

# Transferring a molecular foundation model for polymer property predictions

Pei Zhang,<sup>\*,†</sup> Logan Kearney,<sup>‡</sup> Debsindhu Bhowmik,<sup>†</sup> Zachary Fox,<sup>†</sup>  
Amit K. Naskar,<sup>‡</sup> and John Gounley<sup>†</sup>

<sup>†</sup>*Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA*

<sup>‡</sup>*Chemical Sciences Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA*

E-mail: zhangp1@ornl.gov

Phone: +1 865 341 0368

## Abstract

Transformer-based large language models have remarkable potential to accelerate design optimization for applications such as drug development and materials discovery. Self-supervised pretraining of transformer models requires large-scale datasets, which are often sparsely populated in topical areas such as polymer science. State-of-the-art approaches for polymers conduct data augmentation to generate additional samples but unavoidably incurs extra computational costs. In contrast, large-scale open-source datasets are available for small molecules and provide a potential solution to data scarcity through transfer learning. In this work, we show that using transformers pretrained on small molecules and fine-tuned on polymer properties achieve comparable accuracy to those trained on augmented polymer datasets for a series of benchmark prediction tasks.

## Introduction

Polymer informatics is a rapidly growing field that leverages data-centric machine learning (ML) approaches to discover and design new polymers.<sup>1–3</sup> For efficient exploration of the vast chemical space, fast ML surrogates<sup>4–14</sup> are often developed for polymer structure-property relationships. Although these surrogates differ in structure representations (e.g., repeat units, polymer chains), feature representations (e.g., string representation, molecular graph, chemical descriptors), and ML algorithms (e.g., random forest, Gaussian process regression, deep neural network [DNN]), they focus on conventional small-scale ML models in the supervised learning paradigm and suffer from inadequately labeled (i.e., structure-property) data issues. Recently, transformer-based large language models (LLMs) TransPolymer<sup>15</sup> and polyBERT<sup>16</sup> have achieved state-of-the-art accuracy for prediction of polymer properties. However, training such LLMs requires data at scale, which is not readily publicly available for polymers. In this study, we evaluate whether the key limitation for developing language models for polymers—the lack of large, publicly available polymer datasets—can be mitigated with a foundation model approach.

Transformers, which were originally developed for natural language processing,<sup>17</sup> have been applied to guide design in multiple scientific applications (e.g., predicting function from protein sequences<sup>18–23</sup> and generating small molecules with desired properties<sup>24–28</sup>). Two factors have facilitated the development of LLMs for these applications. First, these domains have text-based sequence representations (e.g., FASTA<sup>29</sup> and the simplified molecular-input line-entry system [SMILES]) that are easily amenable to representation as natural language text. Second, these applications have massive, previously compiled datasets (e.g., Enamine REAL,<sup>30</sup> PubChem,<sup>31</sup> and ZINC<sup>32</sup> for molecules and Uniprot<sup>33</sup> for proteins) that have enabled the training of these LLMs.

The typical training process involves two stages. First, **pretraining** is primarily self-supervised on large-scale structure data (e.g., SMILES), and the model learns to encode the generic structure information. Second, **fine-tuning** is supervised on smaller task-specific

labeled data (i.e., properties), and the pretrained model learns to predict properties from structures. For example, when ChemBERTa-2<sup>28</sup> was pretrained on 77 million compounds with a SMILES representation from PubChem and fined-tuned on tasks (with 1,500~8,000 labeled samples) from MoleculeNet and when ProteinBERT<sup>23</sup> was pretrained on about 106 million proteins from Uniprot UniRef90 and fine-tuned on nine benchmarks with a training size varying from 2,700 to 53,700, they both achieved the best accuracy for most benchmark tasks. These advancements in LLMs for molecules and proteins are attributed to the high representation capacity of the transformer architecture and, equally important, the diverse large-scale datasets available for pretraining such models.

To improve learning for polymer-property prediction in the presence of limited fine-tuning data, the potential value of large-scale pretraining is clear. Open-source datasets for polymer properties from DFT simulations<sup>34</sup> are usually small with a few hundred or a few thousand samples but are nonetheless larger than datasets of experimental measurements. However, the data scarcity challenge for polymers extends to large-scale structure data for self-supervised pretraining as well. Large polymer databases such as PolyInfo<sup>35</sup> and Polymer Genome<sup>36</sup> do not provide access to curated downloadable datasets. On the other hand, the accessible PI1M<sup>37</sup> database does contain about 1 million polymer structures, but it has been compiled by using a generative AI approach based on a recurrent neural network trained on a much smaller dataset. Not only is PI1M still orders of magnitude smaller than datasets such as Enamine REAL, but the manner of its generation may limit the diverse representations necessary for language model pretraining. Still, TransPolymer<sup>15</sup> obtained state-of-the-art results for property prediction by leveraging pretraining on about 5 million noncanonical SMILES entries augmented from PI1M. Similarly, polyBERT<sup>16</sup> created its 100 million-sample pretraining dataset by fragment-based data augmentation on about 13,800 synthesized polymers.

In this work, we evaluate whether pretraining a transformer on a large-scale small molecule dataset such as Enamine REAL can lead to a model that is similarly effective

to TransPolymer or polyBERT. Our hypothesis is that the information being learned during pretraining, whether on datasets of polymers or small molecules, is primarily generic chemical language that serves as a suitable starting point for fine-tuning property prediction. To test this hypothesis, we pretrain BERT language models<sup>38</sup> on PI1M and Enamine REAL. Along with a randomly initialized model that serves as a control, we evaluate the models on downstream tasks for predicting polymer properties after supervised fine-tuning on an open-source DFT dataset.<sup>34</sup> This labeled DFT dataset is much smaller than typically desired for deep learning, and this should emphasize the role of pretraining. In similar fashion to Kuenneth et al.,<sup>16,34</sup> we employ a multitasking approach which predicts all properties simultaneously for better data efficiency. Our results show that the models pretrained on Enamine REAL achieve similar accuracy to the TransPolymer and polyBERT work, thereby suggesting that a model pretrained on a large-scale, accessible, and general molecular dataset could serve as a foundation model for applications across the molecular design landscape. Such an approach—pretraining on small molecular structures and fine-tuning for multitasking with multiple properties simultaneously (SML-MT)—would avoid the cost of generating and pretraining on large augmented polymer datasets. Moreover, the randomly initialized model learns quite effectively, and this suggests clear limits to the value of current pretraining procedures for molecular data.

## Core results

We trained transformers for polymer properties following the standard two-step procedure: self-supervised pretraining on a large-scale structure dataset and supervised fine-tuning on small structure-property datasets (i.e., the DFT dataset<sup>34</sup> with 8 properties in Table 1). The fine-tuned models are evaluated based on their accuracy when predicting properties for polymers unseen in training. The evaluation metrics are relative root mean square error (RMSE) and R<sup>2</sup> with a 5-fold cross-validation, which are similar to those used for TransPolymer<sup>15</sup>

and polyBERT<sup>16</sup>. The relative RMSE is defined as RMSE scaled by the difference between the maximum and minimum values of each property.

Table 1: Open-source DFT dataset with 8 properties<sup>34</sup>

<b>Property</b>	<b>Unit</b>	<b>Number of samples</b>	<b>Minimum</b>	<b>Maximum</b>
Atomization energy (Eat)	eV·atom <sup>-1</sup>	390	-7.0	-5.0
Crystallization tendency (Xc)	%	432	0.10	100.0
Bandgap (chain) (Egc)	eV	3,380	0.02	10.0
Bandgap (bulk) (Egb)	eV	561	0.4	10.0
Electron affinity (Eea)	eV	368	0.4	5.0
Ionization energy (Ei)	eV	370	4.0	10.0
Refractive index (nc)	1	382	1.0	3.0
Dielectric constant (eps)	1	382	3.0	9.0
In total		6,265		

Figure 1 shows a comparison of the prediction accuracy of our SML-MT model, which was pretrained on Enamine REAL and fine-tuned in the multitasking context, versus the accuracy of TransPolymer<sup>15</sup> and polyBERT.<sup>16</sup> The three surrogates are trained differently.

- TransPolymer<sup>15</sup> is based on RoBERTa and pretrained through masked language modeling (MLM) on approximately 5 million polymers augmented from PI1M.<sup>37</sup> It is finetuned on an augmented version of the DFT dataset shown in Table 1 (with 7 out of 8 properties [no Eat]), which is about  $2\times \sim 25\times$  as large as the original size for each property. Data augmentation for both pretraining and fine-tuning is conducted by removing the canonical constraint in the SMILES representation and subsequently generating multiple noncanonical SMILES for each polymer structure. Polymer sequences are composed of polymer SMILES (of repeating units) and chemical descriptors and tokenized with a chemically aware tokenizer. The SMILES are split using a regular expression (regex), and each descriptor is treated as one token.
- polyBERT<sup>16</sup> is based on DeBERTa and pretrained through MLM on approximately 80 million polymers generated from about 13,800 previously synthesized ones. In finetuning, the pretrained transformer is frozen and used to provide polymer fingerprint-

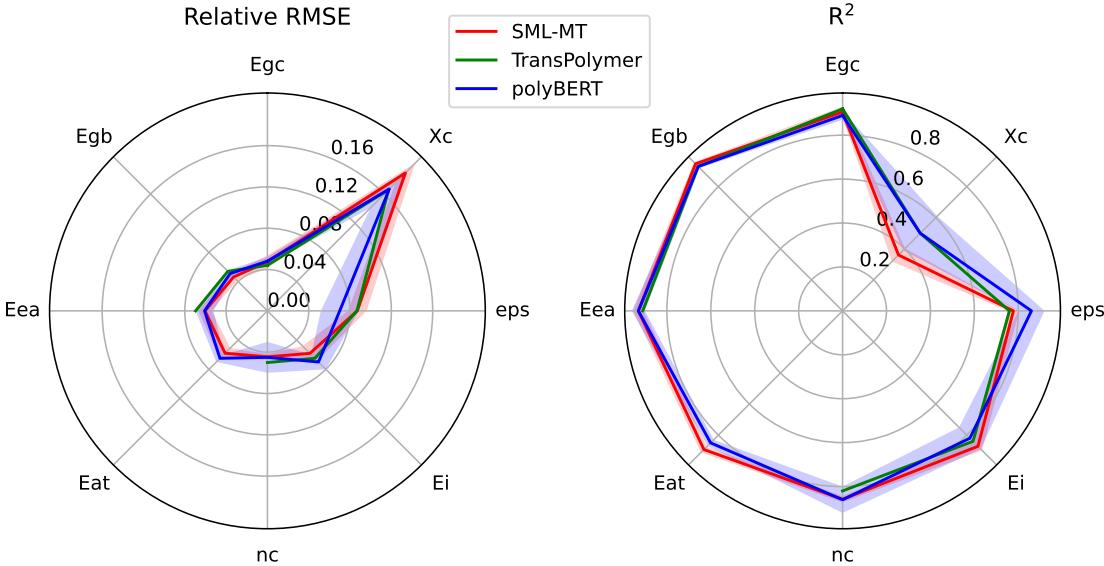


Figure 1: Prediction performance metrics (relative RMSE and  $R^2$ ) of SML-MT, TransPolymer, and polyBERT for 8 DFT properties. The lines represent the mean values and the shaded areas represent the standard deviation (STD) values of the test sets in the 5-fold cross-validation. Performance data for TransPolymer is reproduced from Xu et al.’s Table 4,<sup>15</sup> which has neither Eat nor STD values, and polyBERT is reproduced from Kuenneth et al.’s Fig. 5 and Supplementary Table S1.<sup>16</sup> SML-MT, pretrained on small molecules, achieves similar accuracy to TransPolymer and polyBERT, pretrained on augmented polymer datasets. In particular, of 8 DFT properties, SML-MT has the best accuracy for 5 (i.e., Egb, Eea, Eat, nc, and Ei), comparable accuracy for 2 (i.e., Egc and eps), and poor accuracy for 1 (i.e., Xc). The poor prediction performance of all three models on Xc (i.e., crystallization tendency) is expected. The crystallization process highly depends on polymer structure descriptions across multiple length scales, and the current SMILES representation for only repeat units is inadequate.

ing inputs to multitasking DNNs for property predictions. The DNNs are trained on a dataset with 29 properties from both experimental measurement and DFT (Table 1). These properties are split into six property categories—thermal, thermodynamic and physical, electronic, optical and dielectric, mechanical, and permeability—and one multitasking DNN is trained for each category. Polymer sequences are represented by polymer SMILES (i.e., SMILES for repeating units with “\*”/“[\*]” for polymerization points) and tokenized with a regex-based tokenizer.

- SML-MT is our model based on BERT.<sup>38</sup> It is pretrained through MLM on approximately 1 billion small molecules in Enamine REAL, similar to our previous work,<sup>25</sup>

and fine-tuned on the DFT dataset in Table 1. In fine-tuning, the weights of the entire model (i.e., the BERT encoder followed by a linear NN layer with 8 neurons [one for each property]) are updated. Polymer sequences are represented by polymer SMILES with “\*” for polymerization points and tokenized with a regex-based tokenizer.

Both the mean and standard deviation (STD) values of relative RMSE and R<sup>2</sup> for the test sets over the five folds in cross-validation are plotted in Figure 1. For the 8 DFT properties, SML-MT outperforms TransPolymer and polyBERT for 5 (i.e., Egb, Eea, Eat, nc, and Ei), demonstrates comparable accuracy for 2 (i.e., Egc and eps), and has poor accuracy for 1 (i.e., Xc). The poor prediction performance of all three models on Xc (i.e., crystallization tendency) is expected. Xc describes the tendency of a given polymer structure to form densified regions of ordered crystalline domains. It is highly dependent on polymer structure descriptions across multiple length scales, and the current SMILES representation for only repeat units is inadequate.

Our results show that a transformer pretrained on small molecules can achieve similar accuracy to the models trained on augmented polymer datasets. This suggests that a foundation model is a feasible approach to the data scarcity issue that plagues polymer informatics.

## Methods

### Transformers for structure-property relationships

We developed transformer-based models that take polymer sequence representations as input and then output property predictions. The polymer sequences are first split into tokens (a basic unit in LLMs) that can be processed by transformers via tokenization. Subsequently, the tokenized sequences are fed into transformers with multihead attention layers that learn the correlation features embedded in polymer structures. Finally, the feature embeddings are mapped to properties via a linear NN layer. The following subsections describe the model

and workflow details.

**Polymer representation** The polymer-SMILES (p-SMILES) from PI1M<sup>37</sup> is used to represent polymers. It consists of SMILES for repeating units with “\*” representing the polymerization points. More complicated representations that consider longer-chain structures with multiple repeating units are available, but their impact on prediction accuracy is unclear. For example, the single repeating unit has better performance than higher-degree polymerized representations for glass transition temperature predictions.<sup>4</sup> In our work, we choose p-SMILES for its simplicity, sufficiency in representing the degree-1 polymers in PI1M, and similarity to SMILES representation of small molecules.

**Tokenization** Tokenization is the process of splitting SMILES (or p-SMILES) into smaller pieces (i.e., tokens), which are subsequently converted to numerical representations (via a vocabulary list) that are interpretable to transformers. We employ two splitting rules—regex (i.e., based on regex for individual atoms) and BERT (i.e., the default punctuation-based WordPiece tokenizer from Hugging Face<sup>39</sup>)—following our previous work.<sup>25</sup> The observations are similar for both rules. We will present the results with regex in the paper and provide the comparison with BERT in the Supplementary Materials (Figures S.6, S.7, and S.8). For regex, the vocabulary list has 78 tokens consisting of 1–9, atoms, bonds, special tokens (i.e., [PAD], [UNK], [CLS], [SEP], and [MASK]), and “\*” for polymerization points.

**Model architecture** The transformer model used is BERT<sup>38</sup> with MLM. We employed the standard `BertForMaskedLM` implementation from Hugging Face with 12 hidden layers, 12 attention heads, and hidden size of 768. The pretraining is conducted with MLM (at a default 15% masking), in which the model inputs are the preprocessed tokenized sequences (i.e., with [MASK] for masked tokens, prepended with [CLS], and appended with [SEP]), and the model is trained to predict the masked tokens. In fine-tuning for polymer property predictions, a dropout layer with a probability of 0.1 and a linear NN layer is attached to

the [CLS] embedding output with the number of neurons equal to the number of properties.

**Transfer learning** Transfer learning broadly refers to applying trained ML models to new applications that differ from the training data. In this work, we are interested in transferring the BERT model pretrained on small molecules to polymers. Training BERT requires large-scale datasets, which are well-documented for small molecules but not readily publicly available for polymers. Fundamentally, polymers and small molecules share basic chemical structure (which are transferable), but polymers have more complicated long-chain structures. It is thus unclear how much of the knowledge that the pretrained model learns from small molecules is transferable to polymers. In this work, we evaluate such transferability. To the best of our knowledge, this is the first example of transfer learning from small molecules to polymers using a transformer architecture.

**Multitasking** Multitasking, in this case predicting multiple properties simultaneously in a single model, is a popular strategy to leverage the potential correlations among properties for better data efficiency in deep learning. It has been used to alleviate the data scarcity issue in polymers,<sup>16,34,40</sup> including in polyBERT, for which 6 multitasking NNs were designed—with one for each property category. In our work, we train one multitasking model simultaneously for all properties.

## Experiment settings

The transformers are trained following the standard pretraining and fine-tuning procedure.

### Pretraining

**Datasets** To evaluate the transferability of transformers from small molecules to polymers, we consider two datasets in pretraining: Enamine REAL for small molecules and PI1M for polymers. Enamine REAL is a popular compound library<sup>30</sup> for deep learning studies.<sup>25,26,41,42</sup> The dataset used here is taken from our previous work<sup>25</sup> and consists of

approximately 1 billion small molecules. PI1M consists of approximately 1 million degree-1 polymers generated from PolyInfo, which has also been used in multiple deep learning studies,<sup>12,15,43</sup> including for TransPolymer. To compare the molecular structures in two datasets, we calculate the extended-connectivity fingerprints (ECFPs) with 1,024 bits from SMILES using RDKit. Figure 2 shows the data density plot of the ECFPs projected on the first two principal components (PCs) in a principal component analysis for 50,000 randomly selected samples from the two datasets. The density plot shows that the two datasets have very different ECFP distributions and presents potential challenges in the transfer learning task.

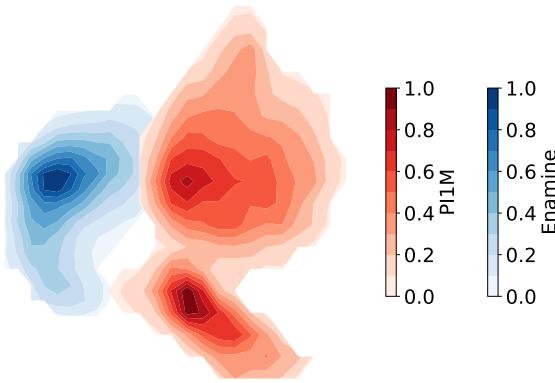


Figure 2: Density plot of extended-connectivity fingerprints projected on the first two principal components for structures in PI1M and Enamine REAL. It shows that the two datasets contain very different structures, thereby presenting potential challenges in the transfer learning task.

**Training on Summit** The two models pretrained on Enamine REAL and PI1M are referred to as SML (small) and PLM (polymer), respectively, as shown in Table 2. The model without pretraining (i.e., INIT [randomly initialized]) is included as a reference. We use the tokenizer and BERT with MLM implementations from Hugging Face<sup>39</sup> along with the DeepSpeed optimization library.<sup>44</sup> In training, the learning rate was set to  $10^{-3}$ . The WarmupDecayLR scheduler was set with a warmup ratio of 0.1 and the LAMB optimizer,<sup>45</sup> and the batch size was set to 64 per GPU. The models were trained for 20 epochs with

gradient accumulation every 3 steps with 16-bit mixed-precision training. All training runs were performed on Summit (with 6 NVIDIA V100 GPUs for each node) at the Oak Ridge Leadership Computing Facility. The SML training used 334 nodes and took 6.1 hours on about 1 billion SMILES strings. PLM was trained using 1 node for 2.6 hours on about 1 million SMILES strings.

Table 2: Pretrained models

Dataset	Small molecule (Enamine REAL)	PI1M	No pretraining
Model	SML	PLM	INIT

Table 3: Fine-tuned models

Model	SML	PLM	INIT
Single-tasking (ST)	SML-ST	PLM-ST	INIT-ST
Multitasking (MT)	SML-MT	PLM-MT	INIT-MT

## Fine-tuning

In fine-tuning, the three pretrained models, SML, PLM, and INIT (with one dropout layer and one linear NN layer attached in the end for property outputs), are trained on the polymer DFT dataset in Table 1. In total, six types of models are trained depending on the single-tasking and multitasking context summarized in Table 3; or equivalently,  $27 = 3 \times (8 + 1)$  models are trained, where 3 is for the 3 pretrained models, 8 is for the 8 single-tasking models with one for each property, and 1 is for the multitasking model for all the DFT properties.

For each regression task, the dataset in Table 1 was split into 5 folds for cross-validation; therefore, 5 models were trained. By default, the AdamW optimizer was used with a learning rate of  $10^{-5}$  and zero weight decay, the batch size was set to 16, and the training runs were performed on one GPU for 600 epochs with L1 loss function.

## Model evaluation

The models (Table 3) learn the mappings from sequence space to feature embedding space and subsequently to property space. In pretraining, the model identifies a mapping from structure space to embedding space that is close to the mapping (of interest for downstream tasks) by learning generic knowledge from diverse, large-scale structure data. In fine-tuning, the model adjusts the mapping and simultaneously learns the subsequent mapping to property space with a low-cost iterative training on structure-property data. The final model accuracy is jointly determined by knowledge learned in both stages. A good pretrained model can lead to a faster convergence to the final solutions in fine-tuning. To understand the feasibility of transferring transformers pretrained on small molecules to polymers, we will evaluate not only the final model accuracy but also the quality of pretraining.

For the prediction accuracy, we will calculate both the  $R^2$  and relative RMSE with STD from the 5-fold cross-validation and compare them with the state-of-the-art values from TransPolymer<sup>15</sup> and polyBERT.<sup>16</sup> For the pretraining quality, we will evaluate the pretrained models in (1) the MLM tasks for prediction accuracy of the masked tokens in test sets from Enamine REAL and PI1M and (2) the fine-tuning tasks (i.e., prediction of DFT properties) for the convergent behaviors by varying the training set size from zero-short learning (i.e., with no fine-tuning) to full-scale training with all training samples.

## Results and discussions

### Prediction accuracy of fine-tuned models on DFT properties

We evaluate the predictive accuracy of the models for polymer properties pretrained on Enamine REAL (SML) and PI1M (PLM) and fine-tuned for individual property (single-tasking [ST]) and all properties simultaneously (multitasking [MT]). These models are compared with the two state-of-the-art models: TransPolymer<sup>15</sup> and polyBERT.<sup>16</sup> Figure 1

shows the results of SML-MT, which achieves accuracy comparable with both TransPolymer and polyBERT. In the following, we present the evaluation results of all models in Table 3. Both the relative RMSE and the  $R^2$  from a 5-fold cross-validation are calculated. We present relative RMSE only, and the corresponding  $R^2$  results are available in the Supplementary Material.

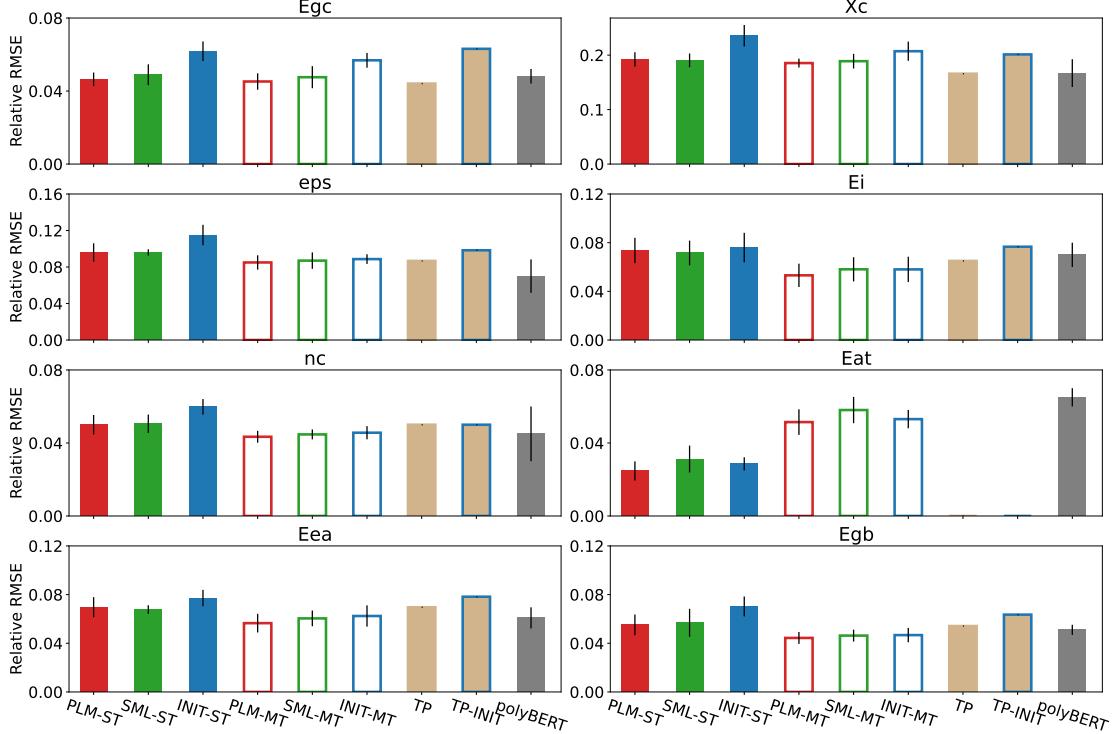


Figure 3: Relative RMSE prediction error of DFT properties. Error bars show the STD from the 5-fold cross-validation. Results of TransPolymer<sup>15</sup> (with no STD reported and hence no error bars plotted), where TP and TP-INIT represent the models with and without pretraining, respectively, and polyBERT<sup>16</sup> are also shown for reference. Pretraining increases model prediction accuracy for downstream tasks. The best pretrained models achieve accuracy comparable to or better than TransPolymer and polyBERT. Multitasking (MT, empty bars) shows an obvious accuracy improvement for eps, Ei, nc, Eea, and Egb; comparable accuracy for Egc and Xc; and degraded accuracy for Eat versus single-tasking (ST, solid bars).

Figure 3 shows the bar plot of the mean (over 5 folds) relative RMSE values for the test sets along with the TransPolymer (TP), TransPolymer without pretraining (TP-INIT), and polyBERT results reproduced from the literature.<sup>15,16</sup> The error bars represent the STD over 5 runs. The models without pretraining (i.e., INIT and bars with blue edge color) in general

show higher predictive errors than the models pretrained on Enamine REAL and PI1M. This is consistent in both ST (solid bars) and MT (empty bars). The best models from ST and MT obtain accuracy comparable to or better than TransPolymer and polyBERT. The models pretrained on Enamine REAL (i.e., SML and green bars) have similar predictive errors to the models pretrained on PI1M (i.e., PLM and red bars) in both ST and MT. This, together with Figure 1, supports the idea that BERT<sup>38</sup> trained on small molecules works as a foundation model for polymers. The numerical values for the bar plots are available in Table S.1 in the Supplementary Material, where the results of the training sets are also presented; the two best models for each property are highlighted in bold.

For the comparison between ST (solid bars) and MT (empty bars), the detailed performance results are property-dependent, although MT generally shows better overall performance than ST. In particular, MT achieves an obvious accuracy improvement for  $\text{eps}$ ,  $\text{Ei}$ ,  $\text{nc}$ ,  $\text{Eea}$ , and  $\text{Egb}$ ; comparable accuracy for  $\text{Egc}$  and  $\text{Xc}$ ; and degraded accuracy for  $\text{Eat}$  versus ST. This is consistent with the observations reported in the polymer property prediction work<sup>34</sup> with chemistry-informed Polymer Genome fingerprint embeddings (that considers hierarchical length scales), where ST performed slightly better than MT on thermodynamic and physical properties (i.e.,  $\text{Eat}$  and  $\text{Xc}$ ). The different performance of ST and MT across properties is a combination of available training data and the potential physical correlations between them. In general, the different tasks in MT interact in a collaborative and competing manner—highly correlated (uncorrelated) ones assist (compete with) each other. For the polymer training data in Table 1, strong overlapping across properties exists. That is, given a polymer SMILES, there can be multiple properties. Table 4 shows such overlapping. For the 390 polymers in  $\text{Eat}$  (a thermodynamic and physical property showing decreased accuracy from ST to MT), more than 91% share the same structures with 6 properties from the other two distinct categories. The learning task is expected to be more difficult in MT, and, consequently, ST is preferred for better accuracy.

Table 4: Number of overlapping samples across DFT properties in Table 1

Property	Eat	Xc	Egc	Egb	Eea	Ei	nc	eps
Eat (thermodynamic and physical)	390							
Xc (thermodynamic and physical)	15	432						
Egc (electronic)	365	191	3380					
Egb (electronic)	358	31	543	561				
Eea (electronic)	366	14	351	343	368			
Ei (electronic)	368	15	353	345	368	370		
nc (optical and dielectric)	380	15	360	353	361	363	382	
eps (optical and dielectric)	380	15	360	353	361	363	382	382

## Quality of pretraining

### Prediction accuracy of pretrained models on masked tokens in MLM

In pretraining, the PLM and SML models are trained with the standard MLM strategy with a 15% masking rate on PI1M (PLM) and Enamine REAL (SML). Two corresponding test sets that consist of 9,999 SMILES strings unseen during training are used to evaluate model accuracy in predicting masked tokens.

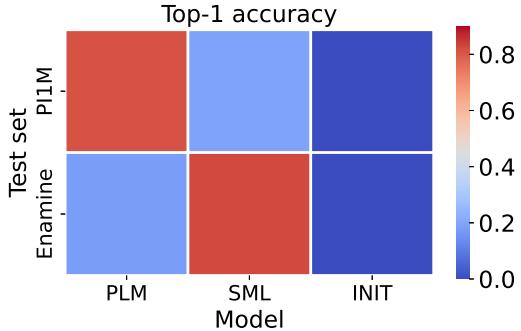


Figure 4: Top-1 accuracy of token predictions in MLM of models PLM, SML, and INIT for PI1M and Enamine test sets with 9,999 SMILES.

Figure 4 shows the top-1 accuracy of PLM and SML together with INIT (i.e., without pretraining) on the two test sets. The pretrained PLM and SML models perform well in the in-distribution test sets PI1M and Enamine, respectively, with top-1 test accuracy higher than 81%. Without pretraining, the INIT model fails to predict any SMILES correctly. In contrast, the cross-testing accuracy (i.e., PLM tested on Enamine and SML tested on PI1M)

is about 20%, which indicates the pretrained models PLM and SML learned transferable structure information between polymers and small molecules.

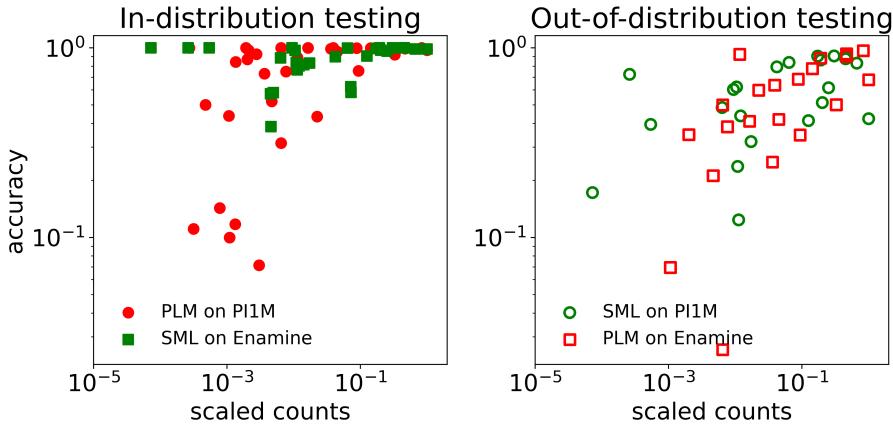


Figure 5: Prediction accuracy of masked tokens in the test sets against the scaled frequency in the training sets. The tokens with higher frequencies in the training sets are generally better captured.

We examine the prediction accuracy of individual tokens in MLM. Figure 5 shows the prediction accuracy of masked tokens from the test sets versus the corresponding scaled occurrence frequency in the training sets. In general, the ones with higher frequencies in training sets are better captured for the in-distribution testing (i.e., models are trained and tested on similar structures). Even for the (out-of-distribution) cross-testing, in which the models are trained and tested on very different structures (Figure 2), a more scattered but still evident correlation can be seen. For  $C$ , which is the most frequent token (with the scaled counts of 1.0) in the training sets, the models achieve decent accuracy of around  $0.4 \sim 0.7$  in the out-of-distribution testing. This again indicates generic transferable knowledge learned by the models and hence supports the foundation model approach. In the Supplementary Materials, we show the top-20 most and least frequent tokens (Figure S.3), the top-20 most and least accurately predicted tokens (Figure S.4), and a few examples of successfully and falsely predicted SMILES strings (Figure S.5) from the out-of-distribution testing.

## Learning efficiency of pretrained models in fine-tuning

The prediction accuracy shown in Figure 3 is determined by both pretraining and fine-tuning—pretraining aims to set up a good initial guess of model weights by learning generic knowledge from a large-scale structure dataset, and fine-tuning further refines them for specific properties with structure-property data. However, only looking at the final accuracy to evaluate the quality of pretraining can be misleading; recent examples have shown that higher pretraining accuracy can saturate or decrease fine-tuning accuracy.<sup>46</sup> Different pretrained models may show equally good prediction accuracy when the structure-property data is adequate in both quantity and quality; on the other hand, all models would show equally poor accuracy when there is no structure-property relationship. What sets a pretrained model apart is learning efficiency: the amount of structure-property data (which is expensive to collect) required to achieve the same prediction accuracy.

Figure 6 shows the learning efficiency of three pretrained models (SML, PLM, and INIT) in both ST and MT. In total, six sets of fine-tuning runs were conducted with the training set sizes at  $[0, 1/16, 1/8, 1/4, 1/2, 1]$  of the full-scale size. The STD from the 5-fold cross-validation is shown as error bars. Results of TransPolymer<sup>15</sup> and polyBERT<sup>16</sup> are also shown for reference, where the brown lines show the mean relative RMSE for TransPolymer, and the gray shadow areas show the mean  $\pm 1$  STD of relative RMSE from the 5-fold cross-validation for polyBERT. For zero-shot learning at a data proportion of 0.0, all the models perform equally poorly because they are agnostic of property outputs with the weights of final NN layer initialized randomly following a normal distribution. With the training size increasing (i.e., the models being exposed to more structure-property examples), some pretrained models start to stand out by showing higher learning efficiencies with lower relative RMSE errors, such as the SML and PLM shown in green and red (in contrast to INIT shown in blue) for Egc. Moreover, although the relative RMSEs of the SML models are marginally higher than for PLM, they are still comparable. This further supports the transfer learning idea from small molecules to polymers, especially when considering the substantial cost saved

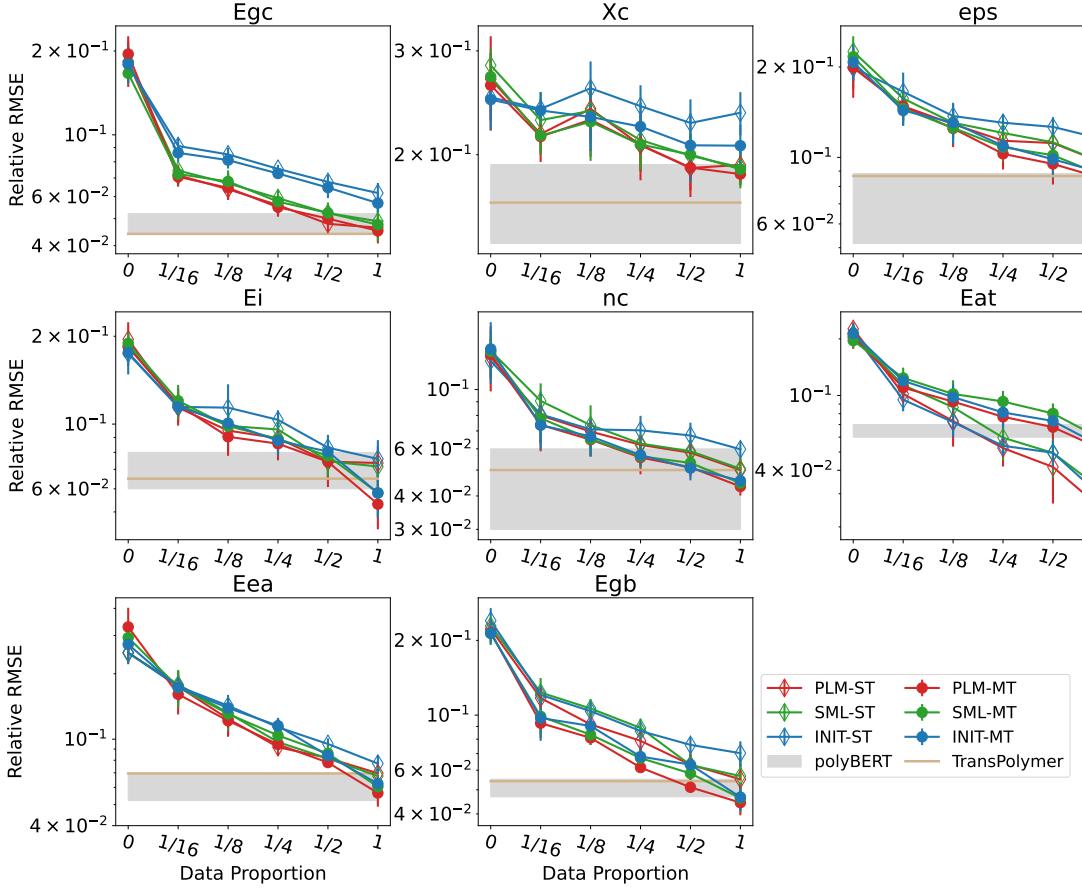


Figure 6: Relative RMSE prediction error of DFT properties at varying training set sizes ([0, 1/16, 1/8, 1/4, 1/2, 1] of the full-scale size) in fine-tuning. Results of TransPolymer<sup>15</sup> (with brown lines for the mean relative RMSE) and polyBERT<sup>16</sup> (with gray shadow areas for the mean  $\pm 1$  STD of relative RMSE from the 5-fold cross-validation) are also shown for reference. Figure 3 corresponds to results for a data proportion of 1.0. The two pretrained models PLM (in red) and SML (in green) show better learning efficiency than the INIT model without pretraining (in blue) and hence require less expensive structure-property data for the same accuracy. Multitasking (MT, solid circles) improves prediction accuracy for most variables versus single-tasking (ST, empty diamonds), especially for models without pretraining (INIT in blue).

in data augmentation and pretraining.

Similar trends hold for the other properties, although they may not be as evident as for Egc. For example, in Xc, the difference between the two pretrained SML and PLM models and INIT increases from 1/8 to 1 but is associated with large overlapping error bars. The different behaviors across properties could be caused by data size. Notably, Egc has several times more samples than the other properties listed in Table 1.

Regarding the learning efficiency for ST (empty symbols) and MT (solid symbols), similar observations can be made of the data plotted in Figure 3: compared to ST, MT shows an obvious improvement for  $\text{eps}$ ,  $\text{Ei}$ ,  $\text{nc}$ ,  $\text{Eea}$ , and  $\text{Egb}$ ; is comparable for  $\text{Egc}$  and  $\text{Xc}$ ; and degrades for  $\text{Eat}$ . The different performance across properties is caused by the training data available for each property and the correlations with others in the collaborative-competitive MT, as detailed in the previous section (Figure 3). Moreover, the improvement of MT is more prominent in INIT (with a larger difference observed between solid blue circles and empty blue diamonds). For some properties (e.g.,  $\text{eps}$  and  $\text{nc}$ ), INIT-MT even achieves accuracy comparable to the pretrained models. This suggests that both MT and pretraining are promising techniques to improve prediction accuracy.

## Conclusion and future work

To summarize, we proposed a foundation model approach to address the data scarcity issue in polymer informatics. In particular, we pretrained a transformer-based model on the Enamine REAL large-scale small molecule dataset, and showed that the model can achieve accuracy comparable with the TransPolymer and polyBERT models without the need for expensive generation and pretraining on large-scale augmented polymer datasets. This further supports the potential of using a model pretrained on a large-scale, accessible, and general molecular dataset as a foundation model for multiple applications across the molecular design landscape.

We compared the models pretrained on Enamine REAL and PI1M with models without pretraining in ST and MT contexts for prediction accuracy on polymer DFT properties and pretraining quality. The models pretrained on the two datasets have similar predictive behaviors: (1) they showed comparable prediction accuracy on polymer properties in ST and MT (consistently higher than models without pretraining); (2) they both achieved about 20% prediction accuracy on masked tokens in pretraining with MLM in cross-testing (in contrast

to the 0% accuracy without pretraining ); and (3) they exhibited similar learning efficiency in the convergence test on the training set size in fine-tuning. All these empirical observations support the feasibility of transfer learning from a model pretrained on small molecules to polymers. Moreover, MT enhances the prediction accuracy in general, even though the detailed performance is property-dependent, as a result of its collaborative-competitive nature.

Although our foundation model approach achieved promising results by accurately predicting local DFT properties that explicitly depend on bond connectivity of the repeat units, we do not expect this to be directly applicable to all polymeric properties, especially those that invoke longer-range dependencies. This is reflected in the models' poor prediction performance for  $X_c$ , which describes the tendency of a given polymer structure to form densified regions of ordered crystalline domains. Excluding that  $X_c$  is highly dependent on the thermomechanical history of the polymer, it is highly mediated by an ensemble of properties that require descriptions at multiple length scales (e.g., polymer chain flexibility, stereoregularity, and the intensity of repeat unit intermolecular interactions). These properties are not adequately encoded within a SMILES string but categorically mediate the dynamics of the macromolecules preceding solidification. This issue is common to all language models (including ours, TransPolymer, and polyBERT) that employ SMILES representations for polymers. To improve the prediction of these properties (e.g.,  $X_c$ , glass transition temperature), a future research direction would be to incorporate additional chemistry-informed layers of increasingly longer-range descriptors (e.g., degree of polymerization, connectivity) that pertain to topology in polymer representations.

## Data and software availability

All datasets used to train and test the models are open source. Enamine REAL<sup>30</sup> can be found at <https://enamine.net/compound-collections/real-compounds/real-database/>. PI1M<sup>37</sup> was downloaded from <https://github.com/RUIMINMA1996/PI1M/>, and the DFT property

dataset<sup>34</sup> was downloaded from <https://khazana.gatech.edu/>.

As for the codes, we used the `BertForMaskedLM` Hugging Face implementation of the BERT model with the MLM. The pretrained and fine-tuned models are available at Hugging Face <https://huggingface.co/mossaic-candle/>.

## Acknowledgement

The research described here was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the US Department of Energy Office of Science and the National Nuclear Security Administration. This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC0500OR22725. LK and AKN acknowledge support from US Department of Energy, Office of Science, Basic Energy Sciences under contract ERKCK60.

This manuscript has been authored by UT-Battelle LLC under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (<https://www.energy.gov/doe-public-access-plan>).

## Supporting Information Available

The following files are available free of charge.

- SM\_1.pdf: More prediction results of fine-tuned models in polymer properties, including bar plots of  $R^2$  in Figure S.1, tables of relative RMSE and  $R^2$  in Tables S.1 and S.2, and parity plots of the training and test sets from the 5-fold cross-validation in

Figure S.2.

- SM\_2.pdf: More results of pretrained models in MLM, including the top-20 most and least frequent tokens in Figure S.3, top-20 most and least accurately predicted tokens in Figure S.4, and a few examples of successfully and falsely predicted SMILES strings in Figure S.5 from the out-of-distribution testing.
- SM\_3.pdf: Prediction results of models with BERT tokenization method, including bar plots of relative RMSE in Figure S.6,  $R^2$  in Figure S.7, and prediction performance comparison of models with BERT and regex tokenizers in Figure S.8.

## References

- (1) Audus, D. J.; de Pablo, J. J. Polymer informatics: Opportunities and challenges. *ACS macro letters* **2017**, *6*, 1078–1082.
- (2) Cencer, M. M.; Moore, J. S.; Assary, R. S. Machine learning for polymeric materials: an introduction. *Polymer International* **2022**, *71*, 537–542, Publisher: John Wiley and Sons Ltd.
- (3) Chen, L.; Pilania, G.; Batra, R.; Huan, T. D.; Kim, C.; Kuenneth, C.; Ramprasad, R. Polymer informatics: Current status and critical next steps. *Materials Science and Engineering: R: Reports* **2021**, *144*, 100595.
- (4) Tao, L.; Varshney, V.; Li, Y. Benchmarking Machine Learning Models for Polymer Informatics: An Example of Glass Transition Temperature. *Cite This: J. Chem. Inf. Model* **2021**, *61*, 5413.
- (5) Jablonka, K. M.; Jothiappan, G. M.; Wang, S.; Smit, B.; Yoo, B. Bias free multiobjective active learning for materials design and discovery. *Nature Communications* **2021**, *12*, 1–10, Publisher: Springer US.

- (6) Kuenneth, C.; Lalonde, J.; Marrone, B. L.; Iverson, C. N.; Ramprasad, R.; Pilania, G. Bioplastic design using multitask deep neural networks. *Communications Materials* **2022**, *3*, 96.
- (7) Nagasawa, S.; Al-Naamani, E.; Saeki, A. Computer-Aided Screening of Conjugated Polymers for Organic Solar Cell: Classification by Random Forest. *The Journal of Physical Chemistry Letters* **2018**, *9*, 2639–2646, Publisher: American Chemical Society.
- (8) Kuenneth, C.; Schertzer, W.; Ramprasad, R. Copolymer Informatics with Multitask Deep Neural Networks. *Macromolecules* **2021**, *54*, 5957–5961, arXiv: 2103.14174.
- (9) Sattari, K.; Xie, Y.; Lin, J. Data-driven algorithms for inverse design of polymers. *Soft Matter* **2021**, *17*, 7607–7622, Publisher: Royal Society of Chemistry.
- (10) Ma, R.; Liu, Z.; Zhang, Q.; Liu, Z.; Luo, T. Evaluating Polymer Representations via Quantifying Structure–Property Relationships. *Journal of Chemical Information and Modeling* **2019**, *59*, 3110–3119, Publisher: American Chemical Society.
- (11) Jha, A.; Chandrasekaran, A.; Kim, C.; Ramprasad, R. Impact of dataset uncertainties on machine learning model predictions: the example of polymer glass transition temperatures. *Modelling and Simulation in Materials Science and Engineering* **2019**, *27*, 024002, Publisher: IOP Publishing.
- (12) Tao, L.; Chen, G.; Li, Y. Machine learning discovery of high-temperature polymers. *Patterns* **2021**, *2*, 100225, Publisher: Elsevier Inc.
- (13) Doan Tran, H.; Kim, C.; Chen, L.; Chandrasekaran, A.; Batra, R.; Venkatram, S.; Kamal, D.; Lightstone, J. P.; Gurnani, R.; Shetty, P.; Ramprasad, M.; Laws, J.; Shelton, M.; Ramprasad AFFILIATIONS, R. Machine-learning predictions of polymer properties with Polymer Genome. *J. Appl. Phys* **2020**, *128*, 171104.

- (14) 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. *Journal of Chemical Physics* **2021**, *154*, Publisher: AIP Publishing, LLC.
- (15) Xu, C.; Wang, Y.; Barati Farimani, A. TransPolymer: a Transformer-based language model for polymer property predictions. *npj Computational Materials* **2023**, *9*, 64.
- (16) Kuenneth, C.; Ramprasad, R. polyBERT: a chemical language model to enable fully machine-driven ultrafast polymer informatics. *Nature Communications* **2023**, *14*, 4099, Number: 1 Publisher: Nature Publishing Group.
- (17) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; Polosukhin, I. Attention is All you Need. Advances in Neural Information Processing Systems. 2017.
- (18) Kotsiliti, E. De novo protein design with a language model. *Nature Biotechnology* **2022**, *40*, 1433–1433.
- (19) Meier, J.; Rao, R.; Verkuil, R.; Liu, J.; Sercu, T.; Rives, A. Language models enable zero-shot prediction of the effects of mutations on protein function. *Advances in Neural Information Processing Systems* **2021**, *34*, 29287–29303.
- (20) Madani, A.; Krause, B.; Greene, E. R.; Subramanian, S.; Mohr, B. P.; Holton, J. M.; Olmos Jr, J. L.; Xiong, C.; Sun, Z. Z.; Socher, R., et al. Large language models generate functional protein sequences across diverse families. *Nature Biotechnology* **2023**, 1–8.
- (21) Ferruz, N.; Schmidt, S.; Höcker, B. ProtGPT2 is a deep unsupervised language model for protein design. *Nature communications* **2022**, *13*, 4348.
- (22) Ferruz, N.; Höcker, B. Controllable protein design with language models. *Nature Machine Intelligence* **2022**, *4*, 521–532.

- (23) Brandes, N.; Ofer, D.; Peleg, Y.; Rappoport, N.; Linial, M. ProteinBERT: a universal deep-learning model of protein sequence and function. *Bioinformatics* **2022**, *38*, 2102–2110.
- (24) Chithrananda, S.; Grand, G.; Ramsundar, B. ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction. **2020**,
- (25) Blanchard, A. E.; Chandra Shekar, M.; Gao, S.; Gounley, J.; Lyngaa, I.; Glaser, J.; Bhowmik, D. Automating Genetic Algorithm Mutations for Molecules Using a Masked Language Model. *IEEE Transactions on Evolutionary Computation* **2022**,
- (26) Blanchard, A. E.; Bhowmik, D.; Fox, Z.; Gounley, J.; Glaser, J.; Akpa, B. S.; Irle, S. Adaptive language model training for molecular design. *Journal of Cheminformatics* **2023**, *15*, 1–12.
- (27) Ross, J.; Belgodere, B.; Chenthamarakshan, V.; Padhi, I.; Mroueh, Y.; Das, P. Large-scale chemical language representations capture molecular structure and properties. *Nature Machine Intelligence* **2022**, *4*, 1256–1264.
- (28) Ahmad, W.; Simon, E.; Chithrananda, S.; Grand, G.; Ramsundar, B. Chemberta-2: Towards chemical foundation models. *arXiv preprint arXiv:2209.01712* **2022**,
- (29) Lipman, D. J.; Pearson, W. R. Rapid and sensitive protein similarity searches. *Science* **1985**, *227*, 1435–1441.
- (30) Shivanyuk, A.; Ryabukhin, S.; Tolmachev, A.; Bogolyubsky, A.; Mykytenko, D.; Chupryna, A.; Heilman, W.; Kostyuk, A. Enamine real database: Making chemical diversity real. *Chemistry today* **2007**, *25*, 58–59.
- (31) Kim, S.; Chen, J.; Cheng, T.; Gindulyte, A.; He, J.; He, S.; Li, Q.; Shoemaker, B. A.; Thiessen, P. A.; Yu, B., et al. PubChem in 2021: new data content and improved web interfaces. *Nucleic acids research* **2021**, *49*, D1388–D1395.

- (32) Irwin, J. J.; Shoichet, B. K. ZINC- a free database of commercially available compounds for virtual screening. *Journal of chemical information and modeling* **2005**, *45*, 177–182.
- (33) Consortium, U. UniProt: a worldwide hub of protein knowledge. *Nucleic acids research* **2019**, *47*, D506–D515.
- (34) Kuenneth, C.; Rajan, A. C.; Tran, H.; Chen, L.; Kim, C.; Ramprasad, R. Polymer informatics with multi-task learning. *Patterns* **2021**, *2*, 100238.
- (35) Otsuka, S.; Kuwajima, I.; Hosoya, J.; Xu, Y.; Yamazaki, M. PoLyInfo: Polymer database for polymeric materials design. 2011 International Conference on Emerging Intelligent Data and Web Technologies. 2011; pp 22–29.
- (36) Kim, C.; Chandrasekaran, A.; Huan, T. D.; Das, D.; Ramprasad, R. Polymer genome: a data-powered polymer informatics platform for property predictions. *The Journal of Physical Chemistry C* **2018**, *122*, 17575–17585.
- (37) Ma, R.; Luo, T. PI1M: A Benchmark Database for Polymer Informatics. *Cite This: J. Chem. Inf. Model* **2020**, *2020*, 4684–4690.
- (38) Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* **2018**,
- (39) Wolf, T. et al. Transformers: State-of-the-Art Natural Language Processing. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Online, 2020; pp 38–45.
- (40) Gurnani, R.; Kuenneth, C.; Toland, A.; Ramprasad, R. Polymer Informatics at Scale with Multitask Graph Neural Networks. *Chemistry of Materials* **2023**, *35*, 1560–1567.
- (41) Acharya, A.; Agarwal, R.; Baker, M. B.; Baudry, J.; Bhowmik, D.; Boehm, S.; Byler, K. G.; Chen, S.; Coates, L.; Cooper, C. J., et al. Supercomputer-based ensemble

- docking drug discovery pipeline with application to COVID-19. *Journal of chemical information and modeling* **2020**, *60*, 5832–5852.
- (42) Blanchard, A. E.; Gounley, J.; Bhowmik, D.; Shekar, M. C.; Lyngaaas, I.; Gao, S.; Yin, J.; Tsaris, A.; Wang, F.; Glaser, J. Language models for the prediction of SARS-CoV-2 inhibitors. *The International Journal of High Performance Computing Applications* **2022**, *36*, 587–602.
- (43) Yang, J.; Tao, L.; He, J.; McCutcheon, J. R.; Li, Y. Machine learning enables interpretable discovery of innovative polymers for gas separation membranes. *Science Advances* **2022**, *8*, eabn9545.
- (44) Rajbhandari, S.; Rasley, J.; Ruwase, O.; He, Y. ZeRO: Memory Optimizations toward Training Trillion Parameter Models. Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. 2020.
- (45) You, Y.; Li, J.; Reddi, S.; Hseu, J.; Kumar, S.; Bhojanapalli, S.; Song, X.; Demmel, J.; Keutzer, K.; Hsieh, C.-J. Large batch optimization for deep learning: Training bert in 76 minutes. *arXiv preprint arXiv:1904.00962* **2019**,
- (46) Abnar, S.; Dehghani, M.; Neyshabur, B.; Sedghi, H. Exploring the limits of large scale pre-training. *arXiv preprint arXiv:2110.02095* **2021**,

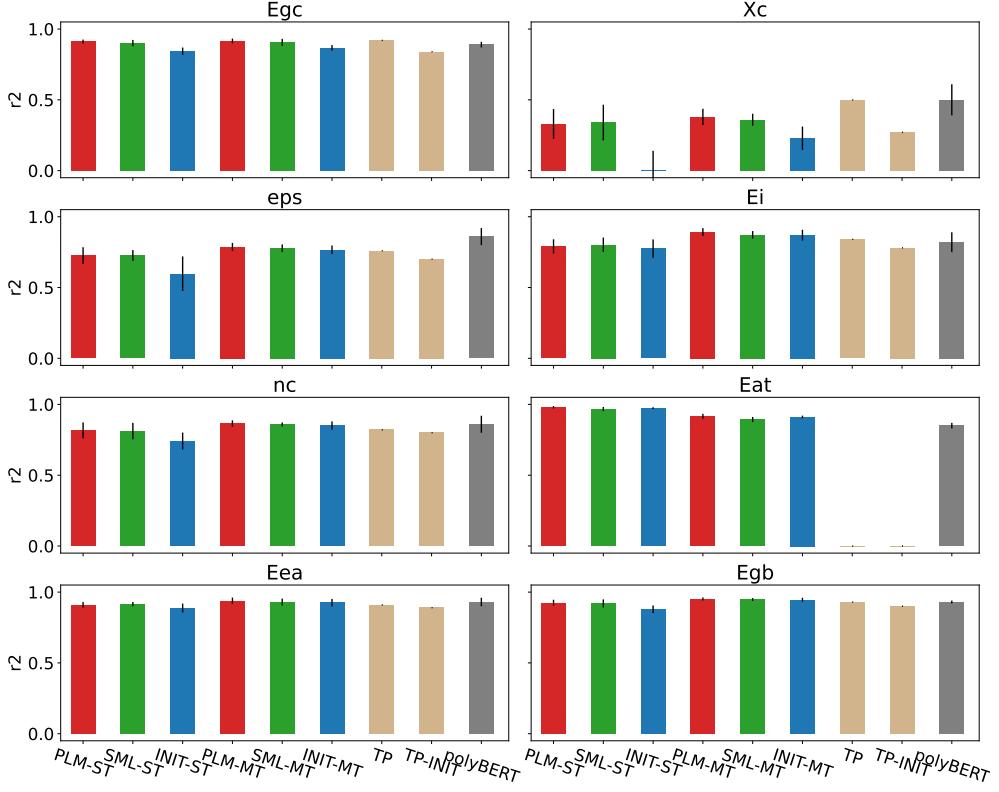


Figure S.1:  $R^2$  prediction accuracy of DFT properties. Error bars show the STD from the 5-fold cross-validation. Results of TransPolymer<sup>15</sup> (no STD reported, hence no error bars plotted), where TP and TP-INIT represent the models with and without pre-training, respectively, and polyBERT<sup>16</sup> are also shown for reference.

Table S.1: Prediction error: mean relative RMSE together with the STD from the 5-fold cross-validation of fine-tuned models on DFT datasets

Model		Egc	Xc	eps	Ei	nc	Eat	Eea	Egb
PLM-ST	train	0.010± 0.0010	0.024± 0.0040	0.017± 0.0028	0.015± 0.0018	0.010± 0.0013	0.010± 0.0021	0.018± 0.0020	0.015± 0.0014
	test	0.046± 0.0037	0.192± 0.0131	0.096± 0.0100	0.074± 0.0104	0.050± 0.0053	<b>0.025± 0.0053</b>	0.070± 0.0082	0.055± 0.0085
SML-ST	train	0.011± 0.0016	0.024± 0.0043	0.019± 0.0024	0.017± 0.0011	0.011± 0.0013	0.014± 0.0019	0.022± 0.0037	0.017± 0.0018
	test	0.049± 0.0057	0.190± 0.0126	0.096± 0.0034	0.072± 0.0101	0.051± 0.0050	0.031± 0.0073	0.068± 0.0035	0.057± 0.0115
INIT-ST	train	0.011± 0.0005	0.025± 0.0013	0.017± 0.0019	0.015± 0.0038	0.011± 0.0025	0.010± 0.0025	0.021± 0.0051	0.017± 0.0037
	test	0.062± 0.0054	0.236± 0.0197	0.115± 0.0112	0.076± 0.0121	0.060± 0.0043	<b>0.029± 0.0036</b>	0.077± 0.0067	0.070± 0.0082
PLM-MT	train	0.011± 0.0012	0.024± 0.0019	0.017± 0.0018	0.015± 0.0015	0.009± 0.0011	0.014± 0.0013	0.020± 0.0010	0.014± 0.0015
	test	<b>0.045± 0.0045</b>	0.185± 0.0082	<b>0.085± 0.0079</b>	<b>0.053± 0.0096</b>	<b>0.043± 0.0032</b>	0.051± 0.0070	<b>0.057± 0.0076</b>	<b>0.044± 0.0049</b>
SML-MT	train	0.013± 0.0007	0.027± 0.0017	0.019± 0.0032	0.016± 0.0012	0.012± 0.0016	0.018± 0.0010	0.025± 0.0020	0.017± 0.0014
	test	0.048± 0.0060	0.189± 0.0134	0.087± 0.0090	<b>0.058± 0.0100</b>	<b>0.045± 0.0027</b>	0.058± 0.0072	<b>0.060± 0.0065</b>	<b>0.046± 0.0049</b>
INIT-MT	train	0.010± 0.0008	0.021± 0.0034	0.013± 0.0014	0.012± 0.0014	0.009± 0.0007	0.014± 0.0022	0.017± 0.0026	0.013± 0.0023
	test	0.057± 0.0041	0.207± 0.0176	0.089± 0.0053	<b>0.058± 0.0104</b>	0.046± 0.0036	0.053± 0.0050	0.062± 0.0087	0.047± 0.0059
TransPolymer	test	<b>0.044</b>	<b>0.166</b>	0.087	0.065	0.050	-	0.070	0.054
TransPolymer-INIT	test	0.063	0.201	0.098	0.077	0.050	-	0.078	0.064
polyBERT	test	0.048± 0.0040	<b>0.167± 0.0255</b>	<b>0.070± 0.0183</b>	0.070± 0.0100	<b>0.045± 0.0150</b>	0.065± 0.0050	0.061± 0.0087	0.051± 0.0042

Table S.2: Prediction accuracy: mean  $R^2$  together with the STD from the 5-fold cross-validation of fine-tuned models on DFT datasets

Model		Egc	Xc	eps	Ei	nc	Eat	Eea	Egb
PLM-ST	train	0.996± 0.0008	0.989± 0.0034	0.992± 0.0030	0.992± 0.0018	0.993± 0.0021	0.997± 0.0013	0.994± 0.0012	0.995± 0.0010
	test	0.911± 0.0141	0.330± 0.1051	0.726± 0.0587	0.791± 0.0497	0.817± 0.0560	0.980± 0.0085	0.910± 0.0198	0.925± 0.0210
SML-ST	train	0.995± 0.0013	0.989± 0.0037	0.989± 0.0027	0.989± 0.0011	0.991± 0.0022	0.994± 0.0016	0.991± 0.0030	0.993± 0.0015
	test	0.901± 0.0222	0.340± 0.1257	0.726± 0.0383	0.802± 0.0511	0.812± 0.0581	0.967± 0.0150	0.915± 0.0153	0.920± 0.0290
INIT-ST	train	0.995± 0.0004	0.989± 0.0011	0.991± 0.0018	0.991± 0.0045	0.991± 0.0036	0.997± 0.0017	0.992± 0.0037	0.992± 0.0032
	test	0.843± 0.0262	-0.002± 0.1420	0.598± 0.1218	0.775± 0.0649	0.741± 0.0597	0.974± 0.0068	0.888± 0.0311	0.879± 0.0269
PLM-MT	train	0.995± 0.0011	0.989± 0.0018	0.992± 0.0014	0.992± 0.0018	0.994± 0.0013	0.994± 0.0012	0.993± 0.0009	0.995± 0.0011
	test	0.916± 0.0169	0.380± 0.0572	0.787± 0.0282	0.892± 0.0272	0.866± 0.0222	0.916± 0.0175	0.938± 0.0229	0.951± 0.0108
SML-MT	train	0.993± 0.0008	0.987± 0.0016	0.989± 0.0039	0.991± 0.0009	0.989± 0.0028	0.990± 0.0013	0.989± 0.0017	0.993± 0.0012
	test	0.906± 0.0245	0.360± 0.0421	0.778± 0.0264	0.872± 0.0263	0.858± 0.0141	0.893± 0.0183	0.930± 0.0240	0.947± 0.0102
INIT-MT	train	0.996± 0.0007	0.992± 0.0025	0.995± 0.0010	0.995± 0.0013	0.995± 0.0009	0.994± 0.0017	0.995± 0.0017	0.996± 0.0015
	test	0.867± 0.0191	0.229± 0.0824	0.767± 0.0295	0.870± 0.0381	0.851± 0.0293	0.911± 0.0096	0.926± 0.0261	0.946± 0.0142
TransPolymer	test	0.920	0.500	0.760	0.840	0.820	-	0.910	0.930
TransPolymer-INIT	test	0.840	0.270	0.700	0.780	0.800	-	0.890	0.900
polyBERT	test	0.890± 0.0200	0.500± 0.1100	0.860± 0.0600	0.820± 0.0700	0.860± 0.0600	0.850± 0.0200	0.930± 0.0300	0.930± 0.0100

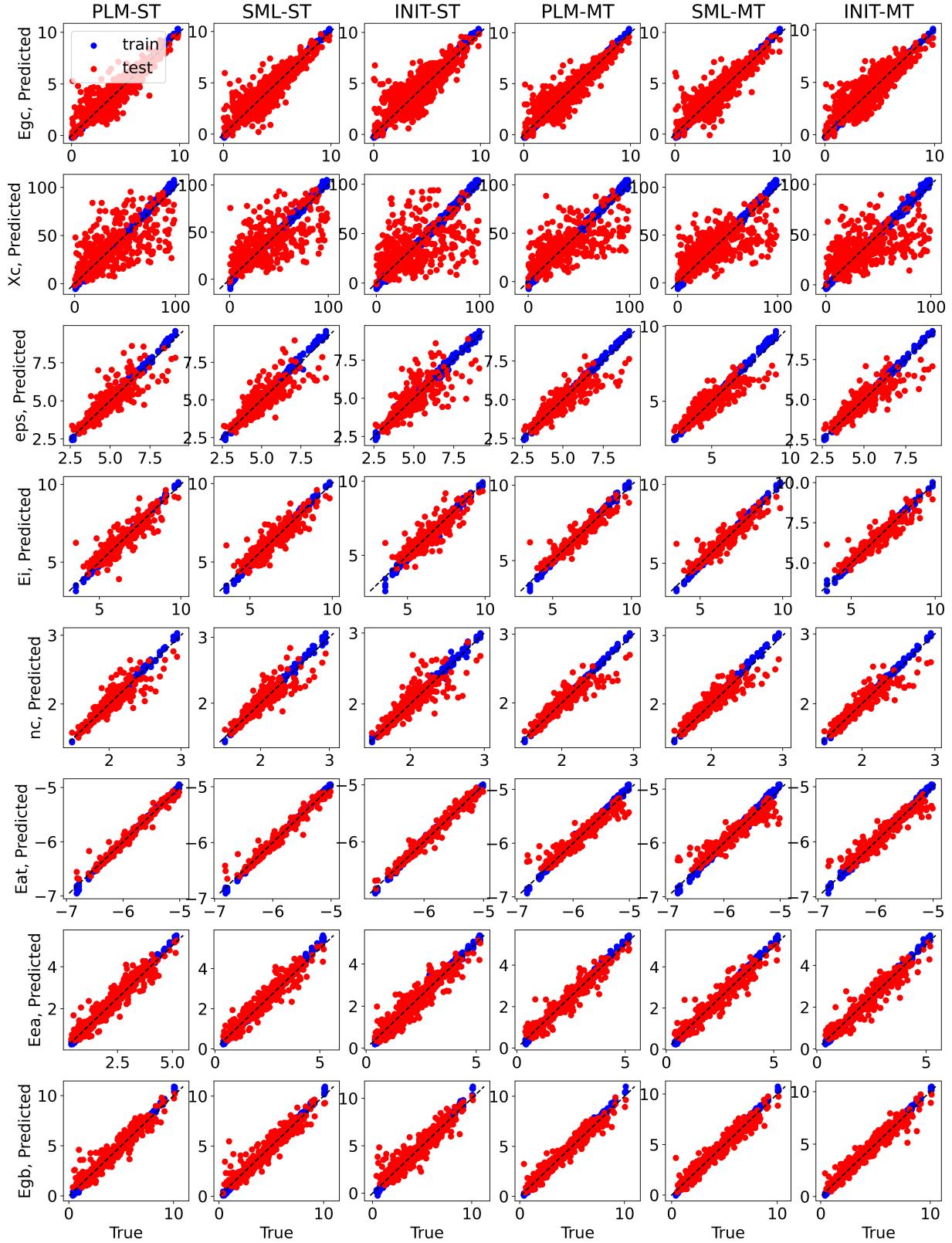


Figure S.2: Parity plots of 8 DFT properties for both training and test sets of all fine-tuned models from the 5-fold cross-validation.

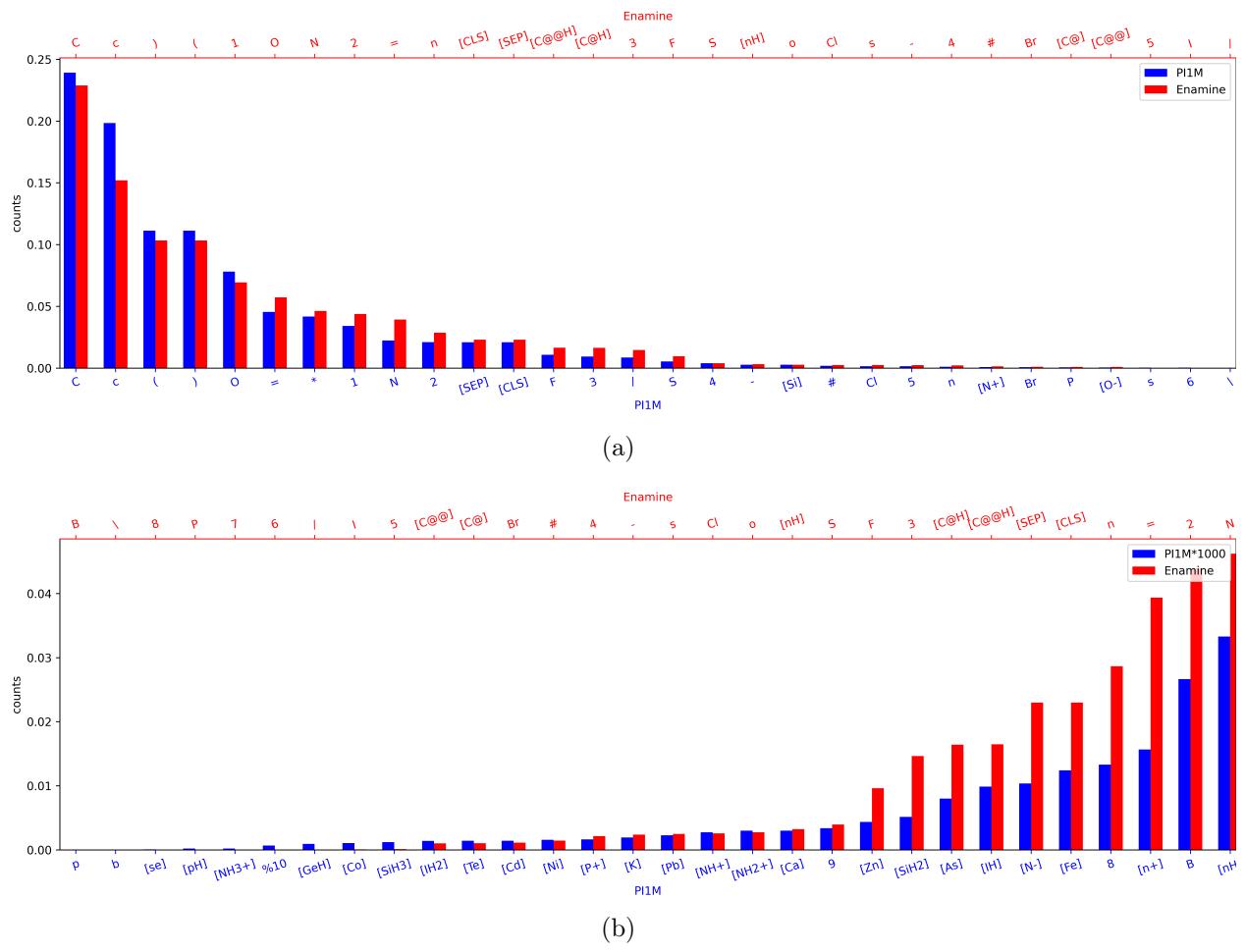
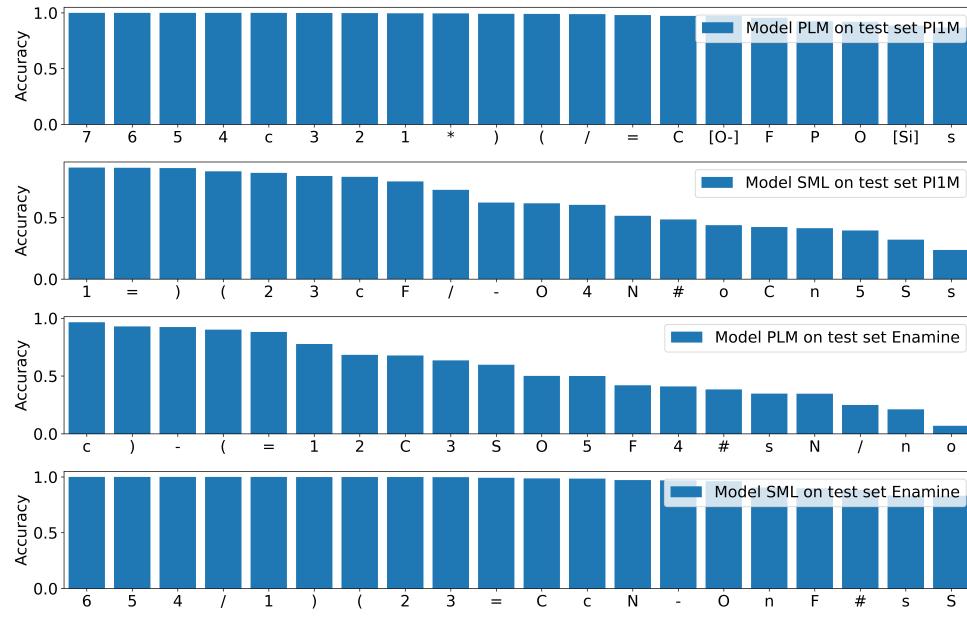
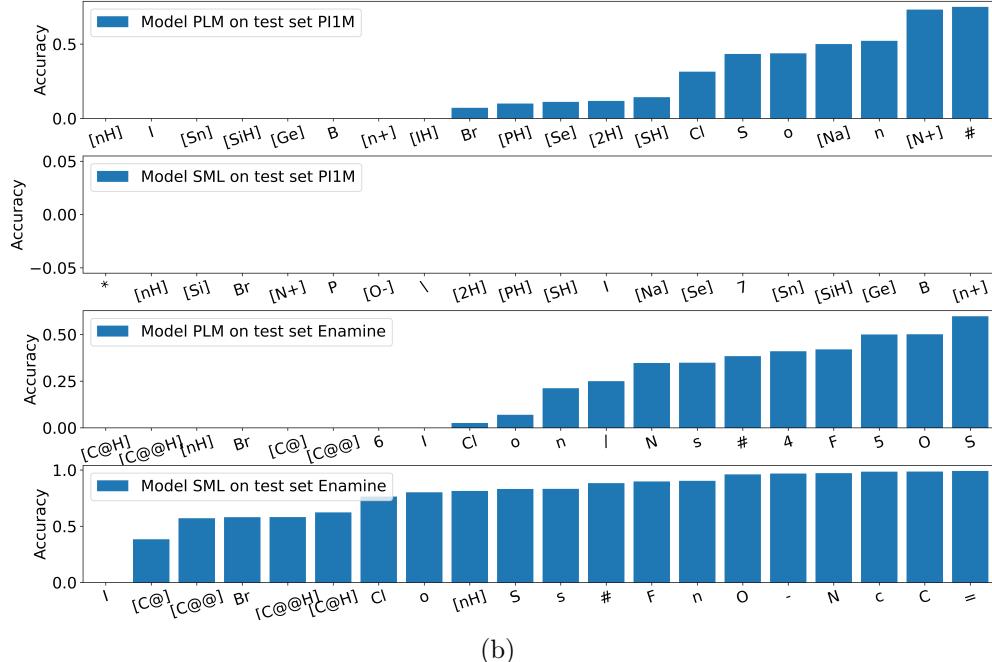


Figure S.3: Counts (percentage) of tokens in the two PI1M and Enamine REAL training sets: (a) top-20 most frequent and (b) top-20 least frequent (0s ignored).



(a)



(b)

Figure S.4: Prediction accuracy of the (a) top-20 most accurately and (b) top-20 least accurately predicted tokens.

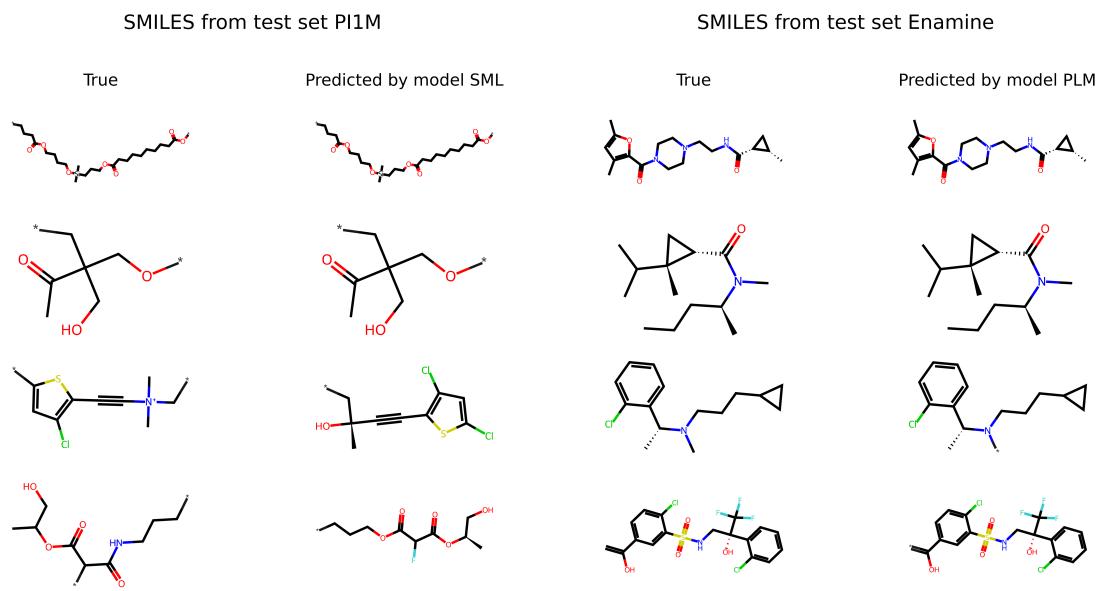


Figure S.5: Two successful (top-two rows) and two failed (bottom-two rows) SMILES in the out-of-distribution testing.

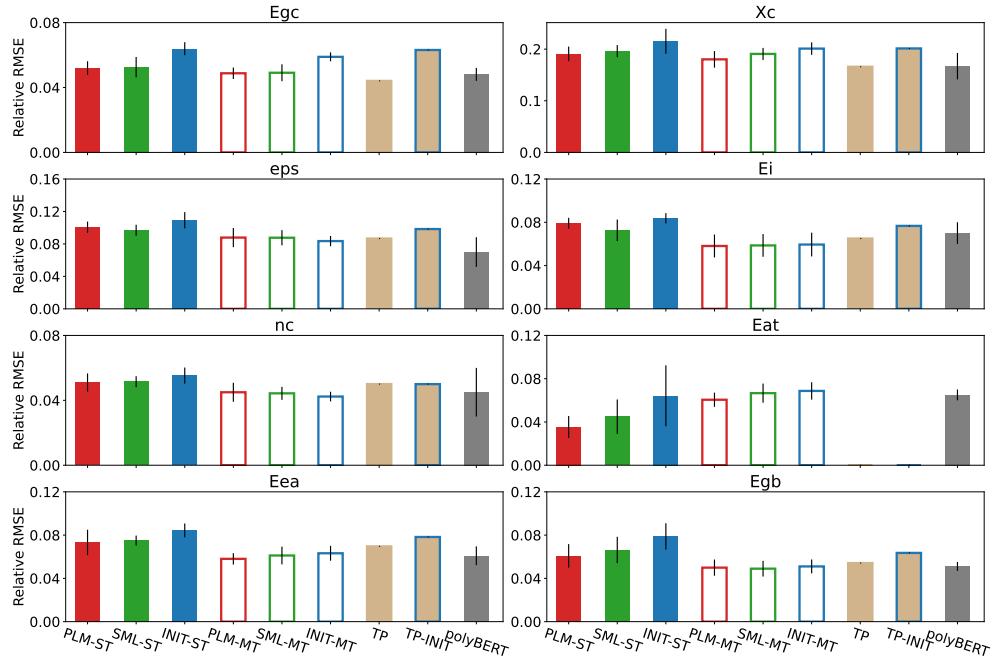


Figure S.6: Relative RMSE prediction error of DFT properties with BERT tokenizer. Error bars show the STD from the 5-fold cross-validation. Results of TransPolymer<sup>15</sup> (no STD reported, hence no error bars plotted), where TP and TP-INIT represent the models with and without pre-training, respectively, and polyBERT<sup>16</sup> are also shown for reference.

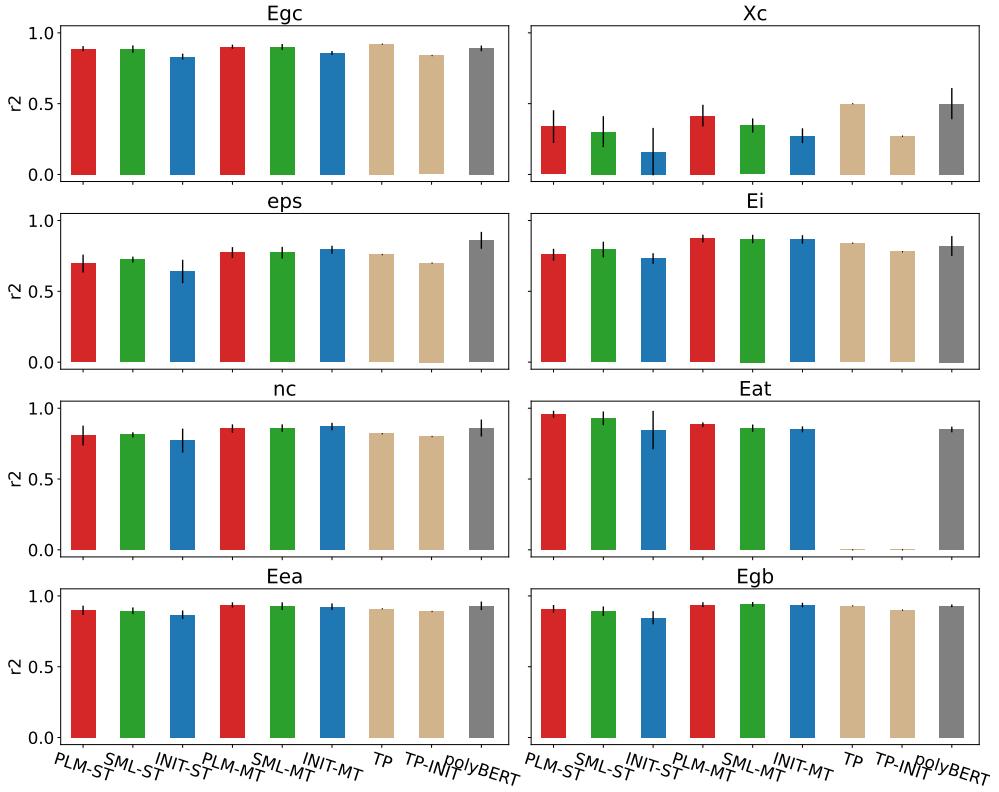


Figure S.7: Relative  $R^2$  prediction accuracy of DFT properties with BERT tokenizer. Error bars show the STD from the 5-fold cross-validation. Results of TransPolymer<sup>15</sup> (no STD reported, hence no error bars plotted), where TP and TP-INIT represent the models with and without pre-training, respectively, and polyBERT<sup>16</sup> are also shown for reference.

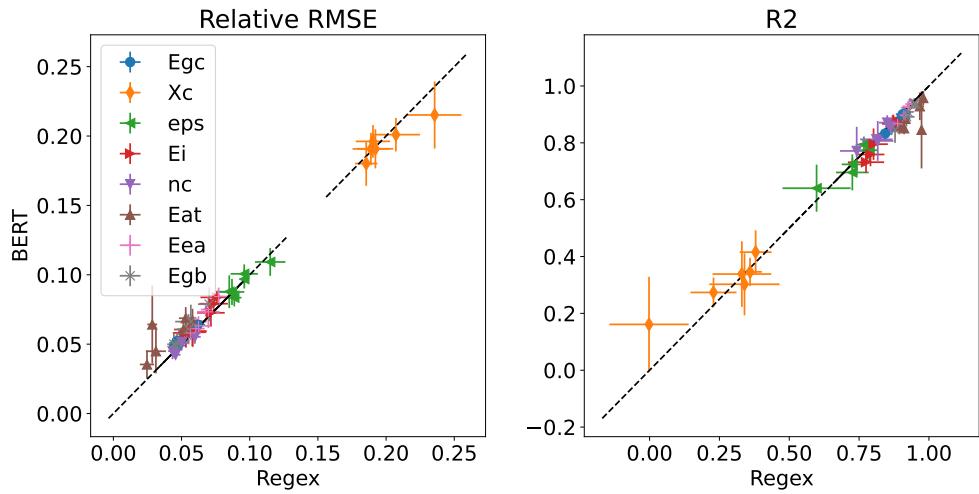


Figure S.8: Prediction performance: relative RMSE and  $R^2$  of models with BERT tokenizer versus regex tokenizer.

# TOC Graphic

