Mitigating Data Scarcity in Polymer Property Prediction via Multi-Task Auxiliary Learning

Gabriel A. Pinheiro[1,3], Marcos G. Quiles[1], Juarez L. F. Da Silva[2], and Xiaoli Z. Fern[3] (✉)

[1] Federal University of São Paulo, Institute of Science and Technology, Brazil
{gabriel.pinheiro, quiles}@unifesp.br
[2] University of São Paulo, Institute of Chemistry, Brazil
juarez_dasilva@iqsc.usp.br
[3] Oregon State University, School of Electrical Engineering and Computer Science, USA xiaoli.fern@oregonstate.edu

Abstract. Polymers are fundamental materials with numerous applications in everyday life, making their synthesis, characterization, and property measurement critically important. Machine learning (ML) algorithms offer promising opportunities to accelerate polymer screening with high accuracy, yet significant challenges persist. Unlike small molecules with fixed structures, polymers, especially copolymers formed by polymerizing two or more distinct monomers, can be modeled at multiple scales (atomic, monomer, or repeat-unit level) and exhibit inherent variability due to the stochastic polymerization process, which affects connectivity, chain length, conformations, and compositional complexity. Additionally, the scarcity of labeled polymer data with high-fidelity, experimentally measured properties poses a challenge for ML training. In this work, we tackle these challenges by (1) proposing CoPolyGNN (CoPolymer Graph Neural Network), a multi-scale model that employs a GNN encoder to learn representations of polymer repeating units or individual monomers, combined with an attention-based readout function that aggregates these representations with explicit monomer proportion information; (2) compiling a large dataset of polymers annotated with both simulated and experimentally measured properties; and (3) introducing a supervised auxiliary training framework to mitigate data scarcity in polymer property prediction. We empirically validate CoPolyGNN on datasets of polymer properties measured under real experimental conditions. Our findings demonstrate that augmenting the main task with auxiliary tasks leads to beneficial performance gains. Consequently, our work provides a neural architecture and training framework enabling practitioners to predict polymer properties from simple text notations of repeat units or monomers and their proportions, achieving strong performance even with limited training data. [4]

Keywords: Multitask Learning · Polymers · Property Prediction.

[4] Code available at https://github.com/CIDAG/CoPolyGNN

1 Introduction

Polymers are macromolecules composed of repeating chemical units covalently bonded to form long chains or networks [21], with versatile properties that make them essential in everyday products from packaging, synthetic clothing to advanced biomedical devices [10]. As technology progresses, it is essential to design polymers with properties that meet the evolving needs of society, such as high-energy-density capacitors [13], molecular imprinting [20], gas separation [27], and biocompatible or (bio)degradable materials [28]. However, designing polymers involves navigating a vast space of possible polymeric materials, where few structure-property relationships are known, and the screening of potential polymers through wet-lab experiments remains expensive and time-consuming, while simulations, such as those relying on force fields, often struggle to reproduce experimental properties [37, 4].

Polymer informatics has emerged to accelerate polymer discovery by leveraging machine learning (ML) for property prediction [37]. However, polymers differ significantly from small molecules, whose fixed and well-defined structures have enabled substantial success using ML models such as graph neural networks (GNNs). The exact structure of a polymer is complex and challenging to characterize precisely due to the stochastic nature of polymerization processes, resulting in inherent variations in chain length, sequence, and network architecture [36]. This poses challenges in adapting ML models originally designed for small and well-defined molecular structures to polymers. As a result, polymers are typically modeled based on their fundamental building blocks, which are small molecules such as monomers or repeating units. In the literature, several studies have focused on polymer property prediction, either using molecular fingerprints or one-hot encoding (OHE) applied to these building blocks [34, 4]. An alternative approach has involved using GNNs to learn fingerprints in an end-to-end fashion, typically based on the graph structure of the monomers [9, 23].

Additionally, polymer informatics also suffers from the challenge of data scarcity, which limits the accuracy and generalizability of predictive models. Existing work has considered multi-task learning (MTL) to mitigate the data scarcity issue through shared learning across related multiple related tasks [25]. In this context, both molecular fingerprints and GNNs have been explored using data from simulations, experiments, or their combination. For example, [28] demonstrated that training an MTL model with both simulated and experimental data improved the accuracy of predicting experimental ring-opening polymerization enthalpy. MTL aims to solve multiple tasks simultaneously, but in practical applications, there is often a primary focus on a specific polymer property. In such scenarios, others tasks can serve as auxiliary tasks, facilitating knowledge transfer and improving generalizability of the learned model. This strategy, known as multi-task auxiliary learning (MTAL), helps the model prioritize the main task while leveraging auxiliary tasks for knowledge transfer. MTAL could provide a promising solution to the challenges in polymer informatics, however,

the effectiveness of this approach depends on careful task selection, as inappropriate auxiliary tasks can introduce negative transfer.

This work presents a three hold effort to address the above mentioned challenges.

- First, we propose CoPolyGNN (CoPolymer Graph Neural Network), a neural polymer encoder that integrates a GNN-based monomer encoder with an attention-based readout function to learn copolymer representations at multiple scales while also handling simpler cases like homopolymers (polymer comprised of a single monomer or repeating unit). Unlike prior polymer representation models, CoPolyGNN explicitly incorporates monomer structural information and their proportions through an attention mechanism, leading to more expressive polymer representations.
- Second, we curate a comprehensive polymer property dataset from existing literature, comprising over 70.000 polymers and their experimental and simulated properties across 35 unique properties (e.g., chemical reactivity, electronic, thermal, and other polymer-relevant characteristics) obtained through simulations and/or experiments. This large-scale dataset serves as a valuable resource for training robust polymer property prediction models, particularly in data-limited scenarios.
- Third, we explore multiple MTAL strategies to effectively address data scarcity. Specifically, we investigate two distinct classes of MTAL approaches: (1) treating all tasks distinct from the main task as predefined auxiliary tasks and integrating them into a unified loss function using weighting heuristics such as Gradient Cosine Similarity (GCS) [6] and Online Learning for Auxiliary Losses (OL-AUX) [18]; and (2) assuming that auxiliary tasks are unknown and dynamically learning an optimal set of auxiliary tasks, as in Task Affinity Grouping (TAG) [8]. By systematically evaluating these strategies, we show that MTAL achieves superior performance in most tasks compared to the single-task learning (STL) version of CoPolyGNN, as well as traditional methods such as fingerprint-based descriptors. Furthermore, our top result improves upon the strongest baseline by approximately 50%. These findings underscore the potential of MTAL in leveraging auxiliary tasks to enhance learning and generalization, offering a promising approach to mitigating data scarcity in polymer informatics.

Together, these contributions provide a self-contained pipeline that allows practitioners to build polymer property prediction models with limited experimental data while leveraging a rich set of auxiliary tasks and a large-scale polymer dataset. Our MTAL framework offers a scalable and generalizable solution for enhancing polymer informatics through data-efficient learning.

2 Background

This section first introduces key aspects of polymers and then guides the reader to the application of ML in polymer science for property prediction.

2.1 Basic Concepts of Polymers

Polymers have always been with us, yet the scientific understanding of their nature became clear in the mid-20th century with the rise of plastics. The term originates from the Greek words *poly*, meaning many, and *mer*, meaning unit. This etymology reflects their fundamental structure, as they are macromolecules generated through polymerization, in which small molecules called monomers join together to form long chains. After being incorporated into the polymer, monomers serve as structural units, establishing stable covalent bonds with adjacent units. If a polymer is composed of a single type of monomer, it is classified as a homopolymer. If it is composed of two or more distinct monomers, it is called a copolymer. Additionally, the arrangement of each monomer in the polymer chain influences the classification of copolymers into random, alternating, block, and graft types [7].

One fundamental concept in polymer science is the repeating unit, which is the smallest structural segment of the polymer that recurs along the chain. The repeating and structural units of a polymer may differ. For instance, in a homopolymer, both units are identical, whereas in copolymers, the arrangement of multiple monomeric units causes the repeating and structural units to differ. Moreover, such arrangements lead to diversity in architecture, composition, and patterning, making copolymers some of the most commercially important polymers, such as those utilized in plastics, rubbers, and coatings. For most synthetic polymers, structural characteristics (e.g., chain length, architecture) can also be influenced by stochastic factors during polymerization, regardless of the method. This results in polymers being polydisperse in molecular weight, meaning that the molecular weights of the polymers follow a distribution. Therefore, we typically refer to their average molecular weight, and experimental measurements reflect this average [7, 21]

In short, polymers pose challenges in structural representation, as their precise connectivity beyond polymerization points remains uncertain. While some polymerization methods reduce variations and yield more uniform structures, a perfectly predictable structure is not guaranteed. For standard ML, which usually relies on well-defined input representations, this structural variability requires a careful choice of representation strategies that capture both structural information and uncertainty. This has resulted in polymers typically being described by their building blocks, such as their monomers or repeating units.

2.2 Machine Learning for Polymer Property Prediction

Given the complexity of polymers and the vast design space, computational approaches, such as physical molecular modeling and data-driven methods, have become attractive for predicting their properties by offering a lower-cost, faster alternative to experimental characterization. In this realm, polymer informatics leverages ML to predict diverse polymer properties, such as thermodynamic and mechanical properties, among others [2]. Such strategies often utilize molecular descriptors, GNNs, or language model-based methods to extract representations

from the monomers or repeating units that describe the polymer. Fingerprints are a traditional type of molecular descriptor, and they rely on domain expertise to capture the presence or absence of chemical substructures. Popular options include the hierarchical editing language for macromolecules, which is widely used in pharmaceutical and industrial applications, the extended-connectivity fingerprint, Molecular ACCess System (MACCS) keys, and others [13, 39, 31]. Furthermore, OHE and topological descriptors, such as those available in RDKit, are also commonly applied in these processes [3, 24].

For example, [39] proposed a hierarchical fingerprinting scheme with four levels to capture physical and chemical interactions that contribute to predicting gas permeation properties. This hierarchical approach starts at the atomic scale by counting the occurrence of atomic triples. The second level involves counting building blocks from a predefined list, the third incorporates quantitative structure-property relationship descriptors, and the fourth accounts for morphological descriptors. Instead of focusing on hierarchical fingerprinting, [19] proposed a data augmentation method based on the iterative rearrangement of polymer fragments. Specifically, it follows a process similar to window slicing, in which the repeating unit is disassembled into smaller fragments and recombined in a way that preserves the polymer backbone. The recombined structure is then processed into molecular fingerprints.

In contrast, there are studies that extract polymer representations without relying on handcrafted features. One such direction is GNN-based methods, in which a graph is used to describe the molecular connectivity of the building block (i.e., monomers or repeating units), where vertices represent atoms and edges represent bonds. In cases involving copolymers, the polymer representation is typically a weighted sum of monomer representations, with weights based on their ratios [35, 9]. This approach is also a common practice in fingerprint methods and OHE [25, 24]. [2] introduced a GNN that uses a graph with parameterized stochastic edges to capture the average structure of the repeating unit. Following up on this, [9] extended such a framework under self-supervised learning, considering node-, edge-, and graph-level pre-training tasks to improve representation learning as a way to reduce the impact of limited data on supervised tasks.

Another approach to representation learning involves textual token-based models, particularly transformer-based methods like TransPolymer [33], PolyBERT [15], and PolyCL [38]. In addition, MTL has been explored to exploit correlations between properties, improving generalizability and mitigating data scarcity [14, 16]. It is typically implemented as a hard parameter model, relying on either GNN- [26, 32, 23] or transformer-based [22, 30, 22] encoders. For example, [23] introduced PolymerGNN, an MTL framework that concatenates the embeddings from a GNN along with resin properties to predict the glass transition temperature and inherent viscosity of homopolymers and copolymers. A different approach uses a fingerprint-based representation in a multilayer perceptron to predict polymer properties from in-house simulation datasets and experimental measurements reported in the literature [14].

Nevertheless, most existing methods still rely on either monomer-based or repeating-unit-based representations separately. For this reason, we propose an ML model that captures polymer structure with a multi-scale approach within a unified architecture and replaces a simple weighted sum of monomer embeddings with an attention-based readout mechanism that dynamically learns the contribution of each polymer component. This flexible design extends its applicability to a broader range of polymer architectures, from homopolymers to copolymers. Furthermore, we explore our model within an MTAL framework, which, to the best of our knowledge, has not been previously applied to polymer data to mitigate data scarcity.

3 Methodology

In this section, we introduce the core workings of CoPolyGNN and the techniques for training it in the MTAL setting, as illustrated in Figure 1.

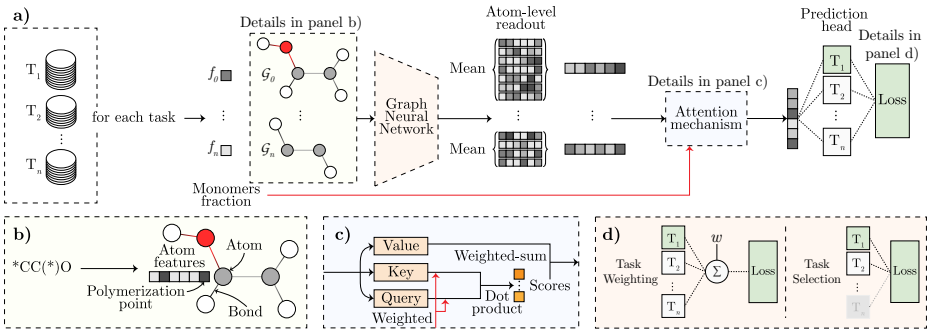

Fig. 1. Overview of the proposed CoPolyGNN in an MTAL setting. (a) For each task in the pool of polymer datasets, we first build the graph representation of the monomers, which are fed to the first component of CoPolyGNN, a GNN encoder that produces monomer-level embeddings. These monomer-level embeddings are passed through a readout function to generate a polymer-level embedding for the predictive tasks. After this, an MTAL scheme is used to assign importance to each auxiliary predictive task according to the main task. (b) Illustrates the encoding process of p-SMILES into a graph and highlights the polymerization atom feature to indicate the connectivity between monomers. (c) Shows the attention-based readout function that weights the importance of monomers for the predictive task. (d) Depicts the two types of MTAL strategies used to train CoPolyGNN.

3.1 Polymer Representation

We express a polymer as a set of monomer graphs $\{\mathcal{G}_0, \mathcal{G}_1, \ldots, \mathcal{G}_n\}$, where each $\mathcal{G}_i$ is built on the p-SMILES (Simplified Molecular-Input Line-Entry System) no-

tation[5] and is assigned a monomer fraction f_i such that $\sum_{i=0}^{n} f_i = 1$. p-SMILES extends SMILES strings by incorporating the '*' character to represent polymerization points in monomers. We convert p-SMILES into a graph following the natural molecular encoding, where atoms are represented as nodes and chemical bonds are treated as edges of the graph. Each atom is represented by a feature vector, which includes the atom type (with OHE), atomic number, aromaticity, hybridization states (sp, sp^2, and sp^3), and the number of explicitly attached hydrogen atoms, with the addition of a binary feature, where 1 indicates a polymerization point and 0 indicates the absence of one. Figure 1(b) shows this process of reading the p-SMILES string and extracting atomic descriptors using the RDKit Python package (version 2024.03.6) [17].

3.2 CoPolyGNN

The representation of a polymer requires more than a conventional molecular graph, as their structure emerges from many factors, such as the combination of multiple monomers and how often each appears. CoPolyGNN was designed to handle such complexity by combining two processing blocks to learn a polymer embedding. The first block is a Graph Isomorphism Network (GIN), which operates on each monomer graph in $\{\mathcal{G}_0, \mathcal{G}_1, \ldots, \mathcal{G}_n\}$ for the given polymer. GIN was chosen for its theoretical equivalence to the Weisfeiler-Lehman graph isomorphism test and its strong performance across diverse chemical tasks [12]. It initiates encoding with a message-passing operation that updates each atom's representation over k iterations by aggregating information from its neighbors and itself, as defined in Equation 1.

$$\begin{aligned} m_v^{(k)} &= \left(1 + \epsilon^{(k)}\right) \cdot h_v^{(k-1)} + \sum_{u \in \mathcal{N}(v)} h_u^{(k-1)} \ , \\ h_v^{(k)} &= \mathrm{MLP}^{(k)} \left(m_v^{(k)}\right) \ , \end{aligned} \tag{1}$$

where $h_v^{(k-1)}$ denotes the representation of atom v at iteration $k-1$ (note that h_v^0 is the initial feature vector of the atom, defined when constructing the graph), $\mathcal{N}(v)$ represents the neighboring atoms of v, $\epsilon^{(k)}$ is either a learnable parameter or a fixed scalar, and $\mathrm{MLP}^{(k)}$ represents the multilayer perceptron associated with the k-th iteration. After k iterations, each atom's representation encodes the structural information from its k-hop neighborhood, and monomer-level representations are then obtained via average pooling.

The second block employs a readout function that aggregates monomer-level representations into a polymer-level representation using attention mechanism, as depicted in Figure 1(c). The attention mechanism dynamically learns the context-dependent contributions of individual monomer units to the overall polymer behavior. This design captures the fact that interactions between monomers

[5] The repeating unit can be interpreted as a simplified form of the monomer-based representation, where a single graph is used and $f_0 = 1$.

in a polymer are neither strictly linear nor merely additive. Monomer-level representations are projected through three linear layers to compute the query (Q), key (K), and value (V) matrices. A dot product is then calculated between the transformed Q and the sum of K across monomers to capture interactions between monomers and the global representation. The result is normalized by the square root of the key dimensionality (d_k), followed by a softmax operation, $\text{softmax}\left(\frac{(FQ)(\sum_i F_i K_i)^\top}{\sqrt{d_k}}\right)V$, where F is a diagonal matrix with f_i on the diagonal, i.e., $F = diag(f_1, f_2, ..., f_n)$. This scales the query and key vectors of the i-th monomer by its proportion f_i before computing the dot product, ensuring that the monomer contribution to the attention scores are weighted according to their fractions in the polymer.

3.3 CoPolyGNN Multi-Task Auxiliary Learning

In this section, we discuss the MTAL training strategy for CoPolyGNN. Our MTL data contains polymer data on 35 different tasks, each with varying number of examples. During training, we adopt a sampling approach wherein each task contributes an equally sized batch in each iteration. This ensures balanced representation across tasks, irrespective of the data size of individual tasks. After each batch is processed, the auxiliary task selection strategy determines how to update the parameters based on the relevance of different tasks, as shown in Figure 1(d). Specifically, we examine three strategies for MTAL: GSC [6], OL-AUX [18], and TAG [8].

The first two methods, GCS and OL-AUX, integrate auxiliary tasks into a unified loss function $\mathcal{L}$ by assigning heuristic-based weights w_i to each auxiliary task i:

$$\mathcal{L} = \frac{1}{N_{\text{main}}} \sum_{k=1}^{N_{\text{main}}} \left(y_{\text{main}}^{(k)} - \hat{y}_{\text{main}}^{(k)}\right)^2 + \sum_{i=1}^{T} \frac{w_i}{N_i} \sum_{k=1}^{N_i} \left(y_i^{(k)} - \hat{y}_i^{(k)}\right)^2 , \tag{2}$$

where N_i represents the number of data points for task i.

Specifically, the key idea of GCS is to include an auxiliary task in the total loss only if its gradient exhibits non-negative cosine similarity with the main task gradient on shared parameters. Therefore, GCS assigns $w_i = 1$ when the cosine similarity between the gradients is non-negative and $w_i = 0$ otherwise [6].

OL-AUX follows a similar idea, but it treats w_i as model parameters with its own loss function. This loss function measures the negative dot product between gradients of the main task, $\nabla_{\theta_t}\mathcal{L}_{\text{main}}(\theta_t)$, and each auxiliary task, $\nabla_{\theta_t}L_i(\theta_t)$, at iteration t. The gradient update for w_i is given by $\frac{\partial \mathcal{L}_{\text{weights}}}{\partial w_i} = -\alpha\nabla_{\theta_t}\mathcal{L}_{\text{main}}(\theta_t)^T\nabla_{\theta_t}L_i(\theta_t)$, where θ_t represents the model parameters and α is the learning rate [18].

In contrast, TAG selects an optimal set of auxiliary tasks without using a unified weighted loss function. To achieve this, it maximizes inter-task affinity scores, denoted by $Z_{i\to j}^t$, a metric that quantifies the influence of auxiliary task

i on the main task j [8]. In this context, i represents a set of auxiliary tasks, and j corresponds to the main task, with $Z^t_{i\to j}$ formally defined in Equation 3.

$$Z^t_{i\to j} = 1 - \frac{L_j(x^t, \theta^{t+1}_{s|i}, \theta^t_j)}{L_j(x^t, \theta^t_s, \theta^t_j)} \ . \tag{3}$$

To compute this quantity, the model relies on the parameters of the shared encoder ($\theta^{t+1}_{s|i}$ or θ^t_s) and the parameters of the prediction head for task j, denoted as θ^t_j. Note that in $Z^t_{i\to j}$, the numerator utilizes the shared encoder parameters after being updated for the auxiliary task i ($\theta^{t+1}_{s|i}$), while the denominator employs the shared encoder parameters trained to predict the main task j (θ^t_s). Consequently, a positive value of $Z^t_{i\to j}$ signifies that updating the shared parameters reduces the loss of the main task compared to the values of the original parameters. At each training iteration, a batch of data is sampled for each task, and the model computes the mean squared error. The gradients are then backpropagated only through the parameters associated with the corresponding tasks. TAG selects the subset of auxiliary tasks that maximizes $Z^t_{i\to j}$ as the optimal support set for the main task. We then use this selected auxiliary tasks to fine-tune the weights of the model.

4 Dataset

We curated a dataset of 70 827 polymers by compiling data from recent publications on polymers published between 2020 and 2024, where SMILES strings are used to describe the building blocks of the polymers [2, 11, 14, 13, 1, 5, 29, 28, 23, 24, 27]. This dataset encompasses several chemical species, H, C, N, O, F, Na, Si, P, S, Cl, Ge, Br, and I, as well as homopolymers and copolymers with structures such as linear, branched, and cyclic, which is an indicator of the diversity within it. Moreover, it spans 35 distinct properties, some of which have experimental and simulated data, while others are available only from simulations or experiments. Figure 2 illustrates the distribution of diverse properties in the dataset, which can include chemical reactivity, electronic-nuclear properties, thermal properties, and others. We also indicate whether the data come from experiments (in blue) or simulations (in black). As can be seen, 3280 polymers (about 5 %) correspond to experimental data, which points to the limited size of the experimental dataset and the challenges associated with gathering such data.

We applied a series of preprocessing steps to prepare the datasets. First, we ensured consistency by reading p-SMILES with RDKit. Some datasets represented polymerization points using special characters such as '[Th]' and '[Ce]', which we replaced with '*', the standard notation for polymerization points. Next, we canonized p-SMILES to standardize their representation and removed duplicate entries within each task by averaging target values and merging identical strings. Overall, polymers were described using two common formats: (1) monomers with their respective fractions and (2) repeating units. Of the total

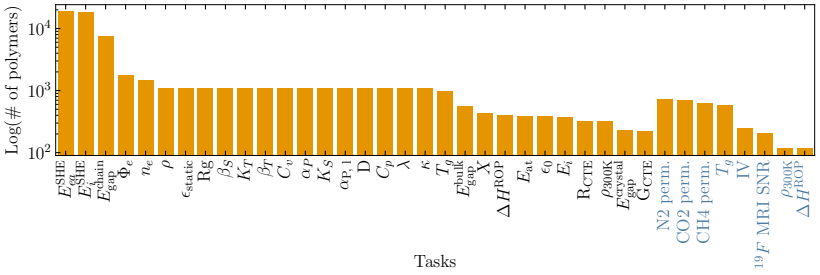

Fig. 2. Distribution of the number of polymers per task: Electron affinity with respect to the standard hydrogen potential - SHE (E_{ea}^{SHE}), Ionization potential with respect to the SHE (E_{i}^{SHE}), Bandgap - Chain ($E_{\mathrm{gap}}^{\mathrm{chain}}$), Electron injection barrier (Φ_e), Refractive index (n_e), Density (ρ), Static dielectric constant ($\epsilon_{\mathrm{static}}$), Radius of gyration (Rg), Isentropic compressibility (β_S), Bulk modulus (K_T), Compressibility (β_T), Constant volume (C_v), Volume expansion coefficient (α_P), Isentropic bulk modulus (K_S), Linear expansion coefficient ($\alpha_{\mathrm{P,\,l}}$), Self-diffusion coefficient (D), Constant pressure (C_p), Thermal conductivity (λ), Thermal diffusivity (κ), Glass transition temperature (T_g), Bandgap - Bulk ($E_{\mathrm{gap}}^{\mathrm{bulk}}$), Crystallization tendency (X), Ring-opening polymerization enthalpy (ΔH^{ROP}), Atomization energy (E_{at}), Dielectric constant (ϵ_0), Ionization energy (E_i), Rubber coefficient of thermal expansion ($\mathrm{R_{CTE}}$), Density at 300K ($\rho_{300\mathrm{K}}$), Bandgap - Crystal ($E_{\mathrm{gap}}^{\mathrm{crystal}}$), Glass coefficient of thermal expansion ($\mathrm{G_{CTE}}$), N2 permeability (N2 perm.), CO2 permeability (CO2 perm.), CH4 permeability (CH4 perm.), Inherent viscosity (IV), and ^{19}F magnetic resonance imaging signal-to-noise ratio (^{19}F MRI SNR). Experimentally measured properties are shwon with blue labels, and simulated properties are shown with black labels. Three properties include both experimental and simulated data.

polymers in our dataset, 33 342 are described using repeating units, while the remainder are characterized by monomers. Among these, 36 994 polymers are composed of two monomers, with the maximum number of monomers in a polymer reaching 7. Additionally, we standardized property units across datasets for consistency. We did not merge polymers with identical p-SMILES when properties were derived from different simulation methodologies, as a way to enhance diversity and incorporate additional data into our MTAL framework.

Moreover, our final dataset contains polymers relevant for a variety of applications, such as gas separation membranes, high-voltage insulation, and controlled drug release. Yet, property co-occurrence exists. The mean number of polymers that overlap in property pairs is 252 and the median is 49. This is due to the relatively small size of most datasets, typically under 1000 samples, and the broad diversity of polymer applications. Among the 703 possible task pairs, 454 have fewer than 100 co-occurring polymers, while three tasks exhibit minimal co-occurrence across nearly all task pairs. This diversity and overlap create a possible scenario for MTAL.

5 Experiment Configuration

First, we perform hyperparameter optimization by modifying the batch size to 16, 32 and 64, the latent representation dimensionality to 32 and 64, dropout to $1{\times}10^{-1}$, $2{\times}10^{-1}$ and $4{\times}10^{-1}$, and the Adam optimizer learning rate to 1×10^{-2} and 1×10^{-3}. Optimization was conducted using MTL without auxiliary task selection to ensure unbiased hyperparameter tuning. We performed a grid search where, for each hyperparameter combination, training involved sequentially sampling a batch per task, computing the mean squared error, and backpropagating gradients only through the corresponding task parameters. The best hyperparameter combination was selected based on the performance of the model across the majority of tasks. Subsequently, we evaluated the impact of incorporating the attention mechanism compared to that of weighting the monomer embeddings according to their respective fractions. We again selected the optimal model as the one that consistently achieved superior performance in the majority of property prediction tasks.

For MTAL, all models were trained for 10 epochs, each epoch containing a number of iterations proportional to the batch count of the largest dataset. A k-fold-like approach ($k = 10$) was adopted for the experimental datasets (i.e., main task), where $k-2$ folds were used for training, 1 fold for validation, and the remaining fold for testing. In contrast, the simulated datasets were exclusively used for training, as they were designed for auxiliary tasks. Furthermore, we saw that OL-AUX and TAG have update rules (see Section 3.3) that involve additional parameter updates. This introduces computational overhead, so updates for MTAL here were performed every T steps. For OL-AUX, we initialized the weights of all auxiliary tasks to 1 and updated them every 10 epochs. For TAG, we set the step interval to 10 as well, and fine-tuning was performed over 10 epochs using the optimal auxiliary task group.

As a reference for evaluating the MTAL framework, we trained a random forest model on individual tasks. Specifically, we conducted experiments using three different descriptors commonly used for polymers: (1) a set of molecular features available in RDKit, (2) the MACCS fingerprint, and (3) the OHE representation of the monomers. For polymers composed of multiple monomers, we computed a weighted sum of the descriptors of individual monomers to construct a single feature vector representing the entire polymer. This setup allows us to explore whether models based on handcrafted features and classical learners provide a viable alternative. For additional comparison, we also included an STL version of our model to evaluate the added benefit of MTAL compared to the core architecture.

6 Results and Discussion

This section presents results obtained from our experiments. We began by optimizing the hyperparameters of the proposed model, followed by an investigation of the attention mechanism for the readout function, and finally evaluating the performance of our MTAL model in predicting 9 experimental properties.

6.1 Optimizing Hyperparameters

Hyperparameter choices are essential in guiding a model to its optimal performance. To determine the ideal model configuration, we started our investigation with a grid search using the hyperparameter space defined in Section 5. The optimal settings identified were a learning rate of 1×10^{-3}, a batch size of 16, a dropout rate of 0.1 and a latent representation dimensionality of 64. Among the hyperparameters, we observed that the learning rate had the most significant impact on reducing error across most properties, while the others introduced only marginal improvements.

Moreover, we determined the number of iterations per epoch by considering the batch count of the largest dataset. Given that the optimal batch size was 16, each training epoch consisted of 1036 iterations. For the experimental datasets, which are the tasks we carry out and aim to improve, this number of iterations proved sufficient to achieve convergence, as a full training cycle is equivalent to 1726 (280) passes through the smallest (largest) experimental datasets. Throughout the training iterations, we observed that the losses initially decreased steadily but tended to stabilize toward the end, with no significant improvement after a certain point. However, there are instances where the training loss decreases rapidly, but the validation loss does not follow the same trend, creating a noticeable gap between them. Such behavior is evident in tasks associated with the smallest datasets, containing 116, 117, and 204 polymers, respectively.

6.2 Impact of the Attention Mechanism

This study examines the impact of using an attention mechanism as part of the readout function, as detailed in Section 3.2. Table 1 shows the results and compares them with a readout function that weights the monomer embeddings according to their respective fractions. We report the performance in terms of the MAE of the validation set over 10 folds. As can be seen, the attention-based readout function generally tends to minimize both the mean and the standard deviation of the error across the properties, showing a subtle, but consistent improvement in performance. In particular, the readout function that uses monomer fractions was shown as a competitive alternative and outperformed the attention-based approach in the ^{19}F MRI SNR prediction task.

We believe that the proposed readout function allowed the model to capture more complex monomer interactions and prioritized their contributions to polymer properties by dynamically adjusting their weights. Moreover, its generalization capability is likely to benefit significantly from the use of larger datasets, as it is naturally a data-hungry method. At the same time, we highlight that the simplicity of the fraction-based method promotes interpretability while still delivering strong performance.

6.3 Multi-Task Auxiliary Learning

Here, we assessed the performance of CoPolyGNN under MTAL setting in predicting experimental polymer properties using the hyperparameter configuration

Table 1. Comparison of the performance of our model using an attention-based readout (w/ attn) and a fraction-weighting readout (w/o attn) for predicting experimental properties: ΔH^{ROP} ($\mathrm{kJ\,mol^{-1}}$), $\rho_{300\mathrm{K}}$ ($\mathrm{g\,cm^{-3}}$), ^{19}F MRI SNR (1), T_g (K), IV ($\mathrm{dL\,g^{-1}}$), and CH_4, CO_2, and N_2 permeability (Barrer).

	w/ attn	w/o attn
ΔH^{ROP}	6.33 ± 2.67	6.51 ± 2.30
$\rho_{300\mathrm{K}}$	0.07 ± 0.03	0.08 ± 0.03
^{19}F MRI SNR	6.42 ± 1.15	6.18 ± 1.15
T_g	4.12 ± 2.01	4.20 ± 1.42

	w/ attn	w/o attn
IV	0.04 ± 0.02	0.05 ± 0.02
T_g	17.75 ± 2.00	19.00 ± 4.44
CH_4 perm.	0.31 ± 0.02	0.33 ± 0.04
CO_2 perm.	0.26 ± 0.02	0.29 ± 0.03
N_2 perm.	0.28 ± 0.02	0.30 ± 0.02

that yielded the best results. These results were compared against three descriptors commonly employed for polymer property prediction as a way to provide a contrast and evaluate whether traditional feature-engineering approaches could offer a viable alternative to our design. For the molecular descriptors, we constructed polymer-level representations by computing a weighted sum of the monomer-level features. Table 2 summarizes the test set errors across the 10 folds, where the same k-fold split was consistently applied to all models.

Table 2. Test Error for Predicting Experimental Polymer Properties. The best result for each task is shown in bold and ties occur where rounding masks marginal differences. See Table 1 for property units.

	STL (Random Forest)			MTAL (Ours)		
	RDKit	MACCS	OHE	GCS	OL-AUX	TAG
ΔH^{ROP}	**8.26 ± 2.79**	10.73 ± 3.69	-	9.75 ± 3.53	9.20 ± 2.65	9.05 ± 2.77
$\rho_{300\mathrm{K}}$	0.12 ± 0.03	0.13 ± 0.04	-	**0.12 ± 0.05**	0.12 ± 0.04	0.12 ± 0.05
^{19}F MRI SNR	8.48 ± 2.23	7.65 ± 1.66	**6.97 ± 1.20**	8.21 ± 1.96	8.94 ± 1.53	8.06 ± 1.47
T_g	6.84 ± 2.40	6.53 ± 2.30	12.28 ± 4.09	6.20 ± 1.93	6.45 ± 2.11	**5.61 ± 1.78**
IV	0.06 ± 0.02	0.06 ± 0.02	0.06 ± 0.02	0.06 ± 0.02	0.06 ± 0.02	**0.05 ± 0.02**
T_g	26.49 ± 4.50	27.24 ± 4.92	-	28.01 ± 4.27	26.48 ± 4.47	**24.02 ± 4.79**
CH_4 perm.	0.45 ± 0.05	0.52 ± 0.05	-	0.45 ± 0.04	0.44 ± 0.04	**0.23 ± 0.04**
CO_2 perm.	0.40 ± 0.05	0.45 ± 0.07	-	0.37 ± 0.05	0.36 ± 0.05	**0.20 ± 0.04**
N_2 perm.	0.41 ± 0.04	0.48 ± 0.05	-	0.38 ± 0.05	0.38 ± 0.05	**0.32 ± 0.03**

The first important point to highlight is that we aimed to train a model capable of leveraging the maximum amount of information available about the polymers. To achieve this, the T_g experimental datasets were divided into two subsets because some polymers with this property included additional features beyond the p-SMILES notation, for instance, the molecular weight. This feature was reported by [23]. In these cases, we concatenated the additional features with the representation learned by the CoPolyGNN and used this combined input for the prediction head. Secondly, our MTAL model outperformed the baseline predictors in 7 out of 9 tasks, as shown in Table 2. While the overall

performance gains were modest, the last three properties in Table 2 showed significant improvements, with approximately 48 %, 50 %, and 22 % gains over the best baseline results for CH_4, CO_2, and N_2 permeability, respectively.

Among MTAL strategies, TAG showed consistent prominence by outperforming the others with a significant margin and achieving the best performance on 6 tasks. However, during task selection for this method, no single auxiliary task appeared consistently across all folds, which may be influenced by the variability present in small datasets and the sensitivity of the model to data splits. In comparison, heuristic-based task weighting approaches showed marginal improvements over the baseline, with more favorable results observed in nearly half of the cases. Moreover, the experimental properties presented in Table 2 are organized by dataset size, with the smallest datasets positioned at the top and the largest at the bottom. We observed that, overall, the models struggled more with datasets containing fewer samples, even when the best results were achieved. This highlights the challenges posed by limited data availability.

We also investigated whether the improvement shown in Table 2 stemmed from the inclusion of auxiliary tasks or from the predictive power of the model itself. To address this question, Table 3 compares the results of our MTAL with a version of our model trained on the main task without auxiliary tasks, i.e., STL. We report the results obtained on the test set averaged over 10 folds. As a result, the MTL model outperformed STL across all tasks, which demonstrates the contribution of auxiliary tasks to improved performance on the main task.

Table 3. Comparison of the performance of STL and MTAL for predicting experimental properties. See Table 1 for property units.

	STL	MTAL
ΔH^{ROP}	9.43 ± 3.34	9.05 ± 2.77
$\rho_{300\mathrm{K}}$	0.13 ± 0.05	0.12 ± 0.05
^{19}F MRI SNR	8.67 ± 1.37	8.06 ± 1.47
T_g	6.61 ± 2.20	5.61 ± 1.78

	STL	MTAL
IV	0.06 ± 0.02	0.05 ± 0.02
T_g	28.14 ± 4.99	24.02 ± 4.79
CH_4 perm.	0.41 ± 0.03	0.23 ± 0.04
CO_2 perm.	0.38 ± 0.05	0.20 ± 0.04
N_2 perm.	0.41 ± 0.05	0.32 ± 0.03

7 Conclusion

In this work, we addressed key challenges in polymer informatics to predict polymer properties by proposing CoPolyGNN, a GNN-based representation model for copolymers, curating a large-scale polymer dataset, and exploring MTAL strategies to tackle data scarcity. We conducted several experiments to optimize hyperparameters and investigated 3 techniques for MTAL, namely, GSC, OL-AUX, and TAG. Our results demonstrate that CoPolyGNN, when combined with MTAL, consistently outperforms STL and baseline models, achieving up to

50 % improvement in prediction accuracy for gas permeability and notable gains in other polymer properties. This validates the practical applicability of our approach across diverse experimental tasks, from glass transition temperature to gas separation performance. Moreover, our findings highlight how data-efficient learning and a flexible, multi-scale model can improve ML performance despite data scarcity. This is particularly relevant for polymer science, where data availability remains a bottleneck for computational design, and a model capable of handling the inherent structural variability of polymers is essential. Future work will investigate how the proposed framework generalizes across different polymer datasets and tasks, with a focus on identifying factors that influence task transferability.

Acknowledgments. The authors gratefully acknowledge support from FAPESP (São Paulo Research Foundation) and Shell, projects No. 2017/11631 − 2, 2021/08852 − 2, 2022/13536−5, and 2022/09285−7. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) – Finance Code 001. The authors also thank for the infrastructure provided to our computer cluster by the Institute of Science and Technology – Campus São José dos Campos. Special thanks to Professor Cory M. Simon from Oregon State University for his valuable contributions and mentorship, which played a crucial role in the completion of this work.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Afzal, M.A.F., Browning, A.R., Goldberg, A., Halls, M.D., Gavartin, J.L., Morisato, T., Hughes, T.F., Giesen, D.J., Goose, J.E.: High-throughput molecular dynamics simulations and validation of thermophysical properties of polymers for various applications. ACS Applied Polymer Materials **3**(2), 620–630 (2020)
2. Aldeghi, M., Coley, C.W.: A graph representation of molecular ensembles for polymer property prediction. Chemical Science **13**(35), 10486–10498 (2022)
3. Arora, A., Lin, T.S., Rebello, N.J., Av-Ron, S.H., Mochigase, H., Olsen, B.D.: Random forest predictor for diblock copolymer phase behavior. ACS Macro Letters **10**(11), 1339–1345 (2021)
4. Cencer, M.M., Moore, J.S., Assary, R.S.: Machine learning for polymeric materials: an introduction. Polymer International **71**(5), 537–542 (2022)
5. Choi, S., Lee, J., Seo, J., Han, S.W., Lee, S.H., Seo, J.H., Seok, J.: Automated bigsmiles conversion workflow and dataset for homopolymeric macromolecules. Scientific data **11**(1), 371 (2024)
6. Du, Y., Czarnecki, W.M., Jayakumar, S.M., Farajtabar, M., Pascanu, R., Lakshminarayanan, B.: Adapting auxiliary losses using gradient similarity. arXiv preprint arXiv:1812.02224 (2018)
7. Ebewele, R.: Polymer Science and Technology. CRC Press (2000)
8. Fifty, C., Amid, E., Zhao, Z., Yu, T., Anil, R., Finn, C.: Efficiently identifying task groupings for multi-task learning. Advances in Neural Information Processing Systems **34**, 27503–27516 (2021)

9. Gao, Q., Dukker, T., Schweidtmann, A.M., Weber, J.M.: Self-supervised graph neural networks for polymer property prediction. Molecular Systems Design & Engineering **9**(11), 1130–1143 (2024)
10. Geyer, R., Jambeck, J.R., Law, K.L.: Production, use, and fate of all plastics ever made. Science advances **3**(7), e1700782 (2017)
11. Hayashi, Y., Shiomi, J., Morikawa, J., Yoshida, R.: Radonpy: automated physical property calculation using all-atom classical molecular dynamics simulations for polymer informatics. npj Computational Materials **8**(1), 222 (2022)
12. Jiang, S., Dieng, A.B., Webb, M.A.: Property-guided generation of complex polymer topologies using variational autoencoders. npj Computational Materials **10**(1), 139 (Jun 2024). https://doi.org/10.1038/s41524-024-01328-0, https://doi.org/10.1038/s41524-024-01328-0
13. Kamal, D., Tran, H., Kim, C., Wang, Y., Chen, L., Cao, Y., Joseph, V.R., Ramprasad, R.: Novel high voltage polymer insulators using computational and data-driven techniques. The Journal of Chemical Physics **154**(17) (2021)
14. Kuenneth, C., Rajan, A.C., Tran, H., Chen, L., Kim, C., Ramprasad, R.: Polymer informatics with multi-task learning. Patterns **2**(4) (2021)
15. Kuenneth, C., Ramprasad, R.: polybert: a chemical language model to enable fully machine-driven ultrafast polymer informatics. Nature Communications **14**(1), 4099 (2023)
16. Kuenneth, C., Schertzer, W., Ramprasad, R.: Copolymer informatics with multitask deep neural networks. Macromolecules **54**(13), 5957–5961 (2021)
17. Landrum, G.: Rdkit: Open-source cheminformatics (2012), http://www.rdkit.org
18. Lin, X., Baweja, H., Kantor, G., Held, D.: Adaptive auxiliary task weighting for reinforcement learning. Advances in neural information processing systems **32** (2019)
19. Lo, S., Seifrid, M., Gaudin, T., Aspuru-Guzik, A.: Augmenting polymer datasets by iterative rearrangement. Journal of Chemical Information and Modeling **63**(14), 4266–4276 (2023)
20. Neres, L.C.S., Feliciano, G.T., Dutra, R.F., Sotomayor, M.D.P.T.: Development of a selective molecularly imprinted polymer for troponin t detection: a theoretical-experimental approach. Materials Today Communications **30**, 102996 (2022)
21. Odian, G.: Principles of Polymerization. Wiley India Pvt. Limited (2004)
22. Qiu, H., Liu, L., Qiu, X., Dai, X., Ji, X., Sun, Z.Y.: Polync: a natural and chemical language model for the prediction of unified polymer properties. Chemical Science **15**(2), 534–544 (2024)
23. Queen, O., McCarver, G.A., Thatigotla, S., Abolins, B.P., Brown, C.L., Maroulas, V., Vogiatzis, K.D.: Polymer graph neural networks for multitask property learning. npj Computational Materials **9**(1), 90 (2023)
24. Reis, M., Gusev, F., Taylor, N.G., Chung, S.H., Verber, M.D., Lee, Y.Z., Isayev, O., Leibfarth, F.A.: Machine-learning-guided discovery of 19f mri agents enabled by automated copolymer synthesis. Journal of the American Chemical Society **143**(42), 17677–17689 (2021)
25. Shukla, S.S., Kuenneth, C., Ramprasad, R.: Polymer informatics beyond homopolymers. MRS Bulletin **49**(1), 17–24 (2024)
26. St John, P.C., Phillips, C., Kemper, T.W., Wilson, A.N., Guan, Y., Crowley, M.F., Nimlos, M.R., Larsen, R.E.: Message-passing neural networks for high-throughput polymer screening. The Journal of chemical physics **150**(23) (2019)
27. Tiwari, S.P., Shi, W., Budhathoki, S., Baker, J., Sekizkardes, A.K., Zhu, L., Kusuma, V.A., Hopkinson, D.P., Steckel, J.A.: Creation of polymer datasets with targeted backbones for screening of high-performance membranes for gas separation. Journal of Chemical Information and Modeling **64**(3), 638–652 (2024)

28. Toland, A., Tran, H., Chen, L., Li, Y., Zhang, C., Gutekunst, W., Ramprasad, R.: Accelerated scheme to predict ring-opening polymerization enthalpy: Simulation-experimental data fusion and multitask machine learning. The Journal of Physical Chemistry A **127**(50), 10709–10716 (2023)
29. Tran, H., Toland, A., Stellmach, K., Paul, M.K., Gutekunst, W., Ramprasad, R.: Toward recyclable polymers: ring-opening polymerization enthalpy from first-principles. The Journal of Physical Chemistry Letters **13**(21), 4778–4785 (2022)
30. Wang, F., Guo, W., Cheng, M., Yuan, S., Xu, H., Gao, Z.: Mmpolymer: A multimodal multitask pretraining framework for polymer property prediction. In: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management. p. 2336–2346. CIKM '24, Association for Computing Machinery, New York, NY, USA (2024). https://doi.org/10.1145/3627673.3679684, https://doi.org/10.1145/3627673.3679684
31. Wilbraham, L., Sprick, R., Jelfs, K., Zwijnenburg, M.: Mapping binary copolymer property space with neural networks, chem (2019)
32. Xie, T., France-Lanord, A., Wang, Y., Lopez, J., Stolberg, M.A., Hill, M., Leverick, G.M., Gomez-Bombarelli, R., Johnson, J.A., Shao-Horn, Y., et al.: Accelerating amorphous polymer electrolyte screening by learning to reduce errors in molecular dynamics simulated properties. Nature communications **13**(1), 3415 (2022)
33. Xu, C., Wang, Y., Barati Farimani, A.: Transpolymer: a transformer-based language model for polymer property predictions. npj Computational Materials **9**(1), 64 (2023)
34. Xu, P., Chen, H., Li, M., Lu, W.: New opportunity: machine learning for polymer materials design and discovery. Advanced Theory and Simulations **5**(5), 2100565 (2022)
35. Yan, C., Feng, X., Li, G.: From drug molecules to thermoset shape memory polymers: A machine learning approach. ACS Applied Materials & Interfaces **13**(50), 60508–60521 (2021). https://doi.org/10.1021/acsami.1c20947
36. Yan, C., Li, G.: The rise of machine learning in polymer discovery. Advanced Intelligent Systems **5**(4), 2200243 (2023)
37. Zhao, Y., Mulder, R.J., Houshyar, S., Le, T.C.: A review on the application of molecular descriptors and machine learning in polymer design. Polymer Chemistry **14**(29), 3325–3346 (2023)
38. Zhou, J., Yang, Y., Mroz, A.M., Jelfs, K.E.: Polycl: contrastive learning for polymer representation learning via explicit and implicit augmentations. Digital Discovery (2025)
39. Zhu, G., Kim, C., Chandrasekarn, A., Everett, J.D., Ramprasad, R., Lively, R.P.: Polymer genome–based prediction of gas permeabilities in polymers. Journal of Polymer Engineering **40**(6), 451–457 (2020)