$0.9882_{\pm.0005}$	$0.9882_{\pm.0005}$	$0.0494_{\pm.0009}$	$0.0576_{\pm.0013}$
IP, PolyBC-M						
random→random	$0.9971_{\pm.0001}$	$0.9944_{\pm.0002}$	$0.9939_{\pm.0003}$	$0.9936_{\pm.0003}$	$0.0284_{\pm.0004}$	$0.0422_{\pm.0009}$
random→alt.	$0.9563_{\pm.0011}$	$0.9563_{\pm.0011}$	$0.9563_{\pm.0011}$	$0.9563_{\pm.0011}$	$0.1043_{\pm.0014}$	$0.1043_{\pm.0014}$
random→block	$0.9702_{\pm.0005}$	$0.9702_{\pm.0005}$	$0.9702_{\pm.0005}$	$0.9702_{\pm.0005}$	$0.0916_{\pm.0008}$	$0.0916_{\pm.0008}$
alt.→random	$0.9043_{\pm.0052}$	$0.8984_{\pm.0041}$	$0.8978_{\pm.0039}$	$0.8967_{\pm.0038}$	$0.1628_{\pm.0044}$	$0.1692_{\pm.0031}$
alt.→alt.	$0.9932_{\pm.0003}$	$0.9932_{\pm.0003}$	$0.9932_{\pm.0003}$	$0.9932_{\pm.0003}$	$0.0413_{\pm.0009}$	$0.0413_{\pm.0009}$
alt.→block	$0.8102_{\pm.0096}$	$0.8102_{\pm.0096}$	$0.8102_{\pm.0096}$	$0.8102_{\pm.0096}$	$0.2314_{\pm.0059}$	$0.2314_{\pm.0059}$
block→random	$0.9657_{\pm.0005}$	$0.9674_{\pm.0015}$	$0.9681_{\pm.0018}$	$0.9683_{\pm.0019}$	$0.0975_{\pm.0007}$	$0.0937_{\pm.0028}$
block→alt.	$0.8357_{\pm.0075}$	$0.8357_{\pm.0075}$	$0.8357_{\pm.0075}$	$0.8357_{\pm.0075}$	$0.2023_{\pm.0046}$	$0.2023_{\pm.0046}$
block→block	$0.9962_{\pm.0002}$	$0.9962_{\pm.0002}$	$0.9962_{\pm.0002}$	$0.9962_{\pm.0002}$	$0.0329_{\pm.0007}$	$0.0329_{\pm.0007}$

Repetition drift per row. On PoLyInfo, soft-alternating rows are not tilings of their R=1 realization and random rows are redrawn at every R, so Theorem 1 does not apply to them; this explains nonzero drift on alternating test cells. Drift concentrates in random test cells. Only the block→block cell of PolyBC-M is exactly invariant (0.00 °C in both regimes), because on PoLyInfo only the block-trained model uses the Neumann bulk closure that Proposition 1b requires. For Coley we report aggregate rather than per-row evidence: Table 9 gives the 5-seed mean and standard deviation of R^2 at every R and of RMSE at $R \in \{1, 20\}$, and per-seed values on alternating test cells agree to four decimals at all R.

Table 10: **Data efficiency on Coley EA and IP.** In-domain / transfer test R^2 at Test1 (means over the three diagonal and six off-diagonal cells) when each architecture is trained on n_{train} labeled copolymers ("full": 11,046 random/block and 3,682 alternating training rows). Every model uses the same training and validation rows ($\max(n_{\text{train}}/6, 50)$ validation rows of the training architecture) and the complete test sets; 5 seeds (GRIN: 3). ‡Trained on $R \in \{1, 3\}$; all others on R=1. Bold: best per column and metric. Chemical LMs lose most of their accuracy at Test20 (Section 4.4).

	EA			IP		
Model	n_{train}=300	n_{train}=1,000	full	n_{train}=300	n_{train}=1,000	full
Atom-level graph models						
GRIN‡ (Zhu et al., 2025)	$0.747_{\pm.006}$ / $0.714_{\pm.005}$	$0.853_{\pm.004}$ / $0.808_{\pm.004}$	0.954 / 0.893	$0.718_{\pm.004}$ / $0.681_{\pm.004}$	$0.836_{\pm.003}$ / $0.783_{\pm.007}$	0.961 / 0.874
CIGA (Chen et al., 2022)	$0.216_{\pm.323}$ / $0.183_{\pm.297}$	$0.739_{\pm.048}$ / $0.684_{\pm.051}$	$0.906_{\pm.011}$ / $0.856_{\pm.016}$	<0 / <0	$0.578_{\pm.100}$ / $0.157_{\pm.349}$	$0.916_{\pm.013}$ / $0.842_{\pm.021}$
Chemical language models						
polyBERT (Kuenneth & Ramprasad, 2023)	$0.765_{\pm.019}$ / $0.725_{\pm.014}$	$0.894_{\pm.002}$ / $0.797_{\pm.009}$	$0.982_{\pm<.001}$ / $0.863_{\pm.007}$	$0.796_{\pm.007}$ / $0.728_{\pm.013}$	$0.888_{\pm.002}$ / $0.797_{\pm.005}$	$0.980_{\pm.001}$ / $0.885_{\pm.001}$
MoLFormer (Ross et al., 2022)	$0.785_{\pm.005}$ / $0.692_{\pm.012}$	$0.885_{\pm.003}$ / $0.738_{\pm.021}$	$0.970_{\pm.001}$ / $0.876_{\pm.003}$	$0.803_{\pm.007}$ / $0.721_{\pm.004}$	$0.885_{\pm.003}$ / $0.794_{\pm.006}$	$0.964_{\pm.001}$ / $0.857_{\pm.003}$
TrinityLLM (Liu et al., 2025)	$0.807_{\pm.003}$ / $0.732_{\pm.009}$	$0.879_{\pm.003}$ / $0.783_{\pm.005}$	$0.971_{\pm.001}$ / $0.876_{\pm.005}$	$0.788_{\pm.008}$ / $0.697_{\pm.013}$	$0.891_{\pm.003}$ / $0.799_{\pm.005}$	$0.967_{\pm.001}$ / $0.864_{\pm.003}$
Ours						
PolyBC-M	$0.813_{\pm.005}$ / $0.763_{\pm.006}$	$0.925_{\pm.003}$ / $0.877_{\pm.003}$	$\mathbf{0.995}_{\pm<.001}$ / $0.929_{\pm.001}$	$0.822_{\pm.005}$ / $0.773_{\pm.007}$	$0.929_{\pm.003}$ / $0.857_{\pm.008}$	$\mathbf{0.995}_{\pm<.001}$ / $0.907_{\pm.002}$
PolyBC-F	$\mathbf{0.815}_{\pm.004}$ / $\mathbf{0.773}_{\pm.005}$	$\mathbf{0.929}_{\pm.003}$ / $\mathbf{0.887}_{\pm.004}$	$0.990_{\pm<.001}$ / $\mathbf{0.939}_{\pm.001}$	$\mathbf{0.850}_{\pm.005}$ / $\mathbf{0.813}_{\pm.005}$	$\mathbf{0.938}_{\pm.002}$ / $\mathbf{0.884}_{\pm.004}$	$0.990_{\pm<.001}$ / $\mathbf{0.926}_{\pm.001}$

Table 11: **PoLyInfo T_g, all 3×3 cells, Test1.** Test R^2, mean$_{\pm sd}$ over 5 seeds. The architecture-blind reference (diagonal only) scores 0.480 / 0.253 / 0.451 on random / alternating / block. "<0": negative mean R^2; "n/a": no alternating-trained SSR run on PoLyInfo-Small.

Model	r→r	r→a	r→b	a→r	a→a	a→b	b→r	b→a	b→b
PoLyInfo-Full (n_{train}=3,690)									
GRIN‡ (Zhu et al., 2025)	$0.835_{\pm.007}$	$0.824_{\pm.018}$	$0.735_{\pm.015}$	$0.674_{\pm.052}$	$0.872_{\pm.001}$	$0.569_{\pm.038}$	$0.683_{\pm.031}$	$0.646_{\pm.136}$	$0.630_{\pm.076}$
CIGA (Chen et al., 2022)	$0.796_{\pm.068}$	$0.770_{\pm.052}$	$0.484_{\pm.069}$	$0.616_{\pm.167}$	$0.766_{\pm.053}$	$0.390_{\pm.153}$	$0.446_{\pm.269}$	$0.481_{\pm.273}$	$0.419_{\pm.219}$
DISGEN (Huang et al., 2024)	$0.629_{\pm.054}$	$0.500_{\pm.042}$	$0.527_{\pm.026}$	$0.519_{\pm.027}$	$0.739_{\pm.016}$	$0.502_{\pm.016}$	$0.339_{\pm.028}$	$0.382_{\pm.033}$	$0.418_{\pm.014}$
RP-GIN (Murphy et al., 2019)	$0.748_{\pm.017}$	$0.530_{\pm.045}$	$0.629_{\pm.038}$	$0.618_{\pm.044}$	$0.827_{\pm.020}$	$0.573_{\pm.023}$	<0	<0	$0.209_{\pm.107}$
GREA (Liu et al., 2022)	$0.770_{\pm.016}$	<0	$0.086_{\pm.671}$	$0.451_{\pm.163}$	$0.784_{\pm.018}$	$0.616_{\pm.098}$	$0.359_{\pm.135}$	$0.138_{\pm.528}$	$0.579_{\pm.052}$
GSAT (Miao et al., 2022)	$0.480_{\pm.260}$	$0.049_{\pm.455}$	$0.244_{\pm.415}$	<0	$0.352_{\pm.259}$	$0.393_{\pm.345}$	<0	<0	<0
SSR (Buffelli et al., 2022)	<0	<0	<0	$0.391_{\pm.154}$	$0.635_{\pm.085}$	$0.574_{\pm.060}$	$0.270_{\pm.275}$	$0.050_{\pm.273}$	$0.402_{\pm.168}$
DIR (Wu et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0
polyBERT (Kuenneth & Ramprasad, 2023)	$0.824_{\pm.013}$	$0.736_{\pm.027}$	$0.628_{\pm.021}$	$0.507_{\pm.036}$	$0.882_{\pm.004}$	$0.526_{\pm.045}$	$0.645_{\pm.015}$	$0.759_{\pm.022}$	$0.694_{\pm.047}$
MoLFormer (Ross et al., 2022)	$0.761_{\pm.015}$	$0.629_{\pm.027}$	$0.548_{\pm.013}$	$0.517_{\pm.039}$	$0.812_{\pm.006}$	$0.648_{\pm.023}$	$0.587_{\pm.018}$	$0.690_{\pm.029}$	$0.667_{\pm.030}$
TransPolymer (Xu et al., 2023)	$0.697_{\pm.014}$	$0.647_{\pm.015}$	$0.557_{\pm.012}$	$0.561_{\pm.051}$	$0.859_{\pm.011}$	$0.518_{\pm.041}$	$0.509_{\pm.031}$	$0.344_{\pm.102}$	$0.430_{\pm.056}$
TrinityLLM (Liu et al., 2025)	$0.717_{\pm.017}$	$0.576_{\pm.021}$	$0.494_{\pm.033}$	$0.502_{\pm.034}$	$0.796_{\pm.004}$	$0.684_{\pm.038}$	$0.406_{\pm.028}$	$0.312_{\pm.102}$	$0.476_{\pm.024}$
PolyBC-M	$0.815_{\pm.020}$	$0.717_{\pm.044}$	$0.776_{\pm.021}$	$0.800_{\pm.020}$	$0.851_{\pm.023}$	$0.807_{\pm.030}$	$0.795_{\pm.016}$	$0.767_{\pm.023}$	$0.612_{\pm.057}$
PolyBC-F	$0.877_{\pm.018}$	$0.758_{\pm.021}$	$0.831_{\pm.028}$	$0.823_{\pm.020}$	$0.884_{\pm.008}$	$0.870_{\pm.021}$	$0.797_{\pm.030}$	$0.784_{\pm.043}$	$0.709_{\pm.023}$
PoLyInfo-Small (n_{train}=460)									
GRIN‡ (Zhu et al., 2025)	$0.640_{\pm.041}$	$0.714_{\pm.006}$	$0.616_{\pm.005}$	$0.612_{\pm.030}$	$0.695_{\pm.002}$	$0.572_{\pm.036}$	$0.418_{\pm.200}$	$0.265_{\pm.316}$	$0.466_{\pm.161}$
CIGA (Chen et al., 2022)	$0.375_{\pm.231}$	$0.572_{\pm.109}$	$0.306_{\pm.062}$	$0.422_{\pm.104}$	$0.483_{\pm.093}$	$0.348_{\pm.185}$	$0.063_{\pm.149}$	$0.000_{\pm.429}$	$0.167_{\pm.139}$
DISGEN (Huang et al., 2024)	$0.391_{\pm.027}$	$0.320_{\pm.014}$	$0.305_{\pm.021}$	$0.316_{\pm.038}$	$0.427_{\pm.042}$	$0.401_{\pm.042}$	$0.123_{\pm.227}$	<0	$0.137_{\pm.191}$
RP-GIN (Murphy et al., 2019)	$0.489_{\pm.157}$	<0	$0.273_{\pm.423}$	$0.486_{\pm.112}$	$0.756_{\pm.025}$	$0.567_{\pm.044}$	<0	<0	$0.105_{\pm.137}$
GREA (Liu et al., 2022)	<0	<0	<0	$0.347_{\pm.120}$	$0.552_{\pm.104}$	$0.420_{\pm.170}$	<0	<0	$0.140_{\pm.226}$
GSAT (Miao et al., 2022)	$0.395_{\pm.080}$	<0	<0	<0	<0	$0.260_{\pm.188}$	$0.037_{\pm.076}$	<0	$0.256_{\pm.072}$
SSR (Buffelli et al., 2022)	$0.098_{\pm.543}$	<0	<0	<0	$0.410_{\pm.210}$	$0.037_{\pm.596}$	<0	<0	$0.236_{\pm.136}$
DIR (Wu et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0
polyBERT (Kuenneth & Ramprasad, 2023)	$0.633_{\pm.019}$	$0.550_{\pm.068}$	$0.439_{\pm.031}$	$0.300_{\pm.049}$	$0.804_{\pm.007}$	$0.444_{\pm.014}$	$0.555_{\pm.039}$	$0.674_{\pm.072}$	$0.554_{\pm.046}$
MoLFormer (Ross et al., 2022)	$0.534_{\pm.027}$	$0.377_{\pm.069}$	$0.359_{\pm.035}$	$0.512_{\pm.005}$	$0.716_{\pm.004}$	$0.639_{\pm.019}$	$0.442_{\pm.030}$	$0.549_{\pm.070}$	$0.614_{\pm.033}$
TransPolymer (Xu et al., 2023)	$0.581_{\pm.013}$	$0.595_{\pm.034}$	$0.506_{\pm.020}$	$0.480_{\pm.022}$	$0.690_{\pm.028}$	$0.562_{\pm.020}$	$0.358_{\pm.033}$	$0.407_{\pm.106}$	$0.454_{\pm.039}$
TrinityLLM (Liu et al., 2025)	$0.552_{\pm.022}$	$0.428_{\pm.019}$	$0.335_{\pm.016}$	$0.461_{\pm.041}$	$0.655_{\pm.016}$	$0.641_{\pm.007}$	$0.293_{\pm.021}$	$0.252_{\pm.020}$	$0.457_{\pm.012}$
PolyBC-M	$0.668_{\pm.044}$	$0.766_{\pm.047}$	$0.780_{\pm.060}$	$0.688_{\pm.036}$	$0.829_{\pm.027}$	$0.749_{\pm.034}$	$0.660_{\pm.020}$	$0.776_{\pm.024}$	$0.632_{\pm.034}$
PolyBC-F	$0.728_{\pm.036}$	$0.653_{\pm.037}$	$0.658_{\pm.037}$	$0.717_{\pm.023}$	$0.838_{\pm.010}$	$0.779_{\pm.019}$	$0.557_{\pm.146}$	$0.725_{\pm.019}$	$0.704_{\pm.054}$

Table 12: **PoLyInfo T_g at Test20** (trained on R=1; ‡on $R \in \{1,3\}$). Test R^2, mean$_{\pm sd}$ over 5 seeds (GRIN: 3), and all-cell mean (Test1 value in parentheses).

Model	r→r	r→a	r→b	a→r	a→a	a→b	b→r	b→a	b→b	All
PoLyInfo-Full										
GRIN‡ (Zhu et al., 2025)	$0.838_{\pm.008}$	$0.826_{\pm.016}$	$0.734_{\pm.015}$	$0.667_{\pm.055}$	$0.872_{\pm.001}$	$0.568_{\pm.040}$	$0.679_{\pm.030}$	$0.646_{\pm.138}$	$0.630_{\pm.075}$	0.718 (0.719)
CIGA (Chen et al., 2022)	$0.839_{\pm.037}$	$0.779_{\pm.102}$	$0.245_{\pm.190}$	$0.695_{\pm.077}$	$0.847_{\pm.023}$	$0.282_{\pm.151}$	$0.541_{\pm.084}$	$0.538_{\pm.122}$	$0.378_{\pm.069}$	0.572 (0.582)
DISGEN (Huang et al., 2024)	$0.538_{\pm.055}$	$0.468_{\pm.069}$	$0.405_{\pm.054}$	$0.484_{\pm.036}$	$0.741_{\pm.013}$	$0.328_{\pm.022}$	$0.368_{\pm.066}$	$0.419_{\pm.082}$	$0.347_{\pm.047}$	0.455 (0.464)
RP-GIN (Murphy et al., 2019)	<0	<0	<0	<0	<0	<0	<0	<0	<0	<0 (0.383)
GREA (Liu et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0	<0 (0.234)
GSAT (Miao et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0	<0 (<0)
SSR (Buffelli et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0	<0 (0.224)
DIR (Wu et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0	<0 (<0)
polyBERT (Kuenneth & Ramprasad, 2023)	$0.444_{\pm.073}$	$0.531_{\pm.063}$	$0.190_{\pm.113}$	$0.040_{\pm.230}$	$0.796_{\pm.022}$	$0.409_{\pm.154}$	$0.445_{\pm.042}$	$0.686_{\pm.024}$	$0.290_{\pm.067}$	0.426 (0.689)
MoLFormer (Ross et al., 2022)	$0.278_{\pm.069}$	$0.409_{\pm.041}$	$0.187_{\pm.091}$	<0	$0.714_{\pm.007}$	<0	$0.187_{\pm.112}$	$0.673_{\pm.030}$	$0.427_{\pm.075}$	<0 (0.651)
TransPolymer (Xu et al., 2023)	$0.234_{\pm.096}$	$0.530_{\pm.026}$	$0.242_{\pm.096}$	$0.079_{\pm.154}$	$0.728_{\pm.050}$	$0.429_{\pm.072}$	$0.212_{\pm.093}$	$0.135_{\pm.134}$	$0.280_{\pm.066}$	0.319 (0.569)
TrinityLLM (Liu et al., 2025)	$0.361_{\pm.078}$	$0.301_{\pm.035}$	$0.171_{\pm.123}$	<0	$0.719_{\pm.025}$	<0	$0.192_{\pm.030}$	$0.376_{\pm.071}$	$0.504_{\pm.029}$	<0 (0.551)
PolyBC-M	$0.812_{\pm.024}$	$0.717_{\pm.044}$	$0.776_{\pm.022}$	$0.799_{\pm.019}$	$0.852_{\pm.023}$	$0.808_{\pm.029}$	$0.794_{\pm.015}$	$0.767_{\pm.024}$	$0.612_{\pm.057}$	0.771 (0.771)
PolyBC-F	$0.874_{\pm.018}$	$0.758_{\pm.021}$	$0.831_{\pm.028}$	$0.821_{\pm.022}$	$0.884_{\pm.008}$	$0.870_{\pm.022}$	$0.797_{\pm.031}$	$0.784_{\pm.043}$	$0.709_{\pm.023}$	0.814 (0.815)
PoLyInfo-Small										
GRIN‡ (Zhu et al., 2025)	$0.611_{\pm.047}$	$0.718_{\pm.008}$	$0.613_{\pm.004}$	$0.608_{\pm.030}$	$0.698_{\pm.003}$	$0.560_{\pm.038}$	$0.420_{\pm.194}$	$0.270_{\pm.310}$	$0.467_{\pm.158}$	0.552 (0.555)
CIGA (Chen et al., 2022)	$0.480_{\pm.100}$	$0.472_{\pm.148}$	$0.195_{\pm.237}$	$0.299_{\pm.224}$	$0.509_{\pm.129}$	<0	<0	$0.127_{\pm.134}$	$0.213_{\pm.122}$	0.188 (0.225)
DISGEN (Huang et al., 2024)	$0.390_{\pm.026}$	$0.357_{\pm.017}$	$0.168_{\pm.025}$	$0.375_{\pm.035}$	$0.426_{\pm.022}$	$0.322_{\pm.026}$	$0.185_{\pm.193}$	$0.082_{\pm.314}$	$0.060_{\pm.180}$	0.263 (0.263)
RP-GIN (Murphy et al., 2019)	<0	<0	<0	<0	<0	<0	<0	<0	<0	<0 (0.184)
GREA (Liu et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0	<0 (<0)
GSAT (Miao et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0	<0 (<0)
SSR (Buffelli et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0	<0 (<0)
DIR (Wu et al., 2022)	<0	<0	<0	<0	<0	<0	<0	<0	<0	<0 (<0)
polyBERT (Kuenneth & Ramprasad, 2023)	$0.390_{\pm.022}$	$0.211_{\pm.075}$	$0.337_{\pm.032}$	$0.300_{\pm.032}$	$0.603_{\pm.022}$	$0.514_{\pm.048}$	$0.400_{\pm.055}$	$0.601_{\pm.055}$	$0.178_{\pm.204}$	0.393 (0.550)
MoLFormer (Ross et al., 2022)	$0.087_{\pm.049}$	$0.046_{\pm.110}$	$0.217_{\pm.067}$	<0	$0.650_{\pm.019}$	$0.007_{\pm.113}$	<0	$0.492_{\pm.089}$	<0	<0 (0.527)
TransPolymer (Xu et al., 2023)	$0.339_{\pm.050}$	$0.373_{\pm.071}$	$0.371_{\pm.032}$	$0.454_{\pm.017}$	$0.519_{\pm.047}$	$0.437_{\pm.061}$	$0.299_{\pm.043}$	$0.387_{\pm.135}$	$0.373_{\pm.043}$	0.395 (0.515)
TrinityLLM (Liu et al., 2025)	<0	$0.083_{\pm.026}$	$0.123_{\pm.090}$	$0.253_{\pm.095}$	$0.651_{\pm.018}$	$0.535_{\pm.042}$	$0.252_{\pm.029}$	$0.327_{\pm.025}$	$0.573_{\pm.019}$	0.306 (0.453)
PolyBC-M	$0.664_{\pm.041}$	$0.772_{\pm.047}$	$0.782_{\pm.059}$	$0.681_{\pm.041}$	$0.830_{\pm.025}$	$0.752_{\pm.031}$	$0.662_{\pm.021}$	$0.775_{\pm.025}$	$0.632_{\pm.034}$	0.728 (0.727)
PolyBC-F	$0.721_{\pm.036}$	$0.653_{\pm.037}$	$0.658_{\pm.038}$	$0.713_{\pm.025}$	$0.839_{\pm.010}$	$0.780_{\pm.019}$	$0.563_{\pm.148}$	$0.726_{\pm.019}$	$0.705_{\pm.054}$	0.706 (0.707)

Table 13: **PoLyInfo per-row repetition drift.** For each cell, the root-mean-square change of a test row's prediction between R=1 and R=20 (°C, averaged over 5 seeds); the last column is the largest single-row change in the matrix. R^2 changes of at most 0.0073 hide these movements.

Model	Train \ Test	random	alternating	block	worst row
PolyBC-F, Full	random	3.96	0.45	1.10	80.7
	alternating	3.30	1.35	1.22	
	block	1.41	0.52	0.29	
PolyBC-M, Full	random	4.69	1.71	2.25	36.1
	alternating	2.03	2.94	0.95	
	block	1.40	0.72	0.00	
PolyBC-F, Small	random	4.13	0.74	1.14	41.1
	alternating	3.86	1.04	1.82	
	block	2.90	1.85	1.18	
PolyBC-M, Small	random	8.13	4.28	4.04	125.3
	alternating	4.35	2.84	1.45	
	block	2.33	2.11	0.00	

E Additional Ablations

Latent features (Table 14, Appendix E). Raw Poly4mer latents collapse under repetition (fragment R^2 from 0.427 to < 0 at n_{train}=300); a fixed PCA removes the collapse on fragment nodes, and the distilled map adds 0.107 there. On monomer nodes of Coley the distilled map does not help: PCA(z) is better on block→block at n_{train}=300 (with a weak monomer head, Tanimoto 0.51), and at full data PCA$_{64}$(z) nodes are marginally less accurate in-domain (0.9948 / 0.9940 vs. 0.9955 on EA / IP) but transfer better (0.935 / 0.921 vs. 0.929 / 0.907). On PoLyInfo raw-latent PCA fails (0.307 vs. 0.636), and concatenating raw z onto $\sigma(g(z))$ hurts (0.707→0.467), because z carries 87% of the feature variance but little T_g signal.

Open/cut bit. Gains are small and inconsistent: $+0.0012$ on Coley EA and IP (7 of 9 cells each), $+0.013$ on PoLyInfo-Small (within cross-process variation), -0.004 on Coley IP at n_{train}=300.

Closure. All closures are accurate (within 0.008 on Coley EA, 0.020 on IP), and the choice matters for repetition behavior: the Neumann bulk drifts less than no closure, and the sequence-blind canonical chain is less accurate for PolyBC-F (all-cell 0.9484 vs. 0.9558).

Depth and aggregation. In a fingerprint-augmented monomer model, Test20 R^2 falls with depth (0.9869 at L=1, 0.9644 at L=5) while Test1 is flat, and max aggregation is less stable than the sum (Table 15).

Table 14: **Ablations** (test R^2). (a) Latent node-feature map on Coley IP, n_{train}=300, block→block, PCA$_{64}$, 3 seeds, Test1 / Test20 (*fragment raw-z run without the open/cut bit). (b) Feature map on PoLyInfo-Small (PolyBC-F, all nine cells; †fitted without the open/cut bit, whose removal alone costs 0.013, panel c). (c) Closure and open/cut bit (all nine cells; Coley EA at Test1 / Test20; not bolded, closures differ at most 0.008; "n/a": monomer graphs have no cut bonds).

(a) Node feature, Coley IP (n_{train}=300)

Feature	Fragment	Monomer
raw z	0.427 / $< 0^*$	0.438 / 0.192
PCA(z)	0.641 / 0.637	**0.699 / 0.641**
PCA($\sigma(g(z))$)	**0.748 / 0.753**	0.636 / 0.625

(b) Feature map, PoLyInfo-Small

PCA($\sigma(g(z))$) (ours)	**0.707**
PCA($[\sigma(z), \sigma(g(z))]$)	0.606
PCA($[z, \sigma(g(z))]$)	0.467
PCA(z)†	0.307

(c) Closure and open/cut bit

Coley EA	PolyBC-F	PolyBC-M
Neumann bulk (ours)	0.9558 / 0.9554	0.9513 / 0.9487
no closure	0.9563 / 0.9550	0.9534 / 0.9471
canonical chain	0.9484 / 0.9484	0.9529 / 0.9529
w/o open/cut bit	0.9546 / 0.9544	n/a
PoLyInfo-Small	with bit	w/o bit
PolyBC-F	0.707	0.694

Closure on Coley IP. For PolyBC-M, the all-cell R^2 at Test1 / Test20 is 0.9365 / 0.9356 (Neumann bulk), 0.9400 / 0.9281 (no closure), and 0.9196 / 0.9196 (canonical chain). For PolyBC-F it is 0.9474 / 0.9452, 0.9470 / 0.9445, and 0.9328 / 0.9328, and validation selects the Neumann bulk for all three training rows. For the fragment model without the open/cut bit it is 0.9462 / 0.9444, 0.9471 / 0.9446, and 0.9330 / 0.9330; with the Neumann bulk this model reaches 0.9897 / 0.9881 in-domain and 0.9245 / 0.9226 in transfer. Selecting the closure per training row on validation R^2 changes the all-cell mean by up to 0.007 (fragment model without the bit on IP: 0.9462→0.9400 at Test1, 0.9444→0.9375 at Test20) and by at most 0.003 for the other models, and the choice depends on the selection rule; we therefore report the Neumann bulk closure for all Coley rows.

Feature map on full Coley data. A monomer model with PCA$_{64}$(z) nodes and otherwise the main recipe (Neumann bulk, L=1, width 300, Δt=0.5, 5 seeds) reaches in-domain 0.9948 / 0.9931 (EA) and 0.9940 / 0.9897 (IP) and transfer 0.9346 / 0.9305 (EA) and 0.9206 / 0.9103 (IP) at Test1 / Test20, compared with 0.9955 / 0.9941, 0.9955 / 0.9943, 0.9293 / 0.9259, and 0.9071 / 0.9062 for PolyBC-M. The distilled map is therefore slightly more accurate and more stable in-domain but transfers worse on monomer nodes of Coley.

Feature-map study on Coley IP (Table 14a). This study uses a reduced recipe, with PCA$_{64}$ features, a composition-weighted readout, Δt=1, no bulk closure and weaker distillation heads than the main models, so its absolute values are below those reported elsewhere and only the ordering between feature maps should be read from it. Its distillation heads are weaker than those of the main

models (4,000 epochs; in-sample Tanimoto 0.51 for monomers and 0.59 for fragments versus 0.91), which together with PCA_{64}, Δt=1, and the absent bulk closure may explain the lower absolute values. At n_{train}=1,000 the ordering is the same: fragments 0.895 / 0.881 with $\text{PCA}(\sigma(g(z)))$ versus 0.882 / 0.866 with $\text{PCA}(z)$, and monomers 0.716 / 0.702 versus 0.870 / 0.813. At full data, raw z on monomer nodes reaches 0.846 at Test1 but 0.657 at Test20 with a mean readout; $\text{PCA}_{64}(z)$ reaches 0.9955 / 0.9186 with a mean readout and 0.9954 / 0.9954 with our type-wise max and bulk closure, so stability at full data comes from the readout.

Depth, aggregation, and integration step. These ablations use a monomer model with Morgan-fingerprint, latent, and xTB node features on IP block→block. Table 15 reports depth and aggregation (5 seeds; 8 at L=5). For the integration step, Test1 / Test20 R^2 is 0.9878 / 0.9848, 0.9889 / 0.9854, 0.9882 / 0.9856, and 0.9844 / 0.9780 for $\Delta t \in \{0.25, 0.5, 1, 2\}$ at L=1 (the Δt=1 point is from the matched 3×3 sweep), and 0.9876 / 0.9820, 0.9872 / 0.9775, 0.9872 / 0.9673, and 0.9769 / 0.9364 at L=5: drift grows monotonically with Δt at L=5 and is flat for $\Delta t \leq 1$ at L=1, consistent with the forward-Euler view of Eq. (4). A matched depth ablation for both reported variants on Coley EA is given in Table 16; the corresponding sweep was not run on Coley IP, where we show the single block→block cell of Table 15.

Table 15: **Depth and aggregation** (fingerprint-augmented monomer model, IP block→block, Test1 / Test20).

Configuration	Test1	Test20
L=1, sum	0.9872	0.9869
L=2, sum	0.9875	0.9781
L=3, sum	0.9868	0.9695
L=5, sum	0.9856	0.9644
L=1, sum (matched runner)	0.9882	0.9856
L=1, max (matched runner)	0.9874	0.9651

Table 16: **Depth on Coley EA, both reported variants.** All-cell test R^2 (mean over the three in-domain and six transfer cells) for $L \in \{1, 2, 3\}$ under each bulk closure, evaluated at R=1 and R=20. L=1 is the configuration reported in Table 1; the deeper runs repeat it with only the layer count changed, on the same splits, caches and five seeds. Depth moves the all-cell figure by at most 0.003 and not monotonically, so one propagation layer is not a compromise on this dataset. The sweep was run on EA only.

	L=1		L=2		L=3	
Bulk closure	Test1	Test20	Test1	Test20	Test1	Test20
PolyBC-M						
Neumann	0.9513	0.9487	0.9539	0.9505	0.9525	0.9490
none	0.9534	0.9471	0.9521	0.9448	0.9501	0.9443
canonical	0.9529	0.9529	0.9541	0.9541	0.9547	0.9547
PolyBC-F						
Neumann	0.9558	0.9554	0.9556	0.9549	0.9544	0.9534
none	0.9563	0.9550	0.9551	0.9534	0.9553	0.9533
canonical	0.9484	0.9484	0.9480	0.9480	0.9485	0.9485

Isolating the effect of the boundary condition. The cross-architecture experiments in Table 1 alter both the chain sequence and the ghost rule simultaneously, making it impossible to evaluate the boundary condition on its own. Table 17 addresses this by separating the two: while keeping the chain structures, model checkpoints, and random seeds strictly fixed, we only change the closure rules at the two ends. The experiments show that the architecture-specified rule yields the highest accuracy for both PolyBC-M on block chains and PolyBC-F on random chains, and discarding the boundary condition is never the optimal choice.

Fragment versus monomer on PoLyInfo by monomer novelty. On the PoLyInfo-Full diagonals, PolyBC-F versus PolyBC-M scores 0.838 vs. 0.735 (random), 0.879 vs. 0.844 (alternating), and

Table 17: **Boundary-condition ablation on Coley EA** (R=1, five seeds, mean$_{\pm sd}$). $1 - R^2$ is the unexplained variance and the last column is its change relative to the prescribed boundary condition.

Ghost rule	R^2	RMSE (eV)	$1 - R^2$	$\Delta(1 - R^2)$
PolyBC-M, trained and tested on block chains				
periodic (h_0=h_N, h_{N+1}=h_1)	$0.9937_{\pm 0.0008}$	$0.0524_{\pm 0.0032}$	0.0063	+57%
Neumann (h_0=h_1, h_{N+1}=h_N) **(prescribed)**	$0.9960_{\pm 0.0002}$	$0.0418_{\pm 0.0013}$	0.0040	+0%
mean-field (h_0=h_{N+1}=$\bar{h}$)	$0.9943_{\pm 0.0007}$	$0.0495_{\pm 0.0031}$	0.0057	+43%
none (h_0=h_{N+1}=0)	$0.9943_{\pm 0.0004}$	$0.0498_{\pm 0.0019}$	0.0057	+43%
PolyBC-F, trained and tested on random chains				
periodic (h_0=h_N, h_{N+1}=h_1)	$0.9901_{\pm 0.0004}$	$0.0650_{\pm 0.0013}$	0.0099	+29%
Neumann (h_0=h_1, h_{N+1}=h_N)	$0.9917_{\pm 0.0003}$	$0.0596_{\pm 0.0013}$	0.0083	+8%
mean-field (h_0=h_{N+1}=$\bar{h}$) **(prescribed)**	$0.9923_{\pm 0.0002}$	$0.0574_{\pm 0.0008}$	0.0077	+0%
none (h_0=h_{N+1}=0)	$0.9898_{\pm 0.0010}$	$0.0660_{\pm 0.0031}$	0.0102	+32%

0.890 vs. 0.784 (block) on test rows with at least one monomer absent from the training rows of that architecture, and 0.928 vs. 0.930, 0.881 vs. 0.859, and 0.343 vs. 0.256 on the remaining rows. With the definition of Table 3 (absent from all training rows), the random diagonal gives 0.815 vs. 0.794 (unseen) and 0.916 vs. 0.823 (seen); the difference comes from four rows, three with T_g of 430 to 460 °C, so the random-diagonal pattern is fragile. The seen-monomer block rows have low label variance.

F Enumerated T_g Benchmark

The enumerated T_g set is built from the binary-copolymer T_g study of Bhati et al. (2026) and contains all $\binom{315}{2} = 49{,}455$ pairs of 315 monomers at $f_A \in \{0.25, 0.5, 0.75\}$ (148,365 rows, all random), with $T_g \in [-74.1, 381.5]$ °C (standard deviation 83.7 °C). Its labels are predictions of an upstream surrogate that ships with per-row uncertainty estimates (mean 16.9 °C), not measurements. The pair split (89,016/14,838/44,511) leaves no test monomer unseen. We train on random chains with R=1 and evaluate at $R \in \{1, 5, 10, 20\}$. For this set we reuse the PoLyInfo encoder, g, and a PCA_{512} basis fitted on the 739 PoLyInfo fragments (99.94% of variance); 178 of the 255 fragments lie outside the fit set of g. We use L=2, width 512, a Huber loss (δ=5 °C), batch 256, and 400 epochs; the monomer node is the mean of its fragment features.

Table 18: **Enumerated T_g, random→random** (44,511 test rows). Single-model mean over 5 seeds; "Drift": root-mean-square change of the 5-seed ensemble prediction relative to R=1. GRIN values from our runs on the same split ($\{1, 3\}$ augmentation; Appendix C.1).

	Test1		Test5		Test10		Test20	
Model	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE
GRIN (Zhu et al., 2025)	0.9989	2.71	0.9862	9.03	0.9848	10.21	0.9828	10.85
PolyBC-M	**0.9999**	**0.90**	**0.9968**	**4.73**	0.9948	6.00	0.9938	6.55
PolyBC-F	**0.9999**	0.96	0.9963	5.11	**0.9957**	**5.48**	**0.9955**	**5.61**
Drift M / F (°C)	n/a		4.38 / 4.86		5.67 / 5.25		6.21 / 5.38	

Table 18 shows that both variants are more accurate than GRIN at every repeat number, with 2.8 to 3.0× lower RMSE at R=1 and 1.7 to 1.9× lower at R=20, and that the fragment model drifts less at large R. Repetition invariance nonetheless does not hold here in the per-row sense: RMSE grows from about 0.9 to between 5.6 and 6.6 °C, and the drift concentrates on f_A=0.5 rows, whose R=1 realization contains only A-B contacts. The benchmark has limited discriminative power: test RMSE at R=1 is far below the surrogate's own stated uncertainty, and a graph-free regressor with monomer-identity embeddings reaches R^2=0.99991 at R=1 (one seed). Since every row is random, forcing alternating or block sequences at test time scores the same random-copolymer labels; for an earlier random-trained PCA_{256}, L=1 fragment model the per-row RMS change was 1.3 °C to alternating and 2.2 °C to block, which measures prediction stability under sequence forcing rather

than architecture transfer. Depth does not affect accuracy at R=1 (all $L \in \{1, 2, 3, 4\}$ within 10^{-5} R^2; 3 seeds each), but at R=20 the shallowest model drifts least (ensemble drift 5.05 °C at L=1 vs. 5.44 to 5.49 °C at $L \geq 2$), in line with Lemma 1. Training five seeds takes 7.0 h for the fragment model and 6.5 h for the monomer model on a single shared GPU; fragment graphs are 3.3× larger than monomer graphs but cost only 7% more time.

G Inverse-Design Details

Design model. The design model is PolyBC-M exactly as reported in Table 1: monomer nodes, latent node features $x = W_p^\top(\sigma(g(z)) - \mu)$ with p=512, one propagation layer of width 300, Δt=0.5, type-wise max readout, Neumann bulk closure, trained on single-repeat chains of one architecture. No separate surrogate is fitted. The design objective is the trained model's own prediction, and the five training seeds of that cell are used to re-score candidates. Because x is a fixed differentiable function of z, the gradient $\nabla_z \hat{y}$ exists and reaches all 768 latent dimensions.

Optimization. The design variables are the 768-dimensional latents z_A, z_B of the two repeat units; composition and architecture are held fixed. We minimize $\mp\hat{y}(z_A, z_B) + \rho(\|z_A - z_A^0\|^2 + \|z_B - z_B^0\|^2)$, the upper sign for maximization, with Adam for 250 steps at $\eta \in \{10^{-3}, 3 \times 10^{-3}\}$ and ρ=0.5, from 28 training copolymers stratified over the three architectures and all compositions, under three modes: optimize B with A fixed, optimize A with B fixed, and optimize both. The trust region is what makes the loop usable. On IP-maximizing block copolymers, raising ρ from 0 to 0.5 moves the share of iterates that decode to a valid repeat unit from 8.0% to 54.8% at η=10^{-3}, from 2.9% to 66.1% at η=3×10^{-3}, and from 0.0% to 27.5% at η=10^{-2}; we therefore drop η=10^{-2} and ρ=0. Averaged over the sweep, 37.1% of iterates decode. Every iterate is decoded by the frozen Poly4mer decoder; a string is *valid* if it parses and carries exactly two attachment points, and *novel* if it is not one of the 683 Coley monomers.

Three numbers per design. A latent that scores well is not a molecule, so we report three changes for every design, each relative to its starting copolymer. Δ_{latent} is the model's prediction at the optimized latent, which is what the optimizer maximizes. Δ_{mol} is the same model's prediction after the decoded string is re-encoded, averaged over the five checkpoint seeds; this is the prediction that belongs to a molecule that exists. Δ_{xTB} is an independent oracle. Signed toward the objective, Δ_{latent}=+0.78 eV against Δ_{mol}=+0.14 eV, so 18% of the gain the optimizer sees survives the round trip through a real molecule. This gap, rather than the optimizer or the property model, is what limits latent-space design.

Oracle. Δ_{xTB} is computed with the protocol that produced the training labels (Bannwarth et al., 2019): attachment points are capped with hydrogen, a conformer is embedded and force-field relaxed, and the vertical ionization potential and electron affinity are evaluated with GFN2-xTB after a loose geometry optimization, using an implicit benzene solvent model at neutral charge and closed shell. On the 25 designed molecules that also appear among the training labels this reproduces the stored values to within 0.005 eV. The oracle scores the designed repeat unit, the unit the optimizer edits, while the two model predictions are chain-level quantities; we therefore compare them by the sign of the change rather than by its magnitude. Designs that decode to a charged or radical species are excluded from every oracle figure, since the neutral closed-shell calculation does not describe them; this removes 73 of 566 novel designs.

Generative baselines. Table 19 compares three published generators under one protocol: G2S-VAE (Vogel & Weber, 2025), trained on the same copolymers as our model, and the MOSES (Polykovskiy et al., 2020) VAE and AAE. Published MOSES checkpoints are fitted to a different polymer corpus, so their novelty is not measurable against the set our design loop starts from; we therefore retrain both architectures on the 683 Coley monomers using the official code package from the authors at its published hyperparameters (VAE: latent dimension 128, 100 epochs, KL weight annealed to 0.05; AAE: latent dimension 128, 120 epochs). Both retrained models memorize: over 5,000 samples the VAE emits three distinct repeat units, all in its training set, and the AAE emits one. This is what 683 training molecules buys, and it is the honest baseline for this task; the asymmetry that remains in our favor is that our decoder is pretrained on a large polymer corpus and never has to learn this chemistry from 683 examples. All three are sampled from their

prior and decoded greedily, and G2S-VAE's copolymer strings are split into their two repeat units so that every row is scored on the same unit. We also examined G2S-VAE's released checkpoint under an around-seed protocol and found its latent collapsed: four very different copolymers encode to within 2.5×10^{-3} of one another and reconstruct to one identical string, so that protocol carries no information about the seed and is not reported.

Table 19: **Generative quality of the design loop and of the generative baselines.** Every method is scored on 5 replicates of 1,000 decoded repeat-unit strings; $\text{mean}_{\pm\text{sd}}$ over replicates. Baseline replicates are independent prior draws; ours are disjoint groups of starting copolymers, so no starting point and no optimization trajectory is shared between replicates, although the same molecule can be reached from different starting points. All decoding is greedy. Validity: the string parses *and* carries exactly two attachment points. Uniqueness: distinct canonical SMILES among the valid ones. Distinct: how many that is, per 1,000 samples. Novelty: absent from the generator's own training corpus, named in the second column. Our rows count *every* optimizer iterate, including latents deliberately pushed off the manifold, which is why validity is low; the baselines sample from a prior they were trained to match. Both MOSES architectures were retrained on the 683 Coley monomers, since the released checkpoints were fitted to a different corpus and their novelty would not be comparable. A novelty percentage computed over one or two distinct molecules, as for the retrained AAE, carries no information.

Method	Training corpus	Validity (%)	Uniqueness (%)	Distinct	Novelty (%)
Generative baselines (prior sampling, greedy decoding)					
G2S-VAE (Vogel & Weber, 2025)	Coley (42,966 copolymers)	$100.0_{\pm 0.0}$	$0.5_{\pm 0.0}$	$5.0_{\pm 0.0}$	$0.0_{\pm 0.0}$
MOSES VAE (Coley) (Polykovskiy et al., 2020)	Coley (683 monomers)	$46.3_{\pm 1.6}$	$0.5_{\pm 0.1}$	$2.2_{\pm 0.4}$	$0.0_{\pm 0.0}$
MOSES AAE (Coley) (Polykovskiy et al., 2020)	Coley (683 monomers)	$14.3_{\pm 0.8}$	$0.7_{\pm 0.0}$	$1.0_{\pm 0.0}$	$100.0_{\pm 0.0}$
Ours: every decoded iterate of the gradient design loop					
PolyBC-M design, optimize B	Coley (683 monomers)	$52.2_{\pm 8.4}$	$4.0_{\pm 1.1}$	$20.6_{\pm 6.2}$	$67.6_{\pm 9.2}$
PolyBC-M design, optimize A	Coley (683 monomers)	$30.3_{\pm 8.6}$	$4.8_{\pm 2.1}$	$13.0_{\pm 3.3}$	$56.5_{\pm 4.2}$
PolyBC-M design, optimize A and B	Coley (683 monomers)	$29.9_{\pm 4.6}$	$10.6_{\pm 2.1}$	$31.2_{\pm 5.8}$	$65.1_{\pm 6.7}$

Table 20: **Oracle-validated design with PolyBC-M.** All 493 novel, neutral, radical-free designs of the 72-cell sweep, grouped by objective and by which repeat unit is optimized. n: designs; mol.: distinct molecules among them. Δ_{latent}: the model's prediction at the optimized latent, minus its prediction at the starting copolymer. Δ_{mol}: the same, after the decoded string is re-encoded (mean over the 5 checkpoint seeds). Δ_{xTB}: the independent GFN2-xTB oracle on the designed repeat unit. Agree: share of designs whose Δ_{xTB} has the intended sign; 50% is chance. All entries $\text{mean}_{\pm\text{sd}}$ over designs, in eV. ↑ maximize, ↓ minimize.

Objective	n	mol.	Δ_{latent}	Δ_{mol}	Δ_{xTB}	Agree
EA						
↑ optimize B	47	33	$+1.12_{\pm 0.59}$	$-0.02_{\pm 0.35}$	$+0.44_{\pm 0.57}$	87%
↑ optimize A	26	13	$+0.79_{\pm 0.56}$	$+0.03_{\pm 0.16}$	$+0.63_{\pm 0.54}$	88%
↑ optimize A and B	27	13	$+1.15_{\pm 0.61}$	$+0.06_{\pm 0.25}$	$+0.44_{\pm 0.45}$	89%
↓ optimize B	50	21	$-0.53_{\pm 0.42}$	$-0.06_{\pm 0.15}$	$-0.15_{\pm 0.43}$	64%
↓ optimize A	45	21	$-0.46_{\pm 0.65}$	$-0.11_{\pm 0.27}$	$+0.24_{\pm 0.50}$	24%
↓ optimize A and B	51	19	$-0.76_{\pm 0.47}$	$-0.06_{\pm 0.23}$	$+0.04_{\pm 0.55}$	45%
IP						
↑ optimize B	37	16	$+0.65_{\pm 0.37}$	$+0.37_{\pm 0.42}$	$+0.06_{\pm 0.61}$	68%
↑ optimize A	7	6	$+1.00_{\pm 0.48}$	$+0.31_{\pm 0.27}$	$-0.33_{\pm 0.49}$	29%
↑ optimize A and B	19	8	$+1.11_{\pm 0.30}$	$+0.29_{\pm 0.57}$	$+0.09_{\pm 0.59}$	68%
↓ optimize B	57	20	$-0.61_{\pm 0.40}$	$-0.14_{\pm 0.23}$	$-0.48_{\pm 0.38}$	89%
↓ optimize A	55	16	$-0.55_{\pm 0.42}$	$-0.14_{\pm 0.30}$	$-0.70_{\pm 0.80}$	62%
↓ optimize A and B	72	18	$-1.05_{\pm 0.70}$	$-0.25_{\pm 0.38}$	$-0.44_{\pm 0.55}$	68%

The oracle confirms the two objectives the decoder can reach by substitution: maximizing electron affinity, at 88% sign agreement and $+0.49$ eV, and minimizing ionization potential, at 73% and -0.53 eV. Both are reached by substituent swaps on a conserved aromatic ring, and the direction of

the swap is the one chemistry predicts: electron-withdrawing groups for EA ↑, where replacing two ring hydroxyls with two fluorines raises the oracle EA by 0.75 eV, and electron-donating groups for IP ↓, where replacing two methyls with two amines lowers the oracle IP by 1.48 eV. These edits are one or two tokens apart in the decoder's vocabulary, which is why the loop finds them. Optimizing B yields more distinct molecules than optimizing A (33 against 13 on EA ↑), as the corpus contains 682 distinct B monomers against 9 distinct A monomers and the B latent therefore sits in a far better covered region of the decoder's manifold. Table 20 reports the mean over all designs rather than the best one, because the extreme values of the sweep come from scaffolds the decoder invents rather than from edits of this kind.

Figure 2 shows two of these trajectories in full. In each, the model's prediction at the latent rises or falls smoothly and by much more than the prediction at the molecules that latent decodes to, while the oracle follows the intended direction on its own scale. That gap, and not the optimizer or the property model, is what limits latent-space design here.

G.1 A SECOND DESIGN STUDY: T_g ON POLYINFO

We repeated the procedure on PoLyInfo, where the target is a measured property rather than a computed one, to test whether the same loop generates usable polymer chemistry when the property model is fitted to experiment.

Design model. The design model is the PolyBC-M row of Table 2. Its node feature is the one that model was trained on, a 256-dimensional projection of the distillation head's output, and each architecture keeps the bulk closure it was trained with: mean field for random chains, none for alternating, Neumann for block. The optimization protocol is the one used on Coley, with 12 starting copolymers per architecture stratified over composition, both objectives, and the same three modes.

Scope. The PoLyInfo model represents a monomer by the mean of its fragment features, which is a function of that monomer's own latent only when the monomer consists of a single fragment. Design is therefore restricted to the 248 of 956 monomers that do, which leaves 847 random, 183 block and 135 alternating records available as starting points.

Measuring a change. On PoLyInfo the stored latents and the decoder disagree about how a molecule is written: the corpus stores monomers in non-canonical form while a decoded molecule can only be re-encoded in canonical form, and the encoder is sensitive to that difference. Comparing a re-encoded design against a stored seed latent therefore mixes two measurement paths and adds a spurious shift of tens of degrees. We put both ends of the molecule-level change on the re-encode path, $\Delta_{\text{mol}} = \hat{y}(\text{re-encoded design}) - \hat{y}(\text{re-encoded seed})$, so that this shift cancels. The Coley study is unaffected, its corpus already being canonical.

No oracle. GFN2-xTB produced Coley's labels and can therefore check its designs. Measured glass-transition temperature can be confirmed only by synthesis and measurement, which lies beyond this study. The one group-contribution estimate available to us is not a substitute: composition-weighted over the full corpus it attains $R^2 = -0.705$ against the measured values, worse than predicting the mean, and it ranks only weakly on the subset used here. We therefore evaluate the PoLyInfo designs by generative quality and report no oracle column.

Generative quality. Table 21 scores every method on five replicates of 1,000 decoded repeat-unit strings, with replicates formed from disjoint groups of starting copolymers. As on Coley, we retrained both MOSES architectures on the corpus the task is defined against. The design loop returns 15 to 29 distinct novel repeat units per 1,000 samples, against 2.0 for the retrained VAE and none for the retrained AAE, which collapses onto a single long alkane. Our validity is lower because our rows count every optimizer iterate, including latents deliberately pushed off the manifold, whereas the baselines sample a prior they were trained to match. Figure 3 shows three design trajectories in detail; the largest molecule-level change among them is $+178$°C, reached in seven steps from a block copolymer of poly(ethylene oxide) and a polycarbonate unit.

H COMPUTATIONAL EFFICIENCY

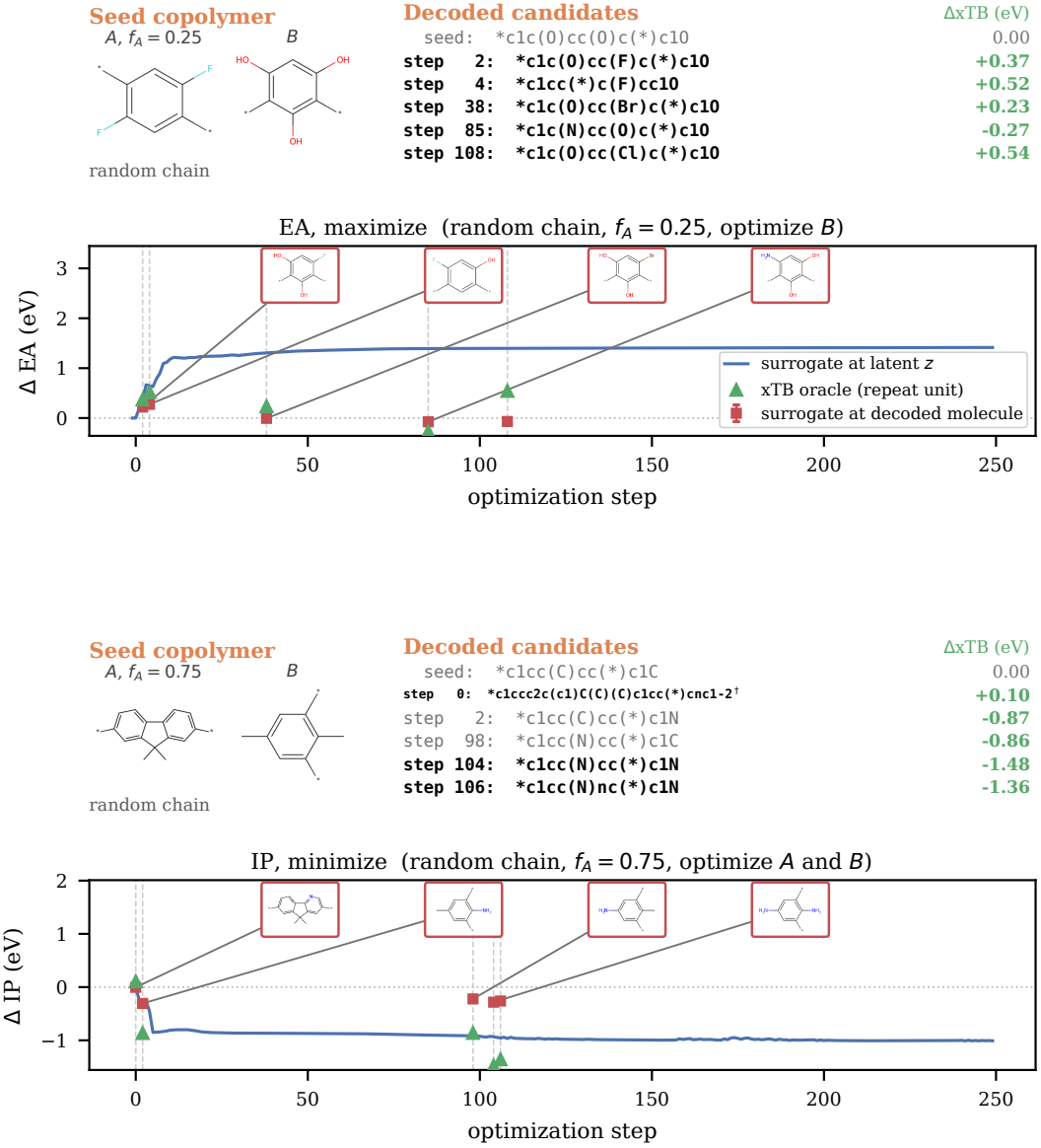

Figure 2: **Inverse design with PolyBC-M on the two objectives the oracle confirms.** Top, maximizing electron affinity from a random chain by optimizing B; bottom, minimizing ionization potential from a random chain by optimizing both units. Left: the starting copolymer, monomer A beside monomer B. Middle: at each step where a decode changes, the repeat unit that changed and that unit's own oracle change; bold marks a unit absent from the 683 training monomers and † marks a row showing monomer A. Right: the trajectory. The line is the model's prediction at the current latent, the squares are the same model's prediction after the decoded molecule is re-encoded (error bars: sd over the five checkpoint seeds), and the triangles are the GFN2-xTB oracle on the designed repeat unit. The first two are chain-level predictions and the third a unit-level oracle, so they are not expected to coincide; the gap between line and squares measures what the decode–re-encode round trip loses, the component a stronger decoder would improve. These are single trajectories chosen to be legible; the aggregate is Table 20.

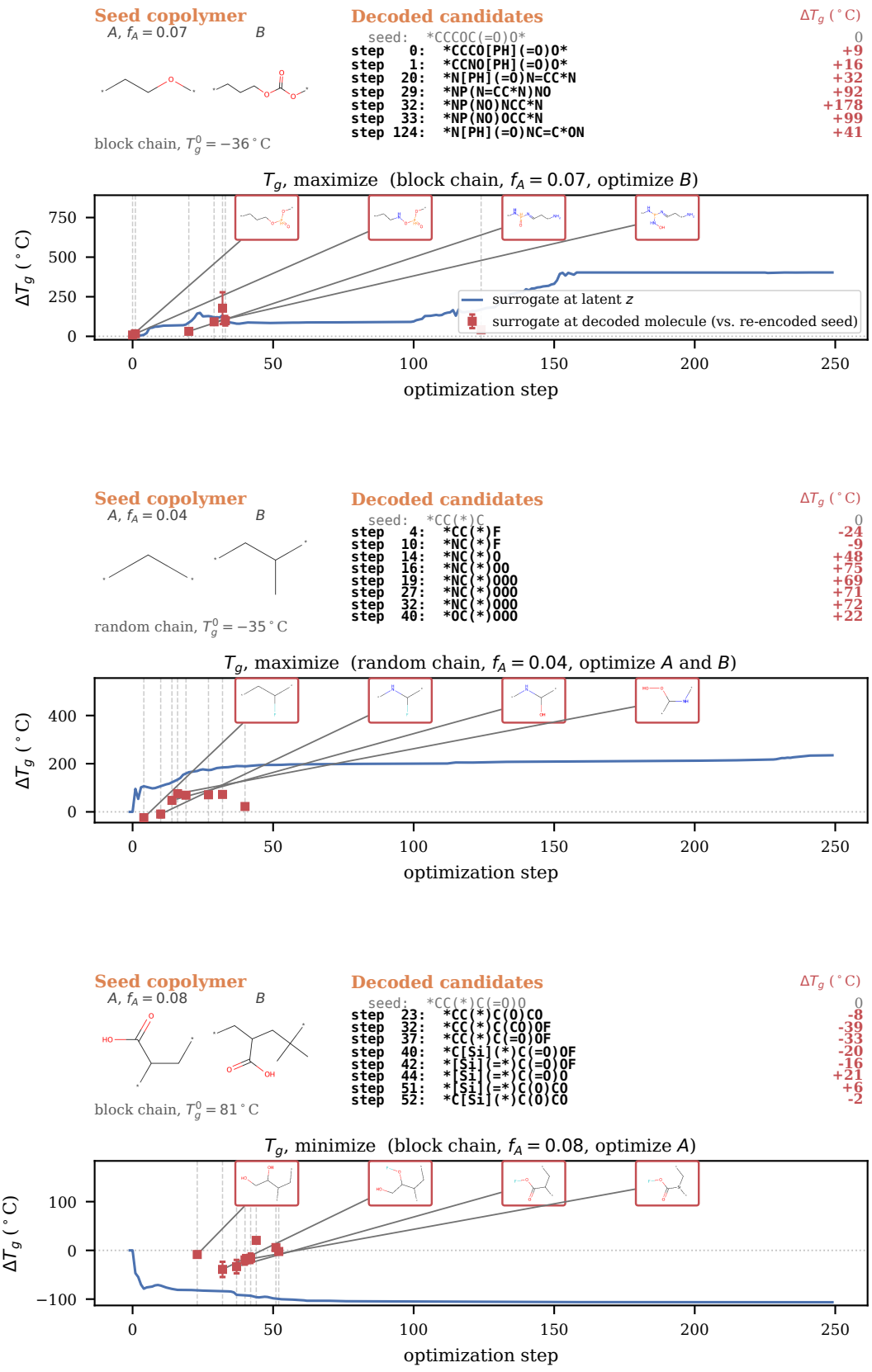

Figure 3: **Inverse design of T_g on PoLyInfo**, drawn as in Figure 2. Top, maximizing from a block chain by optimizing B; middle, maximizing from a random chain by optimizing both units; bottom, minimizing from a block chain by optimizing A. Left: the starting copolymer. Middle: at each step where a decode changes, the repeat unit that changed and its molecule-level change; bold marks a unit absent from the 956 PoLyInfo monomers. Right: the trajectory, with the line the model's prediction at the current latent and the squares the same model's prediction after the decoded molecule is re-encoded, both measured on the re-encode path.

Table 21: **Generative quality on PoLyInfo.** Every method is scored on 5 replicates of 1,000 decoded repeat-unit strings; mean$_{\pm sd}$ over replicates. Baseline replicates are independent prior draws; ours are disjoint groups of starting copolymers, grouped on the distinct monomer pair, composition and architecture rather than on the table row, because the corpus repeats polymers and the design loop is deterministic given that triple. Columns are as in Table 19, and both MOSES architectures were retrained on the same 956 monomers so that novelty is measured against one reference throughout. A novelty percentage computed over two distinct molecules, as for the retrained VAE, carries no information, and none is defined for the retrained AAE.

Method	Training corpus	Validity (%)	Uniqueness (%)	Distinct	Novelty (%)
Generative baselines (prior sampling, greedy decoding)					
MOSES VAE (Polykovskiy et al., 2020)	PoLyInfo	$27.4_{\pm 1.2}$	$0.7_{\pm 0.0}$	$2.0_{\pm 0.0}$	$100.0_{\pm 0.0}$
MOSES AAE (Polykovskiy et al., 2020)	PoLyInfo	$0.0_{\pm 0.0}$	$0.0_{\pm 0.0}$	$0.0_{\pm 0.0}$	$0.0_{\pm 0.0}$
Ours: every decoded iterate of the gradient design loop					
PolyBC-M design, optimize B	PoLyInfo	$7.0_{\pm 4.2}$	$28.0_{\pm 15.3}$	$15.2_{\pm 5.4}$	$63.6_{\pm 13.7}$
PolyBC-M design, optimize A	PoLyInfo	$16.7_{\pm 7.0}$	$19.8_{\pm 5.6}$	$29.2_{\pm 7.0}$	$80.2_{\pm 3.0}$
PolyBC-M design, optimize A and B	PoLyInfo	$11.6_{\pm 3.5}$	$26.1_{\pm 6.7}$	$29.0_{\pm 6.5}$	$72.8_{\pm 7.9}$

Table 22 reports peak GPU memory and training time per epoch for every model on Coley EA, measured on a single NVIDIA RTX 5090 with one seed per model, block-architecture chains, and each method's published hyperparameters. Peak memory is the tensor high-water mark of the training process. Time per epoch is obtained by running each model twice with the same warm cache, once for E_1 and once for $E_2 > E_1$ epochs, and taking $[T(E_2) - T(E_1)]/(E_2 - E_1)$; this cancels start-up, data construction and the final evaluation without instrumenting twelve different training loops, and E_1 is set above any warm-up phase a method has so that the slope measures the steady state. We report mean$_{\pm sd}$ over three paired measurements.

The two columns say different things. On memory our models are small: 133 and 154 MiB, against 1,304 for GRIN, which trains on twice the graphs, 1,392 for GREA, and 328 to 610 for the chemical language models. On time they are mid-pack: 3.6 and 3.7 s per epoch is faster than CIGA, DIR, GRIN and DISGEN, slower than GSAT, SSR and RP-GIN, and five to eleven times slower than the language models. That last comparison needs a caveat, since all four language models train only a small head on embeddings their frozen encoder computes once, before the epoch loop; the estimator cancels that pass along with every other fixed cost, so their rows are marginal epoch costs rather than end-to-end costs. polyBERT, for instance, spends 201 s on that single encoding against a 0.55 s marginal epoch. Four atom-level baselines allocate less memory than ours and three are also faster, but all of them except DISGEN reach negative R^2 at R=20; that, and not the cost, is the trade the table is meant to expose.

Two further caveats apply. An epoch is not the same amount of work in every row: GRIN sees twice as many training chains because it augments with additional repeat counts, the two chain-level language models see slightly fewer rows because some monomer pairs cannot be stitched, and batch sizes differ between methods. And where a model's fixed cost is large next to its marginal epoch the slope estimate is noisy. DISGEN is the extreme case: its start-up is roughly 370 s against a marginal epoch of a few seconds, so at E_2=30 its three paired measurements ranged from 2.4 to 17.6 s per epoch. We therefore measured that row alone at E_2=60, which widens the difference from 20 to 50 epochs and brings the spread from 81% to 22% of the mean; its entry is the only one in the table obtained with that budget. The comparison is therefore indicative of order of magnitude rather than controlled.

Table 22: **Computational cost on Coley EA.** One NVIDIA RTX 5090, one seed per model, block-architecture chains, published hyperparameters. Time per epoch is the slope of total runtime against epoch budget, mean$_{\pm \mathrm{sd}}$ over three paired measurements. Epochs are not equal work across rows and batch sizes differ between methods; the language-model rows exclude a one-time encoder pass. Both caveats are given in the text.

Model	Peak GPU memory (MiB)	Training time (s / epoch)
Atom-level graph models		
GREA (Liu et al., 2022)	1392	$1.18_{\pm 0.08}$
GSAT (Miao et al., 2022)	70	$1.67_{\pm 0.09}$
SSR (Buffelli et al., 2022)	67	$2.94_{\pm 0.20}$
RP-GIN (Murphy et al., 2019)	68	$3.08_{\pm 0.09}$
CIGA (Chen et al., 2022)	289	$4.30_{\pm 0.37}$
DIR (Wu et al., 2022)	306	$9.97_{\pm 0.32}$
GRIN‡ (Zhu et al., 2025)	1304	$13.12_{\pm 1.34}$
DISGEN (Huang et al., 2024)	24	$16.75_{\pm 3.73}$
Chemical language models (frozen encoder + head)		
MoLFormer (Ross et al., 2022)	330	$0.33_{\pm 0.03}$
TrinityLLM (Liu et al., 2025)	330	$0.47_{\pm 0.20}$
polyBERT (Kuenneth & Ramprasad, 2023)	328	$0.55_{\pm 0.09}$
TransPolymer (Xu et al., 2023)	610	$0.71_{\pm 0.21}$
Ours		
PolyBC-M	133	$3.62_{\pm 0.34}$
PolyBC-F	154	$3.73_{\pm 0.40}$