Property-Guided Generation of Complex Polymer Topologies Using Variational Autoencoders

Shengli Jiang[1], Adji Bousso Dieng[2,3*], and Michael A. Webb[1*]

[1]Department of Chemical and Biological Engineering, Princeton University, Princeton, NJ

[2]Department of Computer Science, Princeton University, Princeton, NJ

[3]Vertaix

*Corresponding Authors: adji@princeton.edu, mawebb@princeton.edu

Keywords: structure-property relationships, machine learning, polymer architecture, generative modeling, multi-task learning, rheology

Abstract

The complexity and diversity of polymer topologies, or chain architectures, present substantial challenges in predicting and engineering polymer properties. Although machine learning is increasingly used in polymer science, applications to address architecturally complex polymers are nascent. Here, we use a generative machine learning model based on variational autoencoders and data generated from molecular dynamics simulations to design polymer topologies that exhibit target properties. Following the construction of a dataset featuring 1,342 polymers with linear, cyclic, branch, comb, star, or dendritic structures, we employ a multi-task learning framework that effectively reconstructs and classifies polymer topologies while predicting their dilute-solution radii of gyration. This framework enables the generation of novel polymer topologies with target size, which is subsequently validated through molecular simulation. These capabilities are then exploited to contrast rheological properties of topologically distinct polymers with otherwise similar dilute-solution behavior. This research opens new avenues for engineering polymers with more intricate and tailored properties with machine learning.

1 Introduction

The topology of a polymer chain, or equivalently the chain architecture, can substantially influence their properties and those of derivative materials. For example, in natural polymers, while linear

amylose forms dense aggregates with low aqueous solubility, the analogous but highly branched structure of amylopectin impedes association of chains, thereby enhancing its solubility [1]. In the realm of synthetic polymers, the branching in low-density polyethylene improves its processability for applications like blow and extrusion molding, whereas linear high-density polyethylene possesses superior mechanical strength and chemical resistance. There is also growing interest in understanding implications of polymer topology due to advancements in various controllable synthetic methodologies [2–5]. These methods enable the creation of polymers with a wide range of complex topologies, such as stars [6, 7], combs [8, 9], branches [10, 11], hyperbranches [12, 13], dendrimers [14, 15], rings [16, 17], and brushes [18, 19].

Establishing quantitative relationships between polymer topology and material properties remains challenging. Both experimental and computational investigations have enhanced understanding of how polymer topology influences properties of interest to many areas, such as enhanced oil recovery [20, 21], coatings and adhesives [22, 23], rheology and fluid dynamics [24–26], energy storage [27–33], and biomedical applications [34–38]. Nevertheless, the efforts of labor-intensive and potentially costly synthesis and characterization typically limits experimental studies to a small set of systems, which may still not yield well-defined topological ensembles [3, 39]. Computationally, although there is no ambiguity associated with the underlying topologies of the polymers or their construction, simulations are often restricted to a particular class of topologies owing to computational costs and perhaps uncertainty with how to tangibly compare diverse topologies [40, 41]. Overall, these factors obfuscate the construction of general topology-property correlations, which also precludes facile design of topologically complex polymers.

Recent advancements in and applications of machine learning have spurred significant developments in polymer design. These efforts span many applications, such as tailoring the structures of single-chain nanoparticles [42, 43], enhancing enzyme stability [44, 45], delivering drugs and therapeutics [46–48], and identifying new gas-separation membranes [49]. Generative machine learning models [50] are a particularly intriguing class of algorithms for chemical design. For example, variational autoencoders (VAEs) are adept at encoding complex data into lower-dimensional latent spaces [51, 52] and have previously facilitated the generation of novel small molecules [53, 54]. Applications of VAEs in polymer science are also emerging [55, 56]. Shmilovich et al. combined VAEs with molecular dynamics (MD) simulations and Bayesian optimization to guide the discovery of π-conjugated oligopeptides [57] with desirable aggregation behavior to influence optoelectronic properties. In devising the Open Macromolecular Genome (OMG), Kim et al. utilized a generative framework with VAEs that can not only provide polymer structures but also retrosynthesis [58], thereby

facilitating optimization of synthetically accessible materials. Nevertheless, these and other studies primarily focus on specific chemical spaces or linear polymers, highlighting the need for methods to generate polymers with complex topologies and tailored properties.

In this study, we create a multi-task VAE to generate polymers with specified topology and desired characteristics. This model is developed using an original dataset comprising coarse-grained MD data for over 1,300 polymers of various topologies, including star, comb, branch, linear, cyclic, and dendrimer structures, spanning a range of molecular weights. Input and encoding strategies are critically assessed by training several models that aim to reconstruct the polymer topology and also perform auxiliary tasks of estimating the characteristic size of the polymer and classifying its topology. We find that auxiliary tasks enhance the physical interpretability of the learned latent space of the VAE, and our most effective generative modeling framework, `TopoGNN`, incorporates both graph and topological descriptor features. For demonstrative purposes, `TopoGNN` is leveraged to produce sets of topologically diverse polymers that exhibit the same characteristic size in dilute solution (Figure 1, top) but contrasting rheological behavior at finite concentrations (Figure 1, bottom). This work expands the utility of generative modeling for polymer design and demonstrates how such algorithms can also facilitate controlled studies across complex, topologically diverse polymers.

2 Results

2.1 Polymer Dataset

We first generate and characterize a topologically diverse set of polymers for training and evaluating the VAE. In particular, we initially prepare and simulate 1,342 polymers across six architectural classes (11 each for linear and cyclic and 330 each for branch, comb, star and dendrimer); the degree of polymerization ranges from 90 to 100 for each architectural class. Figure 2a showcases the diversity of structures across a representative set of these polymers. This diversity is also manifest through the variation of topological descriptors shown in Figure 2b. These descriptors, which are derived purely from knowledge of the molecular graph/polymer connectivity, provide a first means to quantitatively characterize and distinguish polymer topologies. Despite the uniformity in the number of nodes and edges, which are commonly used to characterize polymers, significant variations are observed in other topological descriptors. For instance, comb, branch, star, and dendrimer topologies, exhibit notable differences in descriptors like graph diameter, radius, betweenness centrality, and degree assortativity, even when node and edge counts are identical. Coarse-grained simulations of the

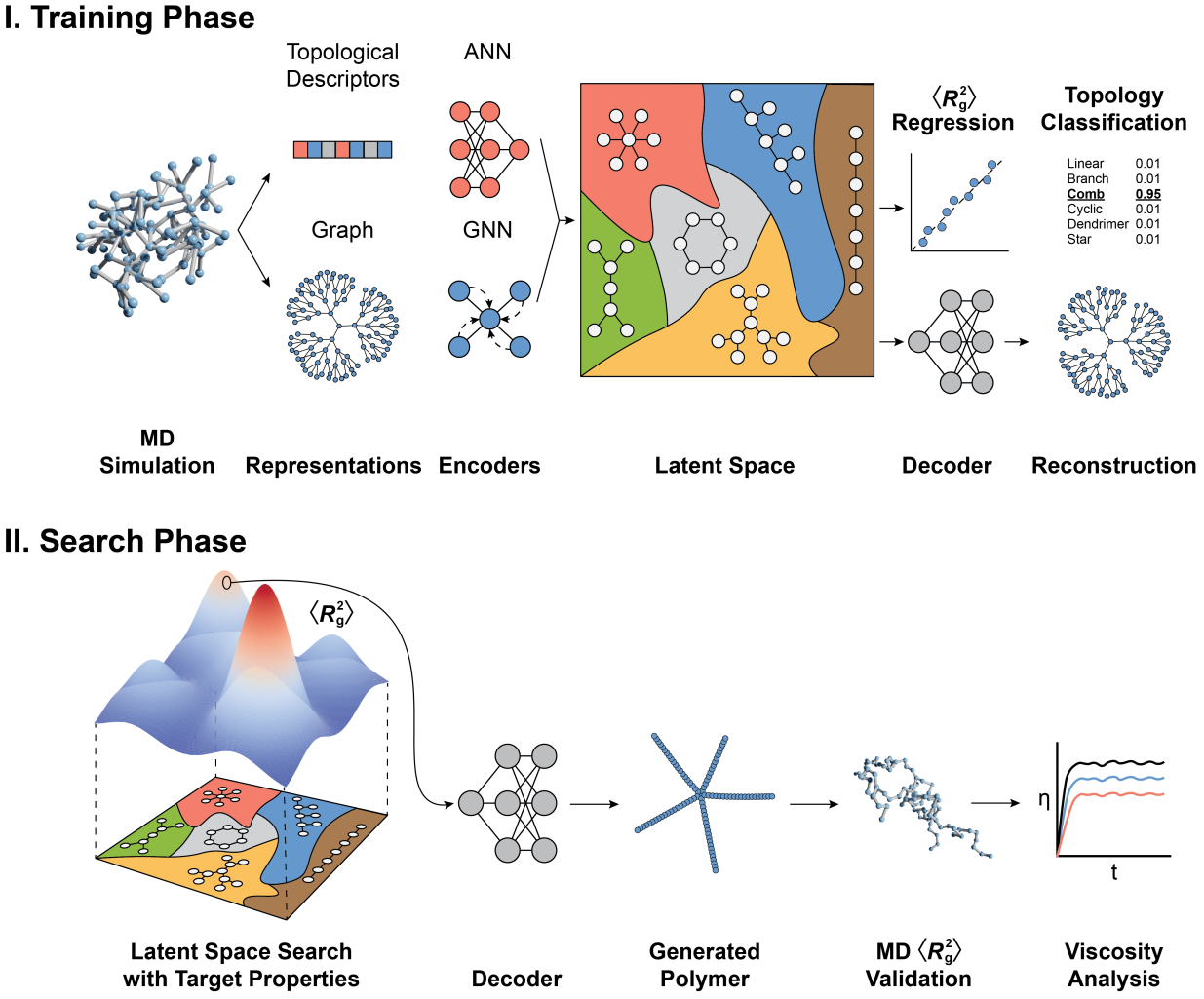

Figure 1: **Strategy underlying a variational autoencoder of polymer topology**. In the Training Phase (top), molecular dynamics (MD) simulations are employed to compute computationally tractable descriptors, such as the average squared radius of gyration, $\langle R_g^2 \rangle$, for a set of polymers. Information regarding topological descriptors and the polymer graph are then encoded into a lower-dimensional latent space using an artificial neural network (ANN) and a graph neural network (GNN). The latent space is decoded to accomplish reconstruction, regression, and classification tasks. These encoded features are concatenated to form a reduced-dimensional latent space, from which a decoder reconstructs the polymer structure. In the Search Phase (bottom), points are sampled from the latent space to proffer polymers that are predicted to exhibit a target $\langle R_g^2 \rangle$ and specified topology. These predictions are evaluated against MD simulations, and post-validation, enable systematic analysis of how topology impacts additional properties, such as viscosity.

generated polymers further distinguish topologies on the basis of their physical characteristics; the use of a Kremer-Grest model permits us to attribute all property variations to the molecular weight (i.e., the number of composite coarse-grained polymer beads) and the topology (i.e., the connectivity of such beads). Figures 2c and S1 illustrate the range of characteristic polymer sizes, as expressed through the mean squared radius of gyration $\langle R_g^2 \rangle$, observed in each class. Because the present study imposes a maximum number of monomers, polymers from the linear, cyclic, and dendrimer classes exhibit relatively narrow distributions in $\langle R_g^2 \rangle$ by contrast to comb, branch, and star classes. Dendrimers notably form compact, globular structures over the range of simulated molecular weights relative to all other classes. Overall, the dataset is partitioned into a 64/16/20 train/validation/test split for future model construction and evaluation; stratified sampling is used to ensure proportional representation of architectural classes across all splits.

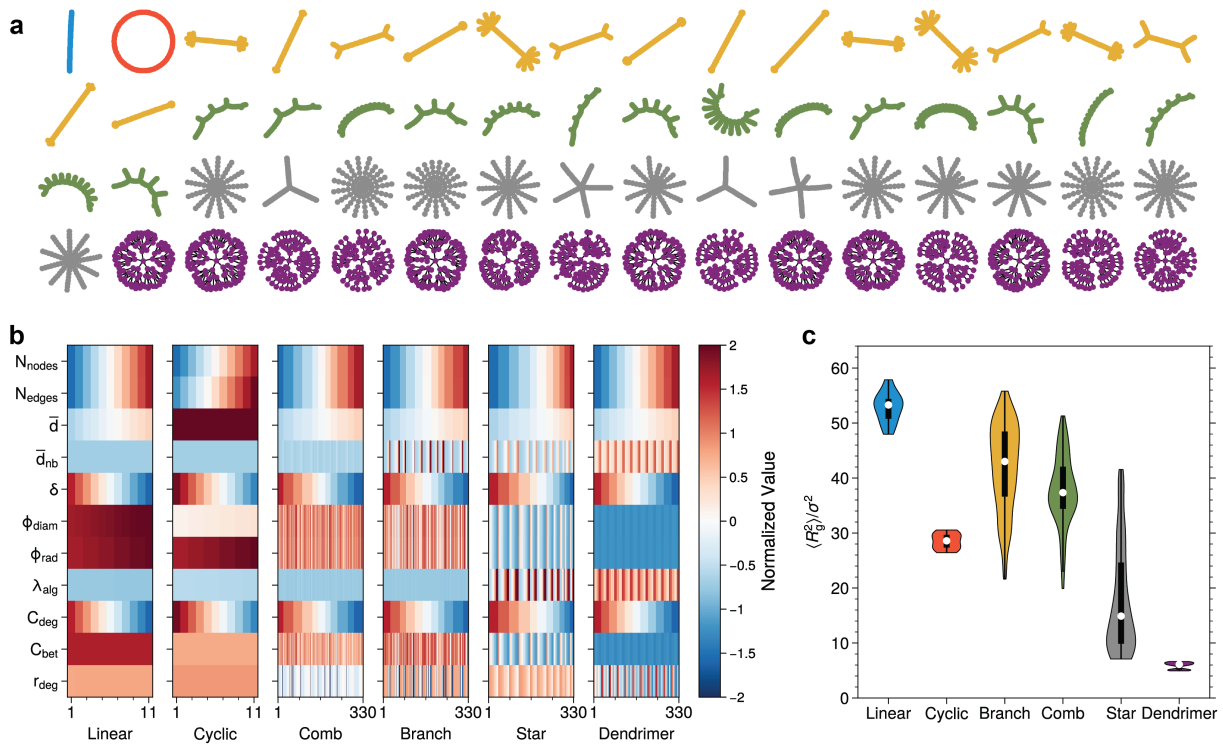

Figure 2: **Characteristics of generated polymers. (a)** Representative graphs of polymers from each architectural class. The number of polymers is proportional to occurrence in the dataset. **(b)** Comparison of topological descriptors across architectural classes. Values are standard-normalized for the dataset for each topological descriptor. Within a class, data for polymers are organized from left-to-right in ascending order of descriptor values, starting with the top (i.e., "Number of nodes") and proceeding downward to successively break ties. **(c)** The distribution of simulated $\langle R_g^2 \rangle$ for each architectural class. The white dot represents the median, the black bar spans the inter-quartile range (i.e., 25% to 75% percentiles), and the width indicates the distribution density. The color of the graphs in **(b)** align with those of the violins positioned over the respective classes in **(c)**.

2.2 Polymer Reconstruction and Property Prediction

Based on prior work on linear polymer featurization [59, 60], we hypothesized that polymer reconstruction with a VAE could be enhanced if derived topological descriptors were supplied as inputs. To examine this, we evaluate three distinct encoding strategies: `TopoGNN`, which integrates topological descriptors with graph features; `GNN`, which exclusively relies on graph features; and `Topo`, which solely employs topological descriptors. For each strategy, we consider a multitude of models with distinct hyperparameters and their performance across a broad range of evaluation metrics. For example, reconstruction performance is quantitatively evaluated with balanced accuracy (BACC), which measures the accuracy of individual entries in the reconstructed adjacency matrix. For topology classification, F_1 score is chosen to address the class imbalance in our dataset. Other metrics include the coefficient of determination R^2 for regression on $\langle R_g^2 \rangle$ and the Kullback-Leibler (KL) divergence. Representative models for each encoding strategy are selected using a comprehensive evaluation score (CES) that simultaneously considers all criteria:

$$\text{CES} \equiv \sqrt{(\overline{1-\text{BACC}})^2 + (\overline{\text{KL}})^2 + (\overline{1-R^2})^2 + (\overline{1-F_1})^2} \tag{1}$$

where $\bar{a}$ denotes the min-max normalized value of a; CES can be interpreted as the distance from the origin (a perfect model) in a vector space spanned by error metrics.

Table 1 summarizes the performance of these representative models. Across encoding strategies, `TopoGNN` emerges as the most overall effective, registering the smallest CES. By comparison, the `Topo` model yields slightly superior performance on regression and comparable F_1 score. Conversely, the `GNN` model demonstrates a slightly higher balanced accuracy in reconstruction tasks and a lower KL divergence; however, it significantly underperforms in regression and classification. These results support the inclusion of topological descriptors during construction of the VAE.

Table 1: Performance of representative models for each encoding strategy on validation set.

Models	Balanced Accuracy	Regression R^2	Classification F_1	KL Divergence	Distance to Origin
`TopoGNN`	0.9439	0.9915	**0.9953**	18.7244	**0.3829**
`GNN`	**0.9448**	0.9634	0.9768	**15.6018**	0.8348
`Topo`	0.9281	**0.9949**	**0.9953**	16.0418	0.3992

To assess model generalizability, we examine the performance of the representative models on the held-out test set. Figure 3 again indicates that `TopoGNN` delivers consistently strong performance across several evaluation criteria, while `GNN` and `Topo` can be deficient in particular metrics. Balanced

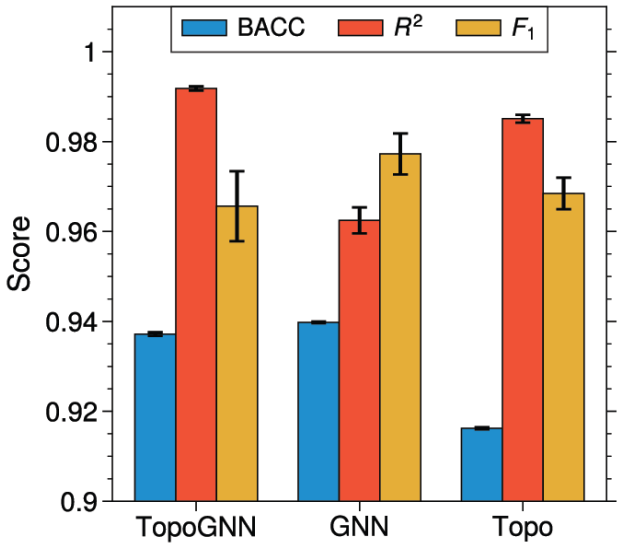

Figure 3: **Performance of variational autoencoder models.** Comparison of TopoGNN, GNN, and Topo in terms of polymer graph reconstruction, $\langle R_g^2 \rangle$ regression, and topology classification. BACC represents balanced accuracy, R^2 is the coefficient of determination, and F_1 measures accuracy based on the harmonic mean of precision and recall. The error bars represent the standard deviation arising from 10 random samplings of the latent space.

accuracy is highest for GNN (0.9397), closely followed by TopoGNN (0.9369) and then Topo (0.9164). This suggests that topological descriptors do not necessarily enhance reconstruction performance, although the ability of Topo to effectively reconstruct certain topologies (e.g., branch polymers) highlights the extensive information content encompassed by the 11 topological descriptors. By contrast, directly supplying topological information is clearly advantageous for predicting the characteristic polymer size. Here, TopoGNN stands out as the most effective, achieving the highest mean value (0.9920), surpassing Topo (0.9854) and GNN (0.9639). Meanwhile, GNN achieves the highest mean F_1 score (0.9783), followed by TopoGNN (0.9689) and Topo (0.9678); however all models display statistically comparable results regarding this classification metric. Taken together, this suggest workflows with VAEs can effectively address complexities induced by these polymer architectures.

For a more nuanced assessment of model quality, Figure 4 breaks down TopoGNN performance across architectures; comparable information for other models is in Figures S2 and S3. In polymer reconstruction, TopoGNN excels but faces challenges with specific cyclic and comb polymers (Figure 4a, gray dashed boxes). Notably, GNN generates errors, especially for star polymers, while Topo exhibits minor errors across most architectures. Regarding the prediction of $\langle R_g^2 \rangle$ (Figure 4b), TopoGNN performs well regardless of polymer class. Both GNN and Topo display high correlation, but errors are generally larger for GNN (Figure S2), indicating the difficulty in establishing a direct relationship be-

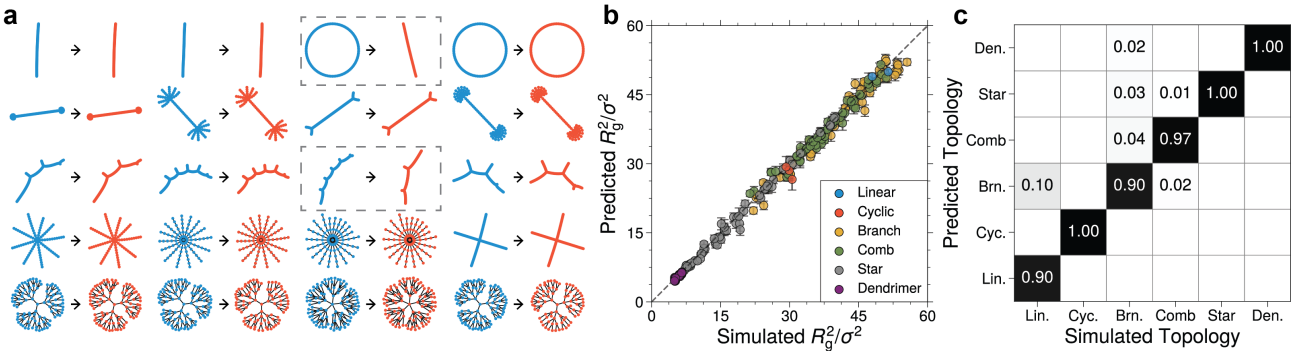

Figure 4: **Performance decomposition of `TopoGNN`. (a)** Polymer graph reconstructions by `TopoGNN`, contrasting true (blue) and predicted (red) polymer topologies. **(b)** Regression parity plot. The diagonal line signifies ideal regression accuracy, and error bars show standard deviation from random latent space sampling. **(c)** Confusion matrix representing the classification performance across various topologies: linear (lin), cyclic (cyc), branch (brn), comb, star, and dendrimer (den). Diagonal entries correspond to accurate classifications, while off-diagonal entries indicate misclassifications.

tween graph features and $\langle R_g^2 \rangle$. A saliency analysis (Figure S4) reveals that graph diameter, betweenness centrality, and algebraic connectivity most strongly influence $\langle R_g^2 \rangle$, aligning with their direct correlation with $\langle R_g^2 \rangle$ (Figure S1). For topology classification, `TopoGNN` (Figure 4c) is broadly effective, with most misclassifications occurring in linear, branched, and comb architectures. These issues are more pronounced in `Topo` and `GNN` (Figure S3) and can be augmented with other misclassifcia-tions. Overall, `TopoGNN`, which utilizes both graph and topological features, not only consistently outperforms other models but also delivers high-quality results. The remainder of the article therefore focuses on analysis and applications of `TopoGNN` to illustrate its practical deployment.

2.3 Latent Space Exploration and Polymer Generation

Figure 5 presents the UMAP projection of the 8-dimensional latent space of `topoGNN` into a 2-dimensional space for visualization. Distinct topological clusters emerge in Figure 5a and b, which reveals organization of the latent space that depends relationships amongst architectures and their physical properties. Dendrimers, characterized by their high orders of branches, form three, mostly isolated and distinct clusters that reflect how the dendrimer architectures were algorithmically generated; they are most closely related to star polymers and branched polymers (particularly those with pom-pom architectures). Branch, comb, and star polymers all notably overlap within the latent space, which is attributed to topological similarities (Figure 2b). Cyclic and linear polymers are interspersed within comb and branch clusters, with linear polymers sharing a long backbone and cyclic polymers possessing a long ring-closed backbone. This organization is clearly informed training with auxiliary

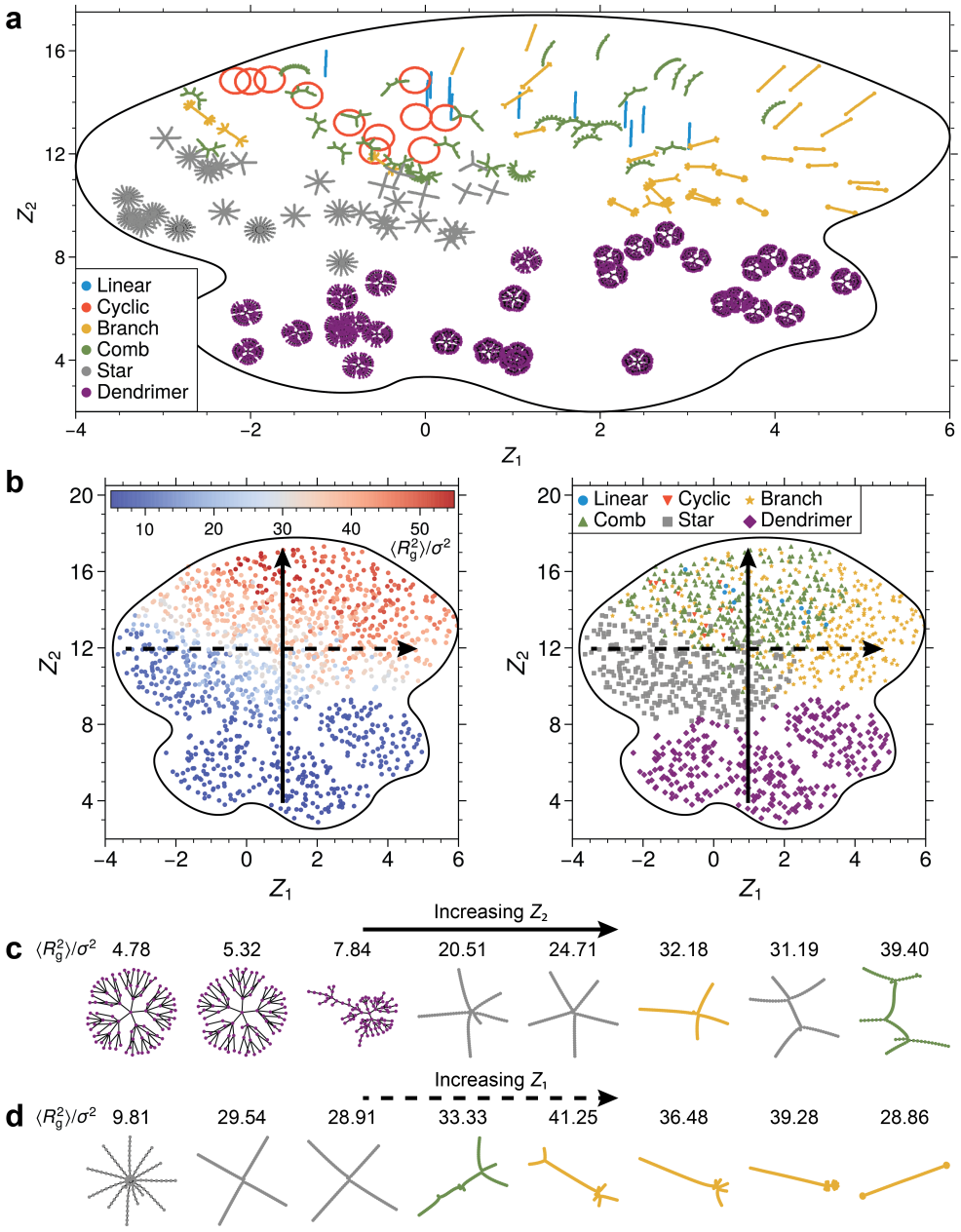

Figure 5: **Visualization and exploration of the latent space. (a)** Two-dimensional visualization of the `TopoGNN` latent space using the Uniform Manifold Approximation and Projection (UMAP) technique. A subset of the data is displayed for clarity, with each marker representing a polymer graph based on its latent vector. Different colors denote distinct topologies. **(b)** Organization of (left) $\langle R_g^2 \rangle$ and (right) topology in the UMAP-coordinate space. The dots signify the latent vectors of polymer graphs. The two arrows mark regions in the latent space targeted for exploration (i.e., new polymer topology generation). **(c)** As exploration progresses with an increase in Z_2 in the latent space (represented by a solid line), there is a near-monotonic rise in $\langle R_g^2 \rangle$ for the generated polymers. Progression with an increase in Z_1 (indicated by a dashed line) showcases shifts in polymer topology, moving through clusters characteristic of star, comb, and branch topologies.

tasks for predicting $\langle R_g^2 \rangle$ and classifying topologies, as illustrated in Figure 5b. A vertical trajectory in the UMAP space (marked by an increase in Z_2) results in an almost monotonic increase in $\langle R_g^2 \rangle$ for the generated polymer topologies (Figure 5c). Conversely, a horizontal trajectory (associated with an increase in Z_1) moreso transitions topology classes with slight variations in $\langle R_g^2 \rangle$ ((Figure 5d). Omitting the auxiliary tasks leads to less distinct separation of topological classes and disrupts the monotonicity of the $\langle R_g^2 \rangle$ (Figure S6) The latent spaces of `GNN` and `Topo` (Figure S5) are prone to similar issues. Overall, this highlights the effectiveness of the workflow for `TopoGNN` to produce an intuitive and physically meaningful latent space.

The latent space of `TopoGNN` can be used to generate a diverse set of polymer topologies. This is exemplified by computing the Vendi Score (VS) for each architecture (see Section 4.4.8 for details) and comparing it to that of the originally constructed dataset. Whereas the VS for the original dataset (1,342 points) is 2.0968, that for 1,342 topologies generated using `TopoGNN` is 5.0684, which exceeds those for `GNN` (4.9580) and `Topo` (4.3305). This indicates that all models can generate a more diverse range of polymer topologies compared to the original handcrafted dataset, which could have implications for downstream tasks, as explored in the next section.

2.4 Property-Guided Polymer Topology Generation

To illustrate one application for `TopoGNN`, we generate a series of distinct polymer topologies that exhibit specific $\langle R_g^2 \rangle$. While $\langle R_g^2 \rangle$ itself is a fundamental characteristic of the polymers, the rationale here is moreso to demonstrate the production of new, alternative materials with similar characteristics and further to assess how topology affects other polymer properties, such as rheology, without conflation of other factors. We therefore select target $\langle R_g^2 \rangle$ ranges of 7.5 ± 2, 30 ± 2, and 50 ± 2 which represent the low, intermediate, and high regions of $\langle R_g^2 \rangle$ in the dataset, respectively (Figure 2) and conditionally sample polymers from the latent space across the different topological classes. The $\langle R_g^2 \rangle$ are then validated for the generated polymer topologies using MD simulation. These results are shown in Figure 6, which illustrates that `TopoGNN` can indeed produce a range of distinct structures that exhibit effectively similar $\langle R_g^2 \rangle$. Targeting $\rangle = 7.5 \pm 2$ predominantly yields dendrimer and star topologies, targeting $\langle R_g^2 \rangle = 30 \pm 2$ yields branch, comb, cyclic, and star topologies, and targeting $\langle R_g^2 \rangle = 50 \pm 2$ mostly yields in branch and comb architectures. With the current approach, however, architectures that satisfy specific targets cannot be arbitrarily produced based on the molecular-weight restrictions. For example, dendrimers are more or less restricted to low $\langle R_g^2 \rangle$, while linear polymers are mostly restricted to larger $\langle R_g^2 \rangle$. Moreover, relatively few polymers meet the ambitious target of 50 ± 2, which

is consistent with the paucity of data points around $\langle R_g^2 \rangle = 50 \pm 2$ within the original dataset; however, the group of polymers here uniformly exceed those of the smaller 30 ± 2 target. Interestingly, TopoGNN also proffers architectures, such as irregular dendrimers and nuanced branching patters in stars and combs, that go beyond those of the original dataset. Overall, these results reflect the intended capability of TopoGNN to generate a broad spectrum of original polymer topologies that align with a target property.

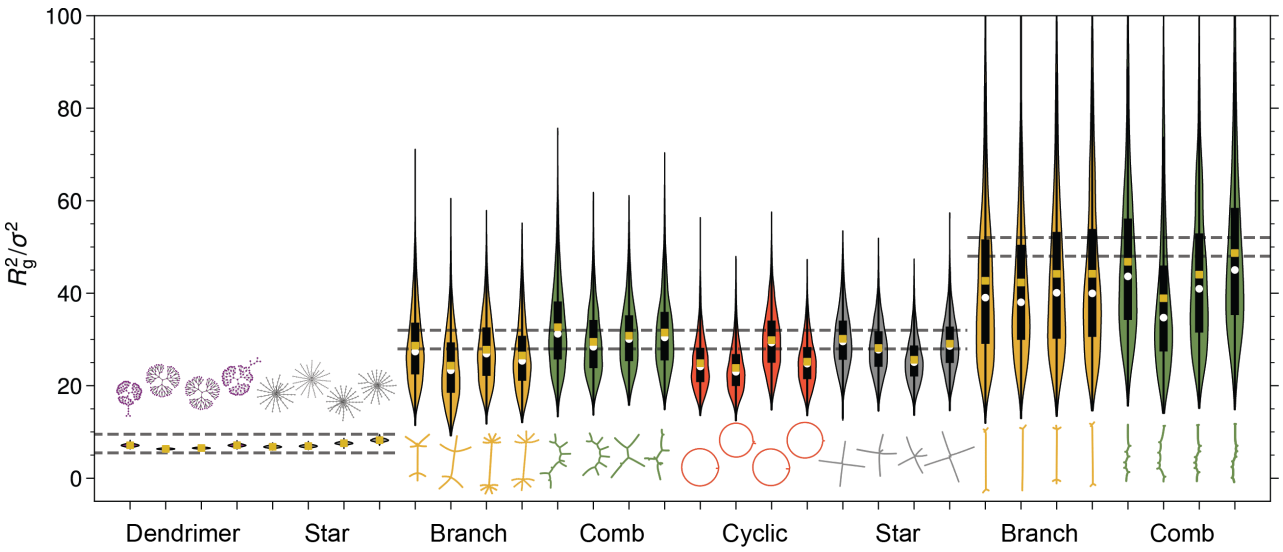

Figure 6: **Generation of polymer topologies with target $\langle R_g^2 \rangle$.**Topologies are generated aiming for target $\langle R_g^2 \rangle$ values of 7.5 ± 2, 30 ± 2, and 50 ± 2. Each generated topology is accompanied by its type and the predicted $\langle R_g^2 \rangle$ from TopoGNN, presented in parentheses on the x-axis. A violin plot showcases the revalidation of $\langle R_g^2 \rangle$ via MD simulation for every topology. The gold dot marks the $\langle R_g^2 \rangle$, while the white dot stands for the median. The black bar represents the interquartile range, and the plot width reflects the distribution density of R_g^2. Two dashed lines highlight the $\langle R_g^2 \rangle$ range used in the guided search.

2.5 Rheological Analysis

Using TopoGNN, we explore the influence of polymer topology on rheological characteristics. While solution viscosity at dilute concentrations is primarily determined by polymer size, which sets the overlap concentration,[61] we control for this factor by designing topologies with specified $\langle R_g^2 \rangle$ and examine topological implications across a range of concentrations. Figure 7a examines the concentration-dependent shear viscosity as determined from MD simulations of four selected topologies. Figure 7a presents concentration-dependent shear viscosity from MD simulations of four selected topologies. Differences emerge beyond 0.4 σ^{-3}, with cyclic polymers showing lower viscosities due to reduced entanglements, and branched polymers exhibiting elevated viscosities due to

extended side chains. Star and comb polymers demonstrate similar, somewhat lower shear viscosities compared to branched polymers, highlighting the impact of side-chain position and density on entanglement effectiveness. Additionally, we observe nuanced differences in frequency-dependent storage and loss moduli, G' and G'', across topologies and concentrations (Figures 7b and 7c). While all solutions exhibit liquid-like viscous behavior at low frequencies and solid-like behavior at high frequencies below 0.6 σ^{-3}, star, branch, and comb polymers display three crossover frequencies as concentration increases. In contrast, cyclic polymers maintain a single crossover frequency, indicating less nuanced viscoelastic behavior. This highlights potential for how rheological properties might be modulated through strategic architecture design.

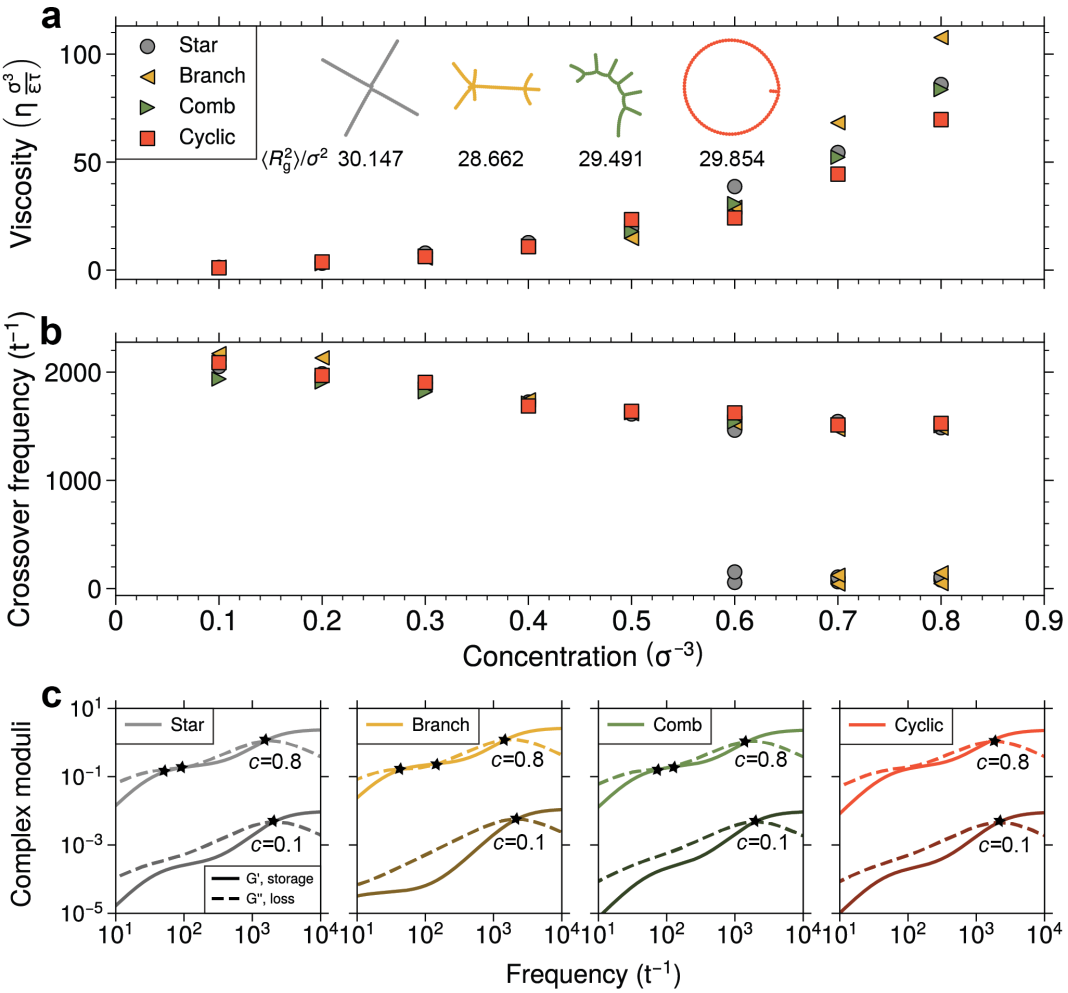

Figure 7: **Effect of polymer topology on shear viscosity and complex moduli at comparable $\langle R_g^2 \rangle$. (a)** Influence of polymer topology and concentration on viscosity, featuring topologies such as star, branch, comb, and cyclic, each with a $\langle R_g^2 \rangle$ of approximately 30 ± 2. **(b)** Relationship between polymer topology, concentration, and complex moduli crossover frequencies. **(c)** Complex moduli for various topologies at concentrations of 0.1 and 0.8, with the star symbol marking the crossover point.

3 Discussion

This study employed variational autoencoders to address energent combinatorial complexity of diverse polymer topologies, which has been scarcely addressed in machine learning of macromolecules. We constructed an extensive dataset featuring the average squared radius of gyration ($\langle R_g^2 \rangle$) for 1,342 polymers with various architectures, including linear, cyclic, branch, comb, star, and dendrimer structures. By analyzing different encoding strategies and input representations, we found that meaningful latent spaces of polymers with complex topologies can be established by *(i)* incorporating both graph-explicit and graph-derived features and *(ii)* coupling graph reconstruction tasks with auxiliary prediction tasks, such as those related to physical properties. Probabilistic sampling over the latent space was shown to result in rich topological diversity. These generative capabilities were then used to produce unique polymer topologies with target characteristic sizes in dilute solution. This enabled subsequent investigation by coarse-grained molecular dynamics into how topology influences rheological properties, such as shear viscosity and viscoelastic moduli, while controlling for polymer size. While all architectures exhibited similar rheological behavior at relatively low concentrations, distinct responses emerged at higher concentrations. For instance, localized branches at chain ends resulted in more viscous solutions compared to other architectures, including cyclic structures that exhibited minimal entanglements. Apart from illustrating how rheological behavior might be tuned or altered via polymer architecture, this also showcases a new paradigm for studying the physical properties of topologically distinct systems.

The methodologies introduced in this study pave the way for several future research directions. Particularly, `TopoGNN` exhibits promising potential as a generative model, offering a cost-effective alternative to experiments or simulations in predicting properties like $\langle R_g^2 \rangle$. While $\langle R_g^2 \rangle$ serves as a straightforward and computationally accessible quantity, there is interest in extending the strategy to incorporate or utilize other properties. Although this work leveraged `TopoGNN` to simply compare rheological properties in systematic fashion, in the future, it may be deployed to guide design efforts aimed at optimizing polymer properties. We also note that the dataset and machine learning framework are currently limited to polymers with a narrow range of bead numbers (equivalently, molecular weights). Future research will explore the extensibility and transferability of machine learning architectures across various molecular weights, potentially through the use of string-based representations.[62–64] Furthermore, while this study specifically investigates the impact of topology, it does not address specific chemistry. However, to understand and control the properties of polymers with chemical, compositional, and topological complexity is an outstanding challenge in

polymer science. This work lays the foundation for innovative approaches towards those ends.

4 Methods

4.1 Description of Dataset

The dataset comprises 1,342 polymer architectures, each containing between 90 and 100 constitutional units, or beads. Polymer architectures encompass a wide range of topologies, including linear, cyclic, branch, comb, star, and dendrimer structures. Due to limitations bead count, linear and cyclic topologies are restricted to 11 distinct polymers each, whereas other topologies are represented by 330 unique polymers each. The polymers are chemically homogeneous with all beads treated equivalently. The procedure for generating polymer graphs is described in the SI Section S2. For each polymer graph, we calculate an 11-dimensional topological descriptor vector[43, 65] using the number of nodes, number of edges, average degree, average neighbor degree, density, diameter, radius, algebraic connectivity, degree centrality, betweenness centrality, and degree assortativity as elements. For further details on these descriptors, readers are referred to SI Section S1.

4.2 Calculation of Polymer Properties

4.2.1 Radius of Gyration

We investigate the structural properties of individual polymer chains using coarse-grained molecular dynamics. To do so, we compute the gyration tensor $\mathbf{S}$:

$$\mathbf{S} = \frac{1}{N}\sum_{i=1}^{N}\left(\mathbf{r}_i - \mathbf{r}_{\mathrm{cm}}\right)\left(\mathbf{r}_i - \mathbf{r}_{\mathrm{cm}}\right)^T \tag{2}$$

where $\mathbf{r}_i$ denotes the position vector of the i-th bead, $\mathbf{r}_{\mathrm{cm}}$ represents the center-of-mass position of the polymer, and T indicates the transpose operation. Diagonalizing yields $\mathbf{S} = \mathrm{diag}(\lambda_1^2, \lambda_2^2, \lambda_3^2)$ where the diagonal elements are the principal moments of the gyration tensor ordered as $\lambda_1 \leq \lambda_2 \leq \lambda_3$. The squared radius of gyration can be subsequently computed as

$$R_{\mathrm{g}}^2 = \lambda_1^2 + \lambda_2^2 + \lambda_3^2 \tag{3}$$

and quantifies the size of a given polymer conformation. The ensemble average $\langle R_{\mathrm{g}}^2 \rangle$ is the constructed using a series all sampled configurations. This ensemble-averaged quantity serves as the

target for the regression auxiliary task.

4.2.2 Rheological Properties

We also characterize several rheology-related properties for select polymer systems. The shear viscosity η of the polymer solution is formally calculated via

$$\eta = \int_0^\infty G(t)dt \tag{4}$$

where $G(t)$ denotes the stress relaxation modulus. We determine $G(t)$ using the Green-Kubo relation

$$G(t) = \frac{1}{3} \sum_{\alpha\beta=xy,xz,yz} \frac{V}{k_B T} \langle \overline{\sigma_{\alpha\beta}(t)\sigma_{\alpha\beta}(0)} \rangle, \tag{5}$$

with V representing the simulation box volume, $\overline{\sigma_{\alpha\beta}}(t)$ signifying the off-diagonal stress tensor components averaged at intervals of 1000 steps, and $\langle \cdots \rangle$ denoting an ensemble-average. Often, $G(t)$ exhibits significant noise at long times, which renders direct numerical integration of Eqn. 4 unreliable. Therefore, following prior work,[66] we fit the simulated $G(t)$ data to a generalized Maxwell model, given by $G(t) = \sum_p G_p \exp(-t/\tau_p)$, where G_p and τ_p represent the modulus and relaxation time of the p-th element, respectively. This approach yields the viscosity $\eta = \sum_p G_p \tau_p$. We also compute the storage modulus (G') and the loss modulus (G'') to better characterize the viscoelastic properties of the polymers. These moduli are obtained from the Fourier transform of the stress relaxation modulus, yielding

$$G^*(\omega) = i\omega \int_0^\infty G(t) e^{-i\omega t} dt \tag{6}$$

$$= G'(\omega) + iG''(\omega). \tag{7}$$

Here, $G'(\omega)$, the storage modulus, reflects the elastic, or energy-storing, aspect of the material, while $G''(\omega)$, the loss modulus, represents the viscous, or energy-dissipating, component. This analysis is thus restricted to linear viscoelasticity.

4.3 MD Simulation Details

MD simulations are used to generate polymer configurations for the characterization of polymer properties. All simulations are conducted using the LAMMPS simulation package [67] in reduced units; the units of mass, distance, and energy are denoted by m, σ, and ε, respectively. The reduced

time unit follows as $(m\sigma^2/\varepsilon)^{1/2}$. All simulations are considered to take place in an implicit-solvent environment, with dynamics of the polymer(s) governed by the Langevin equation. The equations-of-motion are numerically integrated using the velocity-Verlet integration scheme with a 0.001 timestep. The solvent friction coefficient is set to $\varsigma = 0.1$.

Polymer interactions are modeled via a combination of bonded and nonbonded potential energy contributions. The total potential energy U of a system with configuration $\mathbf{r}^N$ is expressed as:

$$U(\mathbf{r}^N) = \sum_{\text{bonds}} U_{\text{vib}}(r_{ij}) + \sum_{i<j} U_{\text{nb}}(r_{ij}), \tag{8}$$

where r_{ij} represents the internal distance calculated from the coordinates $\mathbf{r}^N$. The nonbonded energy contributions for all pairs of beads are computed using the following equation:

$$U_{\text{nb}}(r_{ij}) = \begin{cases} 4\varepsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right] + \epsilon_{ij}, & \text{if } i,j \text{ are bonded and } r_{ij} < 2^{1/6} \\ 0, & \text{otherwise,} \end{cases} \tag{9}$$

where ε_{ij} and σ_{ij} are set to 1. For directly bonded beads, the stretching energy is calculated as:

$$U_{\text{vib}}(r_{ij}) = -\frac{1}{2}K_{ij}(R_{ij}^{(0)})^2 \ln\left[1 - \left(\frac{r_{ij}}{R_{ij}^{(0)}}\right)^2\right], \tag{10}$$

where K_{ij} is assigned a value of 30, and $R_{ij}^{(0)}$ is fixed at 1.5.

4.3.1 Single-chain Simulations

Simulations of single coarse-grained polymer chains (no boundary conditions) are used to characterize $\langle R_{\text{g}}^2 \rangle$. Each simulation is conducted for 2×10^7 steps, allocating the first half for system equilibration. Configurations for analysis are sampled every 2×10^3 timesteps during the latter half of the simulation.

4.3.2 Many-chain Simulations

Simulations of many chains within a simulation cell with cubic periodic boundary conditions are used for rheological analysis of a subset of polymers with comparable ensemble-averaged square radii of gyration, $\langle R_{\text{g}}^2 \rangle$. Simulations are performed across various concentrations (0.1 to 0.8) to cover both semi-dilute and semi-concentrated regimes. Each simulation uses 100 chains with the simula-

tion cell dimensions adjusted to match the desired concentration. Equilibration periods of 10^7 steps are utilized for all simulation concentrations. Upon achieving equilibrium, data are collected for 10^7 steps at a timestep of 0.001.

4.4 Machine Learning Details

4.4.1 Data Preprocessing

Polymers are represented using graph notation $\mathcal{G} = (V, E)$, where V is the set of nodes, and E is the set of edges. To address the variability in node counts across different polymers, ranging from 90 to 100, we introduce "ghost" nodes with zero-edge connections to standardize graph sizes to 100 nodes using node padding [68, 69]. Because all polymer beads are equivalent, the adjacency vector $a_i \in \mathbb{R}^{100}$ serves as the sole node feature for each polymer bead. Elements of this vector are defined such that $a_i = 1$ if node i is connected to the current node, and $\mathbf{a}_i = 0$ otherwise. All bonds are also equivalent, and so edge features are not included in the representation. Polymers are also characterized by an 11-dimensional topological descriptor vector $\mathbf{t} \in \mathbb{R}^{11}$ as previously described. For the task of polymer reconstruction, an adjacency matrix $\mathbf{A} \in \mathbb{R}^{100\times100}$ is associated with each polymer, where $A_{ij} = 1$ indicates an edge between nodes i and j, and $A_{ij} = 0$ indicates no edge. For the auxiliary regression task, each polymer is associated with a label for $\langle R_g^2 \rangle$, denoted $y_r \in \mathbb{R}$. For the auxiliary classification task, each polymer is associated with a one-hot encoded topology label, denoted $\mathbf{y}_t \in \mathbb{R}^6$. The dataset of 1,342 polymers is divided into three subsets: 858 for training (64%), 215 for validation (16%), and 269 for testing (20%). Stratified splitting is used to ensure each subset represents all polymer topologies. The training set is utilized to train the VAE, the validation set for hyperparameter optimization, and the test set to evaluate the model generalizability.

4.4.2 Model Architectures

Overall, we explore three unique encoder architectures while maintaining a uniform decoder architecture. The first model, designated as `TopoGNN`, combines a graph encoder with a topological descriptor encoder, thus operating as a multi-input model. The second model, `GNN`, exclusively employs the graph encoder. The third model, `Topo`, relies solely on the topological descriptor encoder. The architecture of the VAE for `TopGNN` is depicted in Figure 8. The encoder transforms input data into a latent space representation. Graph inputs are represented using an adjacency matrix $\mathbf{A} \in \mathbb{R}^{100\times100}$ and a node feature matrix $\mathbf{X} \in \mathbb{R}^{100\times100}$, with the adjacency vector serving as the node feature due to identical nodes. The Graph Isomorphism Network encoder [70], equipped with two graph convo-

lutional layers, maps these inputs into a 32-dimensional feature vector $\mathbf{h}_\mathrm{g}$. The topological descriptor vector is similarly converted into a 32-dimensional feature vector $\mathbf{h}_\mathrm{t}$ by a dense neural network (DNN) encoder. Subsequently, the feature vectors $\mathbf{h}_\mathrm{g}$ and $\mathbf{h}_\mathrm{t}$ are concatenated to yield a combined feature vector $\mathbf{h} \in \mathbb{R}^{64}$. Additional dense layers generate the parameters of the latent Gaussian distribution: the mean $\boldsymbol{\mu}$ and the logarithm of variance $\log \boldsymbol{\sigma}^2$. These parameters define the latent space embedding $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma})$, which has a dimensionality of 8. The decoder then samples from $\mathbf{z}$ to reconstruct data. A convolutional neural network is used to reconstruct the adjacency matrix $\hat{\mathbf{A}}$. Additionally, two additional and distinct neural networks are tasked with predicting $\hat{y}_\mathrm{r}$ and $\hat{\mathbf{y}}_\mathrm{t}$.

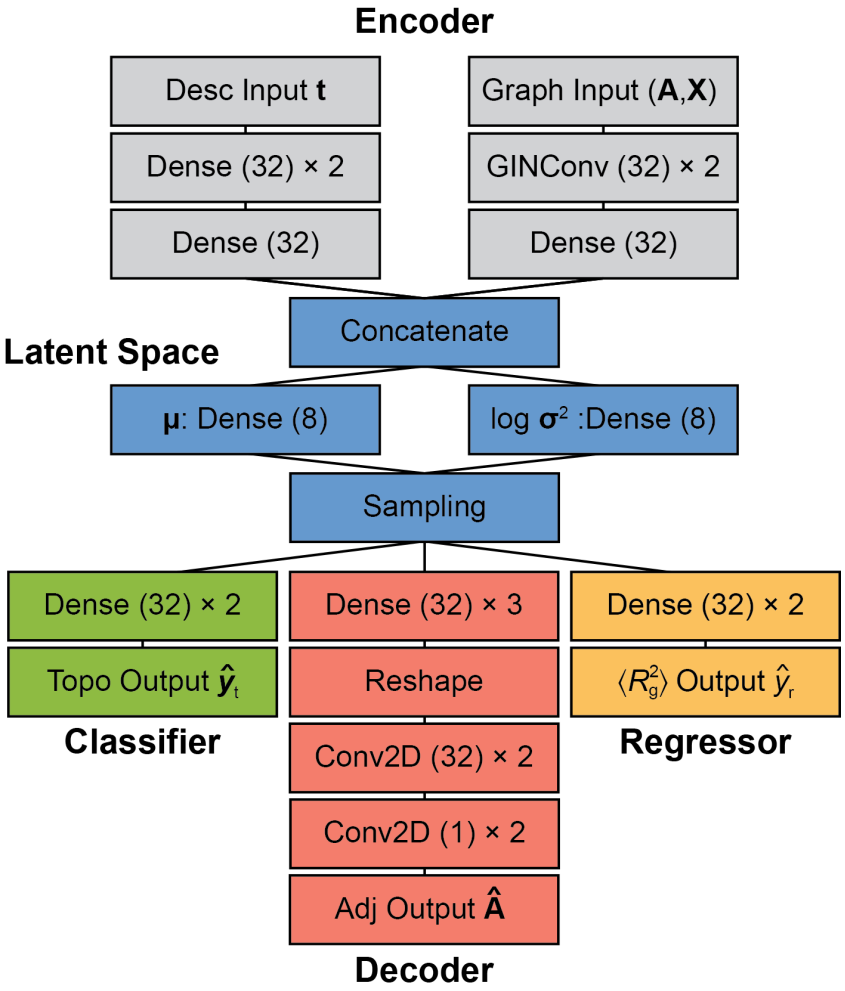

Figure 8: **Architecture of the variational autoencoder (VAE) for `TopGNN`.** The model compresses information from the graph and topological descriptors. These two sets of compressed features are then concatenated and passed to the latent space, where the model learns a normal distribution characterized by parameters μ and σ. Subsequently, samples drawn from this distribution are used by the decoder to reconstruct the adjacency matrix of the input graph. Additionally, the same samples are used in two auxiliary tasks: predicting the radius of gyration and classifying the topology. The numbers in the parentheses indicates the size of the layer.

4.4.3 Loss Functions

Training of the VAE uses a composite loss function $\mathcal{L}_{\text{VAE}}$

$$\mathcal{L}_{\text{VAE}} = \mathcal{L}_{\text{Rec}} + \mathcal{L}_{\text{KL}} + \lambda_{\text{Reg}}\mathcal{L}_{\text{Reg}} + \lambda_{\text{Cls}}\mathcal{L}_{\text{Cls}}, \tag{11}$$

which features terms associated with reconstruction, $\mathcal{L}_{\text{Rec}}$ via binary cross-entropy (BCE); Kullback-Leibler (KL) divergence, $\mathcal{L}_{\text{KL}}$; regression for y_r $\mathcal{L}_{\text{Reg}}$; and classification for y_t via cross-entropy (CE), $\mathcal{L}_{\text{Cls}}$. In Eq. (11), λ_{Reg} and λ_{Cls} are hyperparameter weights that are adjustable for optimizing performance. The individual loss terms are defined as follows:

$$\mathcal{L}_{\text{Rec}} = \text{BCE}(\mathbf{A}, \hat{\mathbf{A}}) \tag{12}$$

$$= -\sum_{i=1}^{100}\sum_{j=1}^{100} A_{ij}\log(\hat{A}_{ij}) + (1 - A_{ij})\log(1 - \hat{A}_{ij}), \tag{13}$$

$$\mathcal{L}_{\text{KL}} = D_{\text{KL}}(\mathbf{z}\,||\,\mathcal{N}(0, \mathbf{I})) \tag{14}$$

$$= -\frac{1}{2}\sum_{i=1}^{8}(1 + \log(\sigma_i^2) - \sigma_i^2 - \mu_i^2), \tag{15}$$

$$\mathcal{L}_{\text{Reg}} = \text{MAE}(y_{\text{r}}, \hat{y}_{\text{r}}) \tag{16}$$

$$= |y_{\text{r}} - \hat{y}_{\text{r}}|, \tag{17}$$

$$\mathcal{L}_{\text{Cls}} = \text{CE}(\mathbf{y}_{\text{t}}, \hat{\mathbf{y}}_{\text{t}}) \tag{18}$$

$$= -\sum_{i=1}^{6} y_{\text{t},i}\log(\hat{y}_{\text{t},i}). \tag{19}$$

4.4.4 Model Training and Hyperparameter Tuning

All models are implmented using TensorFlow [71]. Models undergo training for 1000 epochs with the Adam optimizer [72]. A broad range of hyperparameters is explored, encompassing batch sizes $\{32, 64, 128\}$, learning rates $\{0.0001, 0.001, 0.01\}$, and regularization terms $\lambda_{\text{Reg}} \in \{0.01, 0.1, 1, 10, 100\}$ and $\lambda_{\text{Cls}} \in \{0.01, 0.1, 1, 10, 100\}$. Criteria for model weight saving include overall validation loss, Evidence Lower Bound (ELBO), and reconstruction balanced accuracy. Across three encoder types, this approach results in 2,025 unique hyperparameter combinations. For each encoder type, the optimal hyperparameter configuration is selected based on a composite validation metric that combines several key performance indicators: reconstruction balanced accuracy (BACC), KL divergence, $\langle R_{\text{g}}^2 \rangle$ regression R^2 value, and the topology classification F_1 score.

These metrics are min-max normalized

$$\bar{\mathbf{a}} = \frac{\mathbf{a} - \min(\mathbf{a})}{\max(\mathbf{a}) - \min(\mathbf{a})} \tag{20}$$

and consolidated into a four-dimensional vector as

$$\mathbf{v} = \left[\overline{1 - \mathrm{BACC}}, \overline{\mathrm{KL}}, \overline{1 - R^2}, \overline{1 - F_1}\right]. \tag{21}$$

Subsequently, the optimal hyperparameter configuration is determined as that nearest to the origin $(0, 0, 0, 0)$. Since hyperparameter optimization does not involve updating model weights, compared to abstract loss functions, these metrics are more interpretable and directly related to our objectives, such as improving reconstruction, prediction accuracy, and model generalization.

4.4.5 Random Polymer Generation

To generate random polymer topologies, points are sampled from a predefined latent distribution, and the resultant latent vector, $\mathbf{z}_{\mathrm{gen}}$, is transformed into an adjacency matrix, $\mathbf{A}_{\mathrm{gen}}$. Each element in $\mathbf{A}_{\mathrm{gen}}$ indicates the connectivity between nodes. To avoid spurious and unphysical edge-formation or other errors during reconstruction, generated polymers then undergo a graph-cleansing step. This step principally removes isolated nodes and breaks small rings. Because this modifies the original adjacency matrix, we implement a validation protocol, which is fully described in SI Section S3. Briefly, the cleansed graph and its recalculated topological descriptors are re-encoded to derive new values for $\langle R_{\mathrm{g}}^2 \rangle$ and topology class. Cleansed graphs are considered valid if they satisfy three criteria. First, the difference in $\langle R_{\mathrm{g}}^2 \rangle$ values before and after cleansing is less than 2 σ^2. Second, the topology classification is unchanged. Third, the mean squared difference between the pre- and post-cleansing latent vectors is less than 1. These criteria preserve the inherent properties of the generated polymers.

4.4.6 Polymer Generation with Target Properties

To generate polymers with specific target properties, namely $\langle R_{\mathrm{g}}^2 \rangle$ and topology, "parent" polymers that exhibit these desired characteristics are first identified from the original dataset. The criterion for $\langle R_{\mathrm{g}}^2 \rangle$ is relaxed to allow a tolerance range of ± 2 around the target value. Points are then sampled near the latent-space vectors of the parent polymers by introducing Gaussian noise with a mean of 0 and a variance of 0.1. The $\langle R_{\mathrm{g}}^2 \rangle$ and topology of each generated candidate polymer is then predicted using the trained ML model. Candidates that do not exhibit target topology or deviate in $\langle R_{\mathrm{g}}^2 \rangle$ by

more than 2 σ^2 are discarded. Following this initial screening, polymer graphs undergo cleansing as previously described, except that $\langle R_g^2 \rangle$ of candidates must more stringently remain within 2 σ^2 of both the initial target and pre-cleansing values. Subsequently, non-unique graphs, either duplicated from the original dataset or already present within the generated pool, are identified and removed through graph isomorphism checks. Additional details are in the SI Section S4.

4.4.7 Latent-space Visualization

The latent space is visualized using the Uniform Manifold Approximation and Projection (UMAP) algorithm [73]. The parameters follow that of prior work [43], wherein the UMAP local neighborhood size is fixed at 200, the minimum embedding distance between points is set to 1, and the Euclidean distance metric is utilized in feature space analysis. This results in a mapping from $\mathbb{R}^8$ to $\mathbb{R}^2$: $\mathrm{UMAP}(\mathbf{z}) = \mathbf{u}$, where $\mathbf{z}$ denotes a latent vector and $\mathbf{u}$ its corresponding low-dimensional representation.

4.4.8 Diversity Evaluation

To calculate the diversity of a set of polymer topologies, each graph representation undergoes transformation into a Laplacian spectrum, encapsulating all eigenvalues of the graph Laplacian matrix. The Laplacian matrix is defined as the difference between the adjacency matrix and the degree matrix of the graph. Diversity quantification employs the Vendi Score (VS) [74], defined as:

$$\mathrm{VS}(\mathbf{K}) = \exp\left(-\sum_{i=1}^{n} \lambda_i \log \lambda_i\right), \tag{22}$$

where λ_i represents the eigenvalues of the matrix $\mathbf{K}/n$, with the convention $0 \log 0 = 0$. The similarity function in use is the dot product between normalized Laplacian spectra, denoted as $\mathbf{X} \in \mathbb{R}^{n \times 100}$, with 100 indicating the maximum eigenvalue count. For spectral vectors shorter than 100, zero-padding ensures length standardization. For reference, the minimum VS value is unity.

Data Availability

The data associated with this study are publicly accessible at DOI:10.5281/zenodo.10672434.

Code Availability

The code associated with this study is publicly accessible at https://github.com/webbtheosim/poly-topoGNN-vae.

Acknowledgements

M.A.W. and A.B.D acknowledge funding from the Princeton Catalysis Initiative for this research. M.A.W. and S.J. also acknowledge support from the donors of ACS Petroleum Research Fund under Doctoral New Investigator Grant 66706-DNI7.

References

[1] Bertoft, E. Understanding starch structure: Recent progress. *Agronomy* **7**, 56 (2017).

[2] Gao, Y. *et al.* Complex polymer architectures through free-radical polymerization of multivinyl monomers. *Nature Reviews Chemistry* **4**, 194–212 (2020). URL https://api.semanticscholar.org/CorpusID:214606428.

[3] Blosch, S. E., Scannelli, S. J., Alaboalirat, M. & Matson, J. B. Complex polymer architectures using ring-opening metathesis polymerization: Synthesis, applications, and practical considerations. *Macromolecules* **55**, 4200–4227 (2022).

[4] Matyjaszewski, K. Atom transfer radical polymerization (atrp): current status and future perspectives. *Macromolecules* **45**, 4015–4039 (2012).

[5] Chiefari, J. *et al.* Living free-radical polymerization by reversible addition-fragmentation chain transfer: the raft process. *Macromolecules* **31**, 5559 (1998).

[6] Bazan, G. C. & Schrock, R. R. Synthesis of star block copolymers by controlled ring-opening metathesis polymerization. *Macromolecules* **24**, 817–823 (1991).

[7] Levi, A. E. *et al.* Efficient synthesis of asymmetric miktoarm star polymers. *Macromolecules* **53**, 702–710 (2020).

[8] Yoo, J., Runge, M. B. & Bowden, N. B. Synthesis of complex architectures of comb block copolymers. *Polymer* **52**, 2499–2504 (2011).

[9] Bousquet, A., Barner-Kowollik, C. & Stenzel, M. H. Synthesis of comb polymers via grafting-onto macromolecules bearing pendant diene groups via the hetero-diels-alder-raft click concept. *Journal of Polymer Science Part A: Polymer Chemistry* **48**, 1773–1781 (2010).

[10] Bayer, U. & Stadler, R. Synthesis and properties of amphiphilic "dumbbell"-shaped grafted block copolymers, 1. anionic synthesis via a polyfunctional initiator. *Macromolecular Chemistry and Physics* **195**, 2709–2722 (1994).

[11] Knauss, D. M. & Huang, T. Star-block-linear-block-star triblock (pom-pom) polystyrene by convergent living anionic polymerization. *Macromolecules* **35**, 2055–2062 (2002).

[12] Liu, B., Kazlauciunas, A., Guthrie, J. T. & Perrier, S. One-pot hyperbranched polymer synthesis mediated by reversible addition fragmentation chain transfer (raft) polymerization. *Macromolecules* **38**, 2131–2136 (2005).

[13] Chen, S., Xu, Z. & Zhang, D. Synthesis and application of epoxy-ended hyperbranched polymers. *Chemical Engineering Journal* **343**, 283–302 (2018).

[14] Hawker, C. J. & Frechet, J. M. Preparation of polymers with controlled molecular architecture. a new convergent approach to dendritic macromolecules. *Journal of the American Chemical Society* **112**, 7638–7647 (1990).

[15] Lepoittevin, B., Matmour, R., Francis, R., Taton, D. & Gnanou, Y. Synthesis of dendrimer-like polystyrene by atom transfer radical polymerization and investigation of their viscosity behavior. *Macromolecules* **38**, 3120–3128 (2005).

[16] Lepoittevin, B. *et al.* Synthesis and characterization of ring-shaped polystyrenes. *Macromolecules* **33**, 8218–8224 (2000).

[17] Iatrou, H., Hadjichristidis, N., Meier, G., Frielinghaus, H. & Monkenbusch, M. Synthesis and characterization of model cyclic block copolymers of styrene and butadiene. comparison of the aggregation phenomena in selective solvents with linear diblock and triblock analogues. *Macromolecules* **35**, 5426–5437 (2002).

[18] Zhang, H., Gnanou, Y. & Hadjichristidis, N. Well-defined polyethylene molecular brushes by polyhomologation and ring opening metathesis polymerization. *Polymer Chemistry* **5**, 6431–6434 (2014).

[19] Zhang, H. & Hadjichristidis, N. Well-defined bilayered molecular cobrushes with internal polyethylene blocks and ω-hydroxyl-functionalized polyethylene homobrushes. *Macromolecules* **49**, 1590–1596 (2016).

[20] Wever, D. A. Z., Picchioni, F. & Broekhuis, A. A. Polymers for enhanced oil recovery: A paradigm for structure–property relationship in aqueous solution. *Progress in Polymer Science* **36**, 1558–1628 (2011). URL https://api.semanticscholar.org/CorpusID:94104090.

[21] Wever, D. A. Z., Polgar, L. M., Stuart, M. C. A., Picchioni, F. & Broekhuis, A. A. Polymer molecular architecture as a tool for controlling the rheological properties of aqueous polyacrylamide solutions for enhanced oil recovery. *Industrial & Engineering Chemistry Research* **52**, 16993–17005 (2013). URL https://api.semanticscholar.org/CorpusID:101162782.

[22] Fan, Z. W. *et al.* Topology and dynamic regulations of comb-like polymers as strong adhesives. *Macromolecules* **56**, 1514–1526 (2023).

[23] Xiong, C., Xiong, W., Mu, Y., Pei, D. & Wan, X. Mussel-inspired polymeric coatings with the antifouling efficacy controlled by topologies. *Journal of Materials Chemistry B* **10**, 9295–9304 (2022).

[24] Modica, K. J., Martin, T. B. & Jayaraman, A. Effect of polymer architecture on the structure and interactions of polymer grafted particles: Theory and simulations. *Macromolecules* **50**, 4854–4866 (2017). URL https://api.semanticscholar.org/CorpusID:103925906.

[25] Khabaz, F. & Khare, R. Effect of chain architecture on the size, shape, and intrinsic viscosity of chains in polymer solutions: a molecular simulation study. *The Journal of chemical physics* **141 21**, 214904 (2014). URL https://api.semanticscholar.org/CorpusID:21434947.

[26] Wijesinghe, S., Perahia, D. & Grest, G. S. Polymer topology effects on dynamics of comb polymer melts. *Macromolecules* **51**, 7621–7628 (2018). URL https://api.semanticscholar.org/CorpusID:104656544.

[27] Liu, Y. *et al.* Recent development in topological polymer electrolytes for rechargeable lithium batteries. *Advanced Science* 2206978 (2023).

[28] Zhou, Y. *et al.* Dicationic tetraalkylammonium-based polymeric ionic liquid with star and four-arm topologies as advanced solid-state electrolyte for lithium metal battery. *Reactive and Functional Polymers* **145**, 104375 (2019).

[29] Zhang, L., Wang, S., Wang, Q., Shao, H. & Jin, Z. Dendritic solid polymer electrolytes: A new paradigm for high-performance lithium-based batteries. *Advanced Materials* 2303355 (2023).

[30] Su, Y. *et al.* Rational design of a topological polymeric solid electrolyte for high-performance all-solid-state alkali metal batteries. *Nature Communications* **13**, 4181 (2022).

[31] Webb, M. A. *et al.* Systematic computational and experimental investigation of lithium-ion transport mechanisms in polyester-based polymer electrolytes. *ACS central science* **1**, 198–205 (2015).

[32] Fong, K. D. *et al.* Ion transport and the true transference number in nonaqueous polyelectrolyte solutions for lithium ion batteries. *ACS central science* **5**, 1250–1260 (2019).

[33] Brandell, D., Priimägi, P., Kasemägi, H. & Aabloo, A. Branched polyethylene/poly (ethylene oxide) as a host matrix for li-ion battery electrolytes: A molecular dynamics study. *Electrochimica acta* **57**, 228–236 (2011).

[34] Cook, A. B. & Perrier, S. Branched and dendritic polymer architectures: Functional nanomaterials for therapeutic delivery. *Advanced Functional Materials* **30**, 1901001 (2020).

[35] Yu, C. *et al.* Molecular dynamics simulation studies of hyperbranched polyglycerols and their encapsulation behaviors of small drug molecules. *Physical Chemistry Chemical Physics* **18**, 22446–22457 (2016).

[36] Javan Nikkhah, S. & Thompson, D. Molecular modelling guided modulation of molecular shape and charge for design of smart self-assembled polymeric drug transporters. *Pharmaceutics* **13**, 141 (2021).

[37] Ahmad, S. *et al.* In silico modelling of drug–polymer interactions for pharmaceutical formulations. *Journal of the Royal Society Interface* **7**, S423–S433 (2010).

[38] Martinho, N. *et al.* Molecular modeling to study dendrimers for biomedical applications. *Molecules* **19**, 20424–20467 (2014).

[39] Polymeropoulos, G. *et al.* 50th anniversary perspective: Polymers with complex architectures. *Macromolecules* **50**, 1253–1290 (2017). URL https://api.semanticscholar.org/CorpusID:100066481.

[40] Dhamankar, S. & Webb, M. A. Chemically specific coarse-graining of polymers: methods and prospects. *Journal of Polymer Science* **59**, 2613–2643 (2021).

[41] Gartner III, T. E. & Jayaraman, A. Modeling and simulations of polymers: a roadmap. *Macromolecules* **52**, 755–786 (2019).

[42] Webb, M. A., Jackson, N. E., Gil, P. S. & de Pablo, J. J. Targeted sequence design within the coarse-grained polymer genome. *Science advances* **6**, eabc6216 (2020).

[43] Patel, R. A., Colmenares, S. & Webb, M. A. Sequence patterning, morphology, and dispersity in single-chain nanoparticles: Insights from simulation and machine learning. *ACS Polymers Au* (2023).

[44] Kosuri, S. *et al.* Machine-assisted discovery of chondroitinase abc complexes toward sustained neural regeneration. *Advanced Healthcare Materials* **11**, 2102101 (2022).

[45] Tamasi, M. J. *et al.* Machine learning on a robotic platform for the design of polymer–protein hybrids. *Advanced Materials* **34**, 2201809 (2022).

[46] Kumar, R. *et al.* Efficient polymer-mediated delivery of gene-editing ribonucleoprotein payloads through combinatorial design, parallelized experimentation, and machine learning. *ACS nano* **14**, 17626–17639 (2020).

[47] Kumar, R. Materiomically designed polymeric vehicles for nucleic acids: quo vadis? *ACS Applied Bio Materials* **5**, 2507–2535 (2022).

[48] Panganiban, B. *et al.* Random heteropolymers preserve protein function in foreign environments. *Science* **359**, 1239–1243 (2018).

[49] Barnett, J. W. *et al.* Designing exceptional gas-separation polymer membranes using machine learning. *Science advances* **6**, eaaz4301 (2020).

[50] Sanchez-Lengeling, B. & Aspuru-Guzik, A. Inverse molecular design using machine learning: Generative models for matter engineering. *Science* **361**, 360–365 (2018).

[51] Kingma, D. P. & Welling, M. Auto-encoding variational bayes. *CoRR* **abs/1312.6114** (2013). URL https://api.semanticscholar.org/CorpusID:216078090.

[52] Dieng, A. B., Kim, Y., Rush, A. M. & Blei, D. M. Avoiding latent variable collapse with generative skip models. In *The 22nd International Conference on Artificial Intelligence and Statistics*, 2397–2405 (PMLR, 2019).

[53] Jin, W., Barzilay, R. & Jaakkola, T. Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*, 2323–2332 (PMLR, 2018).

[54] Gómez-Bombarelli, R. *et al.* Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science* **4**, 268–276 (2018).

[55] Batra, R. *et al.* Polymers for extreme conditions designed using syntax-directed variational autoencoders. *Chemistry of Materials* **32**, 10489–10500 (2020).

[56] Chiu, Y.-H., Liao, Y.-H. & Juang, J.-Y. Designing bioinspired composite structures via genetic algorithm and conditional variational autoencoder. *Polymers* **15**, 281 (2023).

[57] Shmilovich, K. *et al.* Discovery of self-assembling π-conjugated peptides by active learning-directed coarse-grained molecular simulation. *The Journal of Physical Chemistry B* **124**, 3873–3891 (2020).

[58] Kim, S., Schroeder, C. M. & Jackson, N. E. Open macromolecular genome: Generative design of synthetically accessible polymers. *ACS Polymers Au* (2023).

[59] Patel, R. A., Borca, C. H. & Webb, M. A. Featurization strategies for polymer sequence or composition design by machine learning. *Molecular Systems Design & Engineering* **7**, 661–676 (2022).

[60] Patel, R. A. & Webb, M. A. Data-driven design of polymer-based biomaterials: high-throughput simulation, experimentation, and machine learning. *ACS Applied Bio Materials* (2023).

[61] Larson, R. G. The rheology of dilute solutions of flexible polymers: Progress and problems. *Journal of Rheology* **49**, 1–70 (2005).

[62] Lin, T.-S. *et al.* BigSMILES: A structurally-based line notation for describing macromolecules. *ACS Central Science* **5**, 1523–1531 (2019).

[63] Lin, T.-S., Rebello, N. J., Lee, G.-H., Morris, M. A. & Olsen, B. D. Canonicalizing BigSMILES for polymers with defined backbones. *ACS Polymers Au* (2022).

[64] Schneider, L., Walsh, D., Olsen, B. & de Pablo, J. Generative bigsmiles: an extension for polymer informatics, computer simulations & ml/ai. *Digital Discovery* **3**, 51–61 (2024).

[65] Hu, G., Yan, W., Zhou, J. & Shen, B. Residue interaction network analysis of dronpa and a dna clamp. *Journal of theoretical biology* **348**, 55–64 (2014).

[66] Liang, H., Webb, M. A., Chawathe, M., Bendejacq, D. & de Pablo, J. J. Understanding the structure and rheology of galactomannan solutions with coarse-grained modeling. *Macromolecules* **56**, 177–187 (2022).

[67] Plimpton, S. J. Fast parallel algorithms for short-range molecular dynamics. *Journal of Computational Physics* **117**, 1–19 (1993). URL https://api.semanticscholar.org/CorpusID:15881414.

[68] Niepert, M., Ahmed, M. & Kutzkov, K. Learning convolutional neural networks for graphs. In *International conference on machine learning*, 2014–2023 (PMLR, 2016).

[69] Grattarola, D. & Alippi, C. Graph neural networks in tensorflow and keras with spektral [application notes]. *IEEE Computational Intelligence Magazine* **16**, 99–106 (2021).

[70] Xu, K., Hu, W., Leskovec, J. & Jegelka, S. How powerful are graph neural networks? In *International Conference on Learning Representations* (2018).

[71] Abadi, M. *et al.* TensorFlow: Large-scale machine learning on heterogeneous systems (2015). URL https://www.tensorflow.org/. Software available from tensorflow.org.

[72] Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[73] McInnes, L., Healy, J. & Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426* (2018).

[74] Friedman, D. & Dieng, A. B. The vendi score: A diversity evaluation metric for machine learning. *arXiv preprint arXiv:2210.02410* (2022).