https://doi.org/10.1038/s44387-026-00113-2

polyRETRO: a language model approach to predict polymerization class and monomers for a target polymer

Sakshi Agarwal, Wei Xiong & Rampi Ramprasad ✉

While machine learning has transformed polymer design by enabling rapid property prediction and candidate generation, translating these designs into experimentally realizable materials remains a critical challenge. Traditionally, the synthesis of target polymers has relied heavily on expert intuition and prior experience. The lack of automated retrosynthetic tools to assist chemists limits the rapid, practical impact of data-driven polymer discovery. To expedite lab-scale validation and beyond, we present a retrosynthetic framework that leverages large language models (LLMs) to guide polymer synthesis. Our approach, which we call polyRETRO, involves two key steps: (1) predicting the most likely polymerization reaction class of a target polymer and (2) identifying the underlying chemical transformation templates and the corresponding monomers, using primarily natural-language-based constructs. This LLM-driven framework enables direct retrosynthetic analysis given just the target polymer SMILES string. polyRETRO constitutes an initial step towards a scalable, interpretable, and generalizable approach to bridge the gap between computational design and experimental synthesis.

The rapid advancement of machine learning (ML) has significantly advanced the field of polymer informatics, enabling efficient prediction of polymer properties and the generation of novel candidate structures with tailored functionalities[1–6]. These data-driven approaches have expanded the explored design space and reduced the reliance on trial-and-error experimentation[7–15]. However, a critical bottleneck that remains is the translation of computationally proposed polymers into synthetically accessible materials[16,17]. Identifying viable polymerization routes, including the appropriate selection of monomers, catalysts, solvents, and reaction conditions, still heavily relies on expert domain knowledge, chemical intuition, and literature mining[17]. This manual, iterative process is labor-intensive, non-systematic, and poorly suited for high-throughput autonomous workflows. Consequently, many ML-generated polymer candidates remain theoretical, with limited experimental realization. The absence of retrosynthetic analysis capabilities due to the complexity of polymer synthesis further exacerbates this gap.

Polymer synthesis typically involves multiple stages: initial reactions between one or more monomeric units via classical organic transformations, formation of oligomers, and subsequent polymerization steps that generate long polymer chains. This hierarchical and multi-step process makes retrosynthetic mapping from a target polymer back to its monomer precursors particularly challenging. Without a reliable way to map desired polymer structures back to feasible synthetic routes, the practical utility of ML-driven polymer design is severely constrained. Bridging this disconnect between in silico design and experimental synthesis is essential for advancing the real-world impact of polymer informatics.

In contrast, retrosynthesis for small molecules has already made significant progress through both rule-based systems and end-to-end machine learning models[18–25]. However, for polymers, retrosynthesis remains underdeveloped due to their structural complexity and the limited availability of high-quality reaction data. To address this gap, Chen et al. collected a dataset of over 10,000 polymerization reactions and developed a system that recommends potential monomers or precursors by matching the target polymer to known reaction templates via structural similarity[17]. Recently, fine-tuned transformer models have also been used to predict both forward reactions and retrosynthetic paths for polymer and small molecule synthesis[26–30]. Rule-based tools like SMiPoly and polyVERSE have also been proposed to encode common polymerization patterns, but they do not include corresponding retrosynthetic logic[5,31,32]. Despite these advances, current methods are often limited in the types of reactions they support, lack flexible template handling, or are hard to interpret chemically, especially when it comes to polymer retrosynthesis.

To overcome these limitations for polymers, we propose a framework called polyRETRO that combines a curated set of fundamental polymerization reactions, represented through interpretable and generalizable reaction templates, with the reasoning capabilities of large pre-

School of Materials Science and Engineering, College of Engineering, Georgia Institute of Technology, Atlanta, GA, USA.
✉e-mail: rampi.ramprasad@mse.gatech.edu

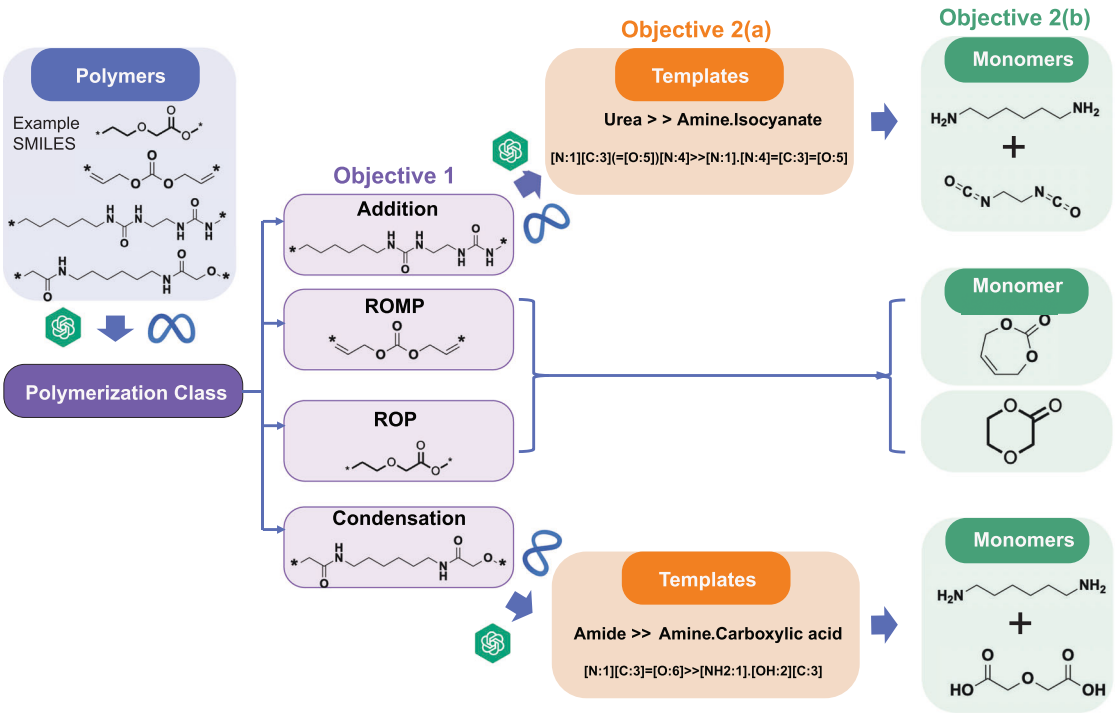


Fig. 1 | Retrosynthetic workflow of the polyRETRO pipeline for predicting monomers from a target polymer. The pipeline begins with the SMILES representation of a polymer repeat unit and proceeds through three sequential objectives. Objective 1 employs a classification language model to identify the polymerization class. Objective 2(a) predicts the corresponding reaction template that captures the underlying functional group transformation. Objective 2(b) utilizes the predicted template to determine the monomer structures required for synthesizing the target polymer.

trained language models (LLMs). polyRETRO proceeds via two objectives based on the polymer synthesis procedure: Objective 1 fine-tunes a LLM to identify the polymerization reaction class that may be used to synthesize a given polymer; Objective 2(a) involves fine-tuning a second LLM to predict the underlying functional group transformations of the reacting monomers in the form of a reaction template. Furthermore, Objective 2(b) finally determines the monomers by locating synthons, the potential synthetic building blocks of the polymer through bond cleavage and cyclization. Figure 1 summarizes the polyRETRO workflow. Beginning with the SMILES representation of a target polymer repeat unit, the polyRETRO pipeline first assigns a polymerization class. Based on this classification, it predicts a corresponding reaction template for addition and condensation polymerizations, and subsequently identifies the monomers required to synthesize the target polymer. In contrast, for ROMP and ROP polymerizations, Objective 2 directly predicts the monomers by cyclizing the polymer structure. Although Objective 2(a) still leverages manually curated reaction templates, polyRETRO differs significantly from traditional template matching methods, as it does not require explicit structure comparison or manual template selection. Instead, we fine-tune a LLM to directly predict a suitable reaction template, in natural language, from the SMILES string of a polymer's repeat unit. polyRETRO retains the interpretability and chemical reasoning embedded in rule-based templates while harnessing the scalability, flexibility, and pattern-recognition strengths of LLMs. As a result, it enables a more efficient and generalizable path toward retrosynthetic analysis of computationally designed polymers, bridging the gap between in silico design and experimental feasibility.

Results

Objective 1: Predicting polymerization class

Objective 1 of polyRETRO focuses on predicting the class of the polymerization reaction necessary to synthesize a target polymer. Possible outcomes include condensation, addition, ring-opening polymerization (ROP), and ring-opening metathesis polymerization (ROMP).

Table 1 | Summary of polymer counts per polymerization class in the dataset used for polyRETRO

Polymerization class	Known polymers	Train polymers	Test polymers
Condensation	8044	500	7544
Addition	2308	500	1808
Ring opening	586	500	86
ROMP	307	250 + 250[a]	57 + 28232[a]

[a]Virtual polymers.

Our dataset of polymerization reactions contains 11,245 previously reported polymerization paths, for 10,051 homopolymers starting from 5105 unique monomers. This dataset was manually collected from various resources, including online repositories and published journal articles[10,33–35]. Four polymerization classes were considered, including condensation (8044 polymers), addition (2308 polymers), ring-opening (586 polymers), and ring-opening metathesis (307 polymers) as shown in Table 1. The polymers and reactant molecules are made up of 12 elements (i.e., C, H, B, O, N, S, P, Si, F, Cl, Br, and I) and a variety of polymer classes. In the present work, the role of other factors, such as solvents, catalysts, and experimental conditions, is neglected. It is worth pointing out that both the training and test datasets are composed of homopolymers and exclude copolymers, polymer blends, ladder, cross-linked, and metal-containing polymers.

The dataset used to train the model for Objective 1 of the polyRETRO pipeline comprises 500 data points from each polymerization class, ensuring a balanced training set of known polymers, as shown in Table 1. This choice was primarily guided by the limited availability of ROP samples (586 in total), and selecting 500 instances per class ensured uniform class representation while retaining sufficient data for evaluation. The selection of 500 data points per class was further guided by a learning curve analysis, which indicated that model performance begins to plateau beyond this training size, suggesting that additional data yields only marginal improvements

(Figs. S1 and S2). Once the 500 from each class were selected for training, the remaining data points, 7544 from condensation, 1808 from addition, and 86 from ring-opening polymerization (ROP), were reserved for testing. In the case of ROMP, due to the limited number of curated examples, the training set includes 250 manually curated polymers and 250 virtual polymers from a total of 28,482 polymers generated in our previous work[32]. The corresponding ROMP test set consists of the remaining 57 known and 28,232 virtual ROMP polymers. The train and test dataset was then used to fine-tune LLaMA and GPT.

The performance of the hyperparameter-optimized fine-tuned model is shown in Fig. 2B. The GPT-based fine-tuned model achieves the highest accuracy (0.98), followed by LLaMA (0.96). The weighted F1-scores follow a similar trend, with GPT achieving 0.98 and LLaMA 0.96, indicating strong and balanced classification performance across all classes. Additionally, both language models exhibit high weighted precision and recall (0.98 for GPT and 0.96 for LLaMA), further confirming their reliability and consistency in predictions. The fine-tuned GPT model (Fig. 2C) demonstrates strong class-specific performance, as shown by the normalized confusion matrix. The diagonal dominance highlights high per-class recall, with values of 89.5% for addition, 93.3% for condensation, 94.2% for ROP, and 100% for ROMP. Minor misclassifications are primarily observed between addition and condensation (5.9%) and between addition and ROP (4.4%), likely due to structural similarities in certain monomers. Predictions for ROP and ROMP exhibit high specificity with minimal cross-class confusion. The GPT model achieves per-class accuracies of 0.89 for addition, 0.93 for condensation, 0.94 for ROP, and 0.99 for ROMP (Fig. 2D), highlighting consistent performance across all reaction classes and especially strong classification of ROMP-type polymers. The per-class accuracies and confusion matrix for LLaMA are provided in Fig. S6. To evaluate the robustness and reproducibility of the model predictions, we conducted a consistency analysis by repeatedly generating predictions on a fixed test set of 100 samples using the same trained model. We performed three independent inference runs and observed highly consistent performance, with only marginal variations across runs (Fig. S7). The accuracy, precision, recall, and F1-score varied within a very narrow range (±0.005), indicating strong stability of the model outputs despite stochasticity introduced during generation.

In addition to fine-tuning LLMs, we trained traditional machine learning (ML) classifiers to predict the polymerization class using the same dataset. As shown in Fig. 2B, traditional machine learning models achieve strong performance but remain slightly inferior to the fine-tuned LLM. Tree-based methods such as XGBoost, LightGBM, Gradient Boosting, Decision Tree, and Extra Trees achieve weighted F1 scores in the range of 0.91-0.93, demonstrating robust classification capability across polymerization classes. In contrast, simpler linear and probabilistic models show comparatively lower performance; for example, the Ridge Classifier achieves an F1 score of 0.90, while Naïve Bayes performs significantly worse (F1 0.57), highlighting its limited ability to capture complex chemical patterns. The results for these non-tree-based models are provided in the Supporting Information (Fig. S7). Furthermore, the results demonstrate consistently high performance across all folds, with very low variance (0.005), indicating that the classification framework is stable and not sensitive to specific data splits. Overall, while traditional machine learning models demonstrate strong performance on the classification task, the language models

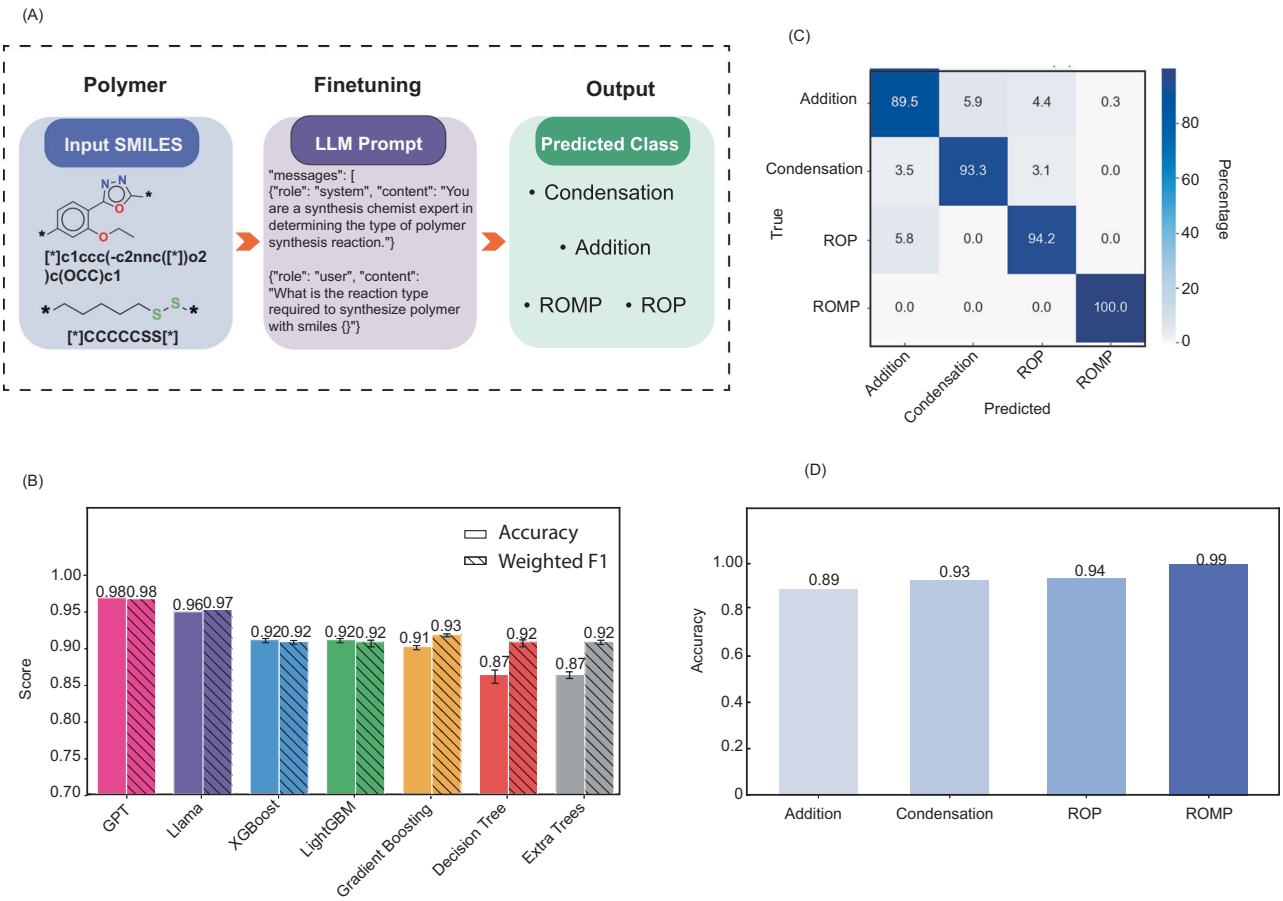


Fig. 2 | Workflow and performance of polymerization class prediction using language models and machine learning approaches. A Schematic representation of the LLM-based classification workflow, where polymer SMILES are mapped to polymerization classes. **B** Comparison of model performance in terms of accuracy and weighted F1-score for language models and traditional machine learning models. **C** Confusion matrix for the fine-tuned GPT model, illustrating prediction recall across polymerization classes. **D** Class-wise accuracy for each polymerization class predicted by the GPT model.

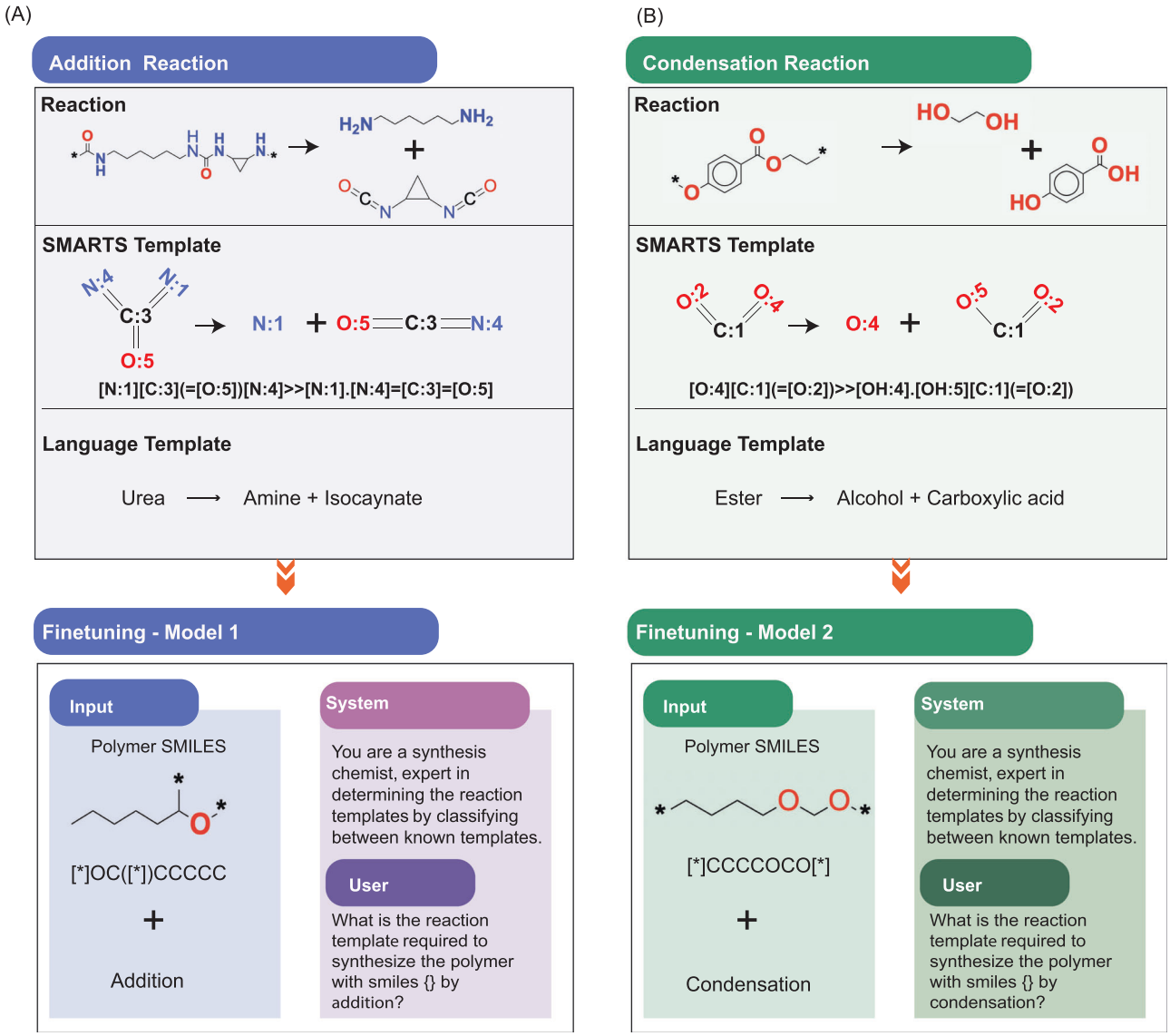


Fig. 3 | Workflow for reaction template generation and LLM fine-tuning in polymer retrosynthesis. A Addition polymerization pathway, illustrating reaction representation, SMARTS-based template extraction, and conversion to natural language templates, followed by fine-tuning of Model 1. **B** Condensation polymerization pathway, showing reaction representation, SMARTS-based template extraction, and language template formulation, followed by fine-tuning of Model 2.

consistently outperform them, achieving the highest weighted F1 scores of 0.983 and 0.973, respectively, along with superior balanced accuracy, precision, and recall even without relying on explicit molecular fingerprints as shown in Table S1. This highlights the ability of LLMs to implicitly capture complex chemical patterns. However, these advantages come with practical limitations. GPT-based models incur usage costs, and fine-tuning open-source models such as LLaMA is computationally intensive. Importantly, the proposed polyRETRO framework is model-agnostic, and any classification model can be integrated into this pipeline for subsequent monomer prediction.

Objective 2: Predicting reaction templates and monomers

Once the polymerization class suitable for a given polymer structure is identified, the next step of polyRETRO is to determine the corresponding monomer(s) that could have led to its synthesis. For ROP and ROMP, this retrosynthetic prediction is straightforward, as the monomers are often cyclic compounds that can be inferred by "closing" the repeating unit present in the polymer backbone. In such cases, a single monomer typically undergoes ring opening to form the polymer chain. Among the 29,287 ROP and ROMP polymers evaluated in this study, the correct monomer was accurately predicted for 29,056 cases, reflecting the well-defined and predictable nature of these polymerization mechanisms.

Objective 2(a): Predicting reaction templates

First, we focus on predicting the underlying reaction template used in the polymer's synthesis. This step of polyRETRO serves as a critical bridge between the polymer structure and its corresponding monomeric precursors. To accomplish this, we first represent polymerization processes through generalized retrosynthetic pathways. These synthesis routes, along with their associated reaction templates, are illustrated in Fig. 3 as standardized retrosynthetic schemes. Following conventional notation, the final polymer is depicted on the left side of each reaction, while the reactants, representing the monomers, are shown on the right. These pathways are encoded using two complementary template formats, each designed to capture the essential chemical transformations that govern polymer formation. This framework enables the LLMs to learn the underlying reaction logic and forms the foundation for the subsequent step of monomer prediction.

Table 2 | Total number of reaction templates for each polymerization class and the corresponding polymer count

Polymerization class	SMARTS	Language templates	Total polymers
Addition	129	42	2,273
Condensation	314	68 + 6[a]	7130 + 8000[a]

[a]Virtual polymers.

Table 3 | Total number of different polymer types and their corresponding language templates for addition polymerization

Polymer type	Templates count	Polymer count	Train count	Test count
Alkane	2	1315	1184	131
Alkene	5	209	188	21
Urea	2	144	130	14
Cycloalkane	3	106	95	11
Urethane	1	103	93	10
Amine	2	64	58	6
Alcohol	4	54	49	5
Carboxylic acid	1	31	28	3
Thiol	2	26	23	3
Ether	1	20	18	2
Aldehyde	4	20	18	2
Sulfone	2	18	16	2
Amine (Alcohol)	1	18	16	2
Amine (Thiol)	1	14	13	1
Ester	4	12	11	1
Silane	1	9	8	1
Silylether	2	6	5	1
Nitrile	1	6	5	1
Amide (Ester)	1	4	3	1
Imine	1	3	2	1
Disulfide	1	1	1	–
Total	**42**	**2308**	**2077**	**231**

The bold values depicts the total number of templates, polymers, train data, and the test data used.

Table 4 | Total number of different polymer types and their corresponding language templates for condensation polymerization

Polymer type	Templates count	Polymer count	Train count	Test count
Amide	2	2973	2676	297
Ester	4	2263	2037	226
Thioether	2	16	15 + 500[a]	1 + 500[a]
Urethane	2	14	13 + 500[a]	1 + 500[a]
Ethersulfone	1	1000[a]	500[a]	500[a]
Furan Maleimide	1	1000[a]	500[a]	500[a]
Imide	1	1000[a]	500[a]	500[a]
Oxime	1	1000[a]	500[a]	500[a]
Triazole	1	1000[a]	500[a]	500[a]
Urea	1	1000[a]	500[a]	500[a]
Ether	3	569	512	57
Amine	6	324	288	36
Cycloalkane	2	255	225	30
Imine	4	122	109	13
Alkene	2	33	29	4
Alkane	3	46	41	5
Thiol	2	48	43	5
Ether (Amine)	2	47	42	5
Disilane	1	35	32	3
Alcohol	2	32	28	4
Aldehyde	1	21	19	2
Sulfonic acid	1	30	27	3
Silane	1	13	12	1
Alkyne	1	18	16	2
Siloxane	1	15	13	2
Silylether	2	17	7	10
Thioamine	1	10	9	1
Amine (Thiol)	1	9	8	1
Disulfide	2	9	8	1
Thiourea	1	5	4	1
Sulfonamide	1	5	4	1
Azo	1	5	4	1
Thioamide	1	4	3	1
Amine (Thioether)	1	1	1	–
Thiocarbamoyl	1	3	2	1
Ester (Amine)	1	2	1	1
Amide (Ester)	1	2	1	1
Cyanide	1	2	1	1
Carboxylic acid	1	1	1	–
Sulfamide	1	1	1	–
Thioester	1	1	1	–
Thionoester	1	1	1	–
Total	**74**	**8044**	**7240 + 4000[a]**	**804 + 4000[a]**

[a]Virtual polymers.
The bold values depicts the total number of templates, polymers, train data, and test data used.

The first format employs SMARTS (SMILES Arbitrary Target Specification)[17] patterns, a flexible language for defining substructural transformations at the atomic level. In this representation, atoms are denoted using the format "[expr:n]", where expr is a valid atomic expression (e.g., element type, hybridization, aromaticity) and n is a mapping index used to track atom correspondence between reactants and products. For example, [C:2] refers to an aliphatic carbon atom labeled with the index 2. These simplified and generalizable SMARTS templates allow precise representation of the core transformation while maintaining broad applicability across diverse polymer structures. Further details on the SMARTS syntax and its application in polymer reaction modeling are provided in our previous work[17].

Within the polyRETRO pipeline, we also introduce natural language reaction templates to represent the same transformations in a human-readable format. This approach was motivated by the hypothesis that LLMs, which are inherently trained on natural language data, may better interpret and generalize from text-based descriptions than from symbolic representations like SMARTS. The language templates explicitly describe the functional group transformations that occur during polymerization, such as the esterification of a carboxylic acid with an alcohol. As illustrated in Fig. 3A, B, both addition and condensation reactions were translated into natural language templates that capture the key chemical logic. By framing reaction knowledge in a linguistically intuitive format, we aim to enhance polyRETRO's ability to reason over chemical processes and improve its

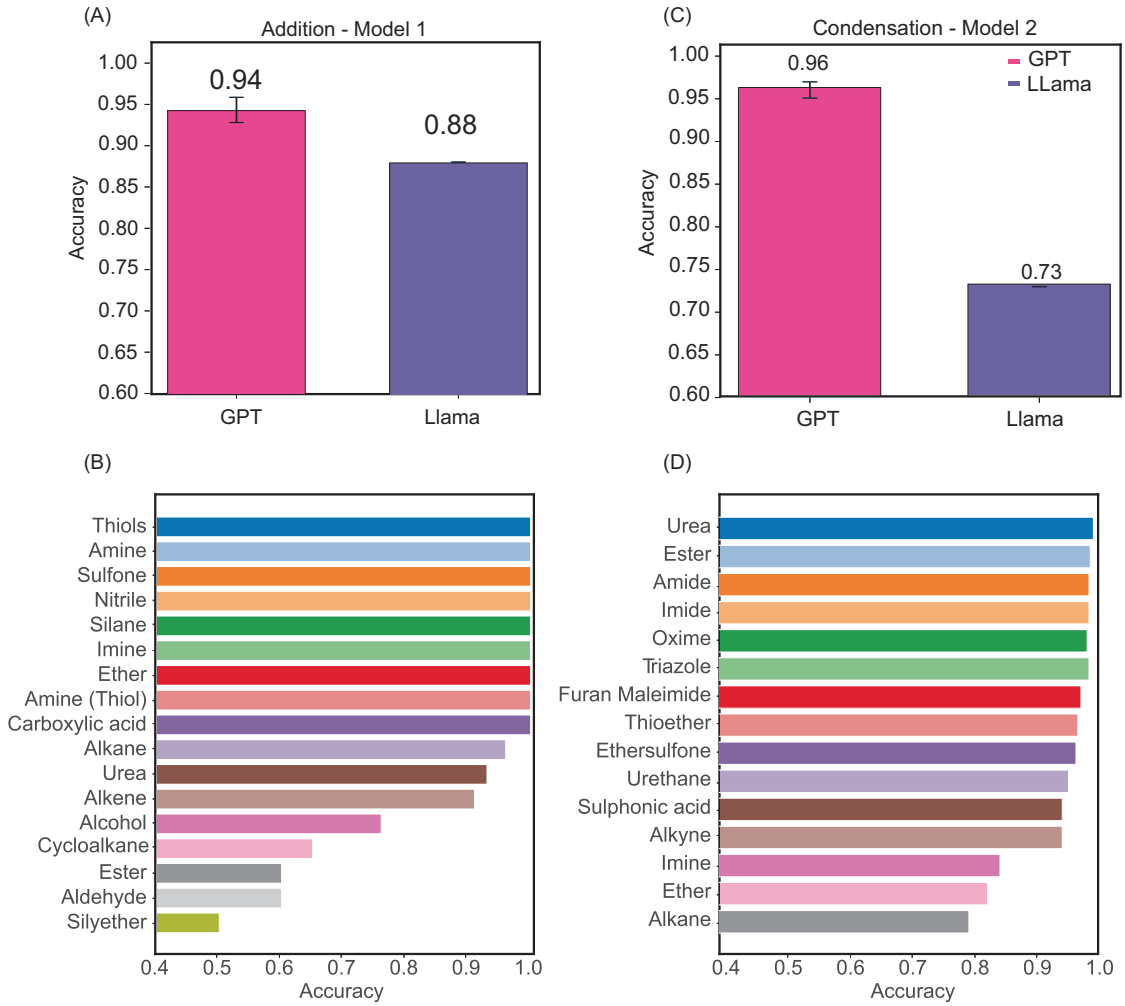
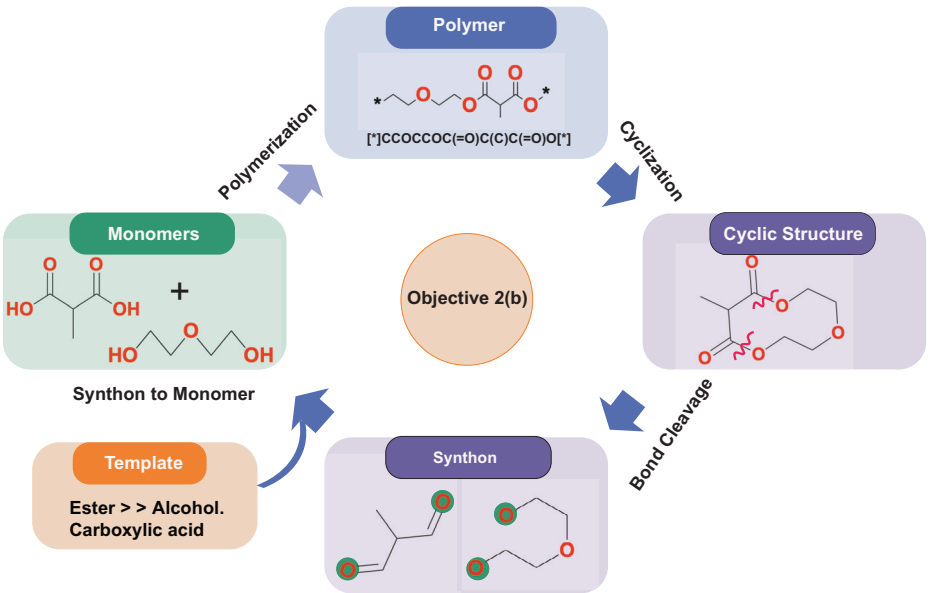


Fig. 4 | Template prediction performance for addition and condensation polymerization models. A Overall template classification accuracy for addition polymerization (Model 1) using GPT and LLaMA. **B** Class-wise template prediction accuracy for addition polymers using GPT. **C** Overall template classification accuracy for condensation polymerization (Model 2) using GPT and LLaMA. **D** Class-wise template prediction accuracy for condensation polymers using GPT.

Fig. 5 | Monomer prediction workflow in Objective 2(b) of the polyRETRO pipeline. The process begins with polymer SMILES, followed by cyclization to represent the continuous backbone. The predicted reaction template guides bond cleavage to generate synthons, which are subsequently transformed into monomer structures corresponding to the target polymer.



performance in downstream tasks such as monomer prediction and reaction classification.

It is important to note that the SMARTS-based representations of these reaction templates are highly detailed and structurally specific. Each SMARTS template encodes atom-level information, including hybridization, formal charge, connectivity, and atom mapping indices. As a result, the number of unique SMARTS templates corresponding to the same reaction type can be very large, since even minor structural variations across polymers lead to different SMARTS patterns. This granularity, while chemically precise, poses a challenge for generalization and model training due to the large template space. This was also one of the reasons for translating the SMARTS templates into natural language templates that abstract away atom-level details and instead focus on the functional group transformations central to the reaction mechanism. This conversion significantly reduced the number of unique templates by capturing reaction logic in a more generalizable and interpretable format. For example, multiple SMARTS templates representing variations of an amide formation reaction could be unified under a single language template such as Amide → Amine.Carboxylic acid. This simplification not only reduces the complexity of the classification task but also aligns better with the capabilities of LLMs. Therefore, the use of generalized language templates plays a crucial role in improving model interpretability, reducing data sparsity, and facilitating more robust learning.

As summarized in Table 2, a total of 42 unique natural language templates were developed for addition reactions, covering 2273 polymer instances, while 68 templates were created for condensation reactions, corresponding to 7130 manually curated known polymers, and 6 templates for click chemistry reactions, corresponding to a total of 8000 virtual polymers selected for this study. The virtual polymers are added to

Table 5 | The overall accuracy of all retrosynthetic steps for the best-performing models in the polyRETRO pipeline

Objective	Prediction task	Polymerization class	Accuracy
Objective 1	Polymerization class	–	0.98
Objective 2	Reaction template	Addition	0.93
Objective 2	Reaction template	Condensation	0.97
Objective 2	Monomer	Addition	0.89
Objective 2	Monomer (Expt. + virtual)	Condensation	0.87
Overall workflow	All tasks	All classes	0.88

Table 6 | Examples of predicted polymerization class, reaction templates, and monomers obtained from polymer SMILES

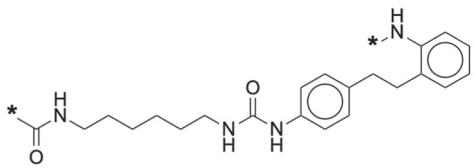
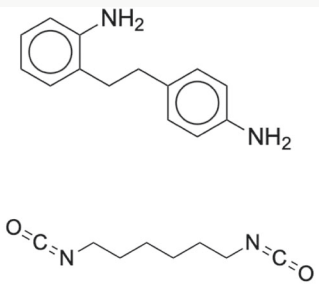
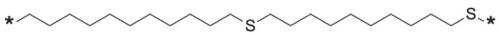
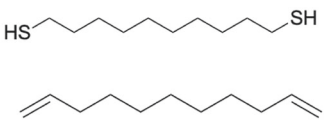
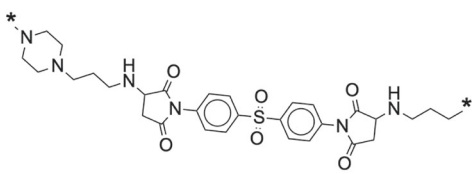
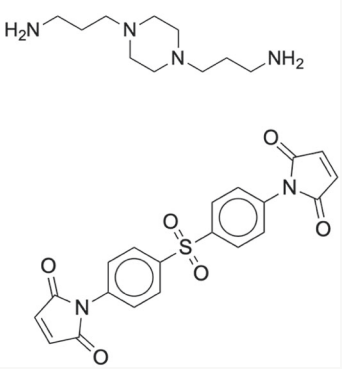
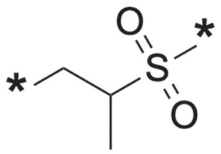
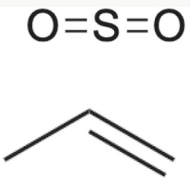
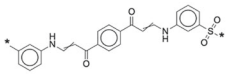
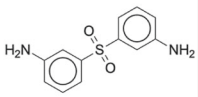
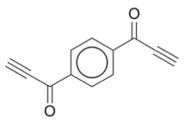
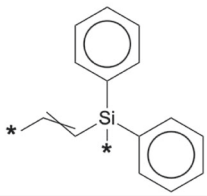
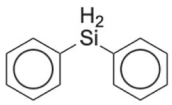
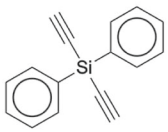
Polymer	Language Template	Monomers
SMILES: [*]Nc1ccccc1CCc1ccc(NC(=O)NCCCCCCNC([*])=O)cc1 	Addition Urea ⇒ Amine.Isocyanate	
SMILES: [*]CCCCCCCCCCSCCCCCCCCCCS([*]) 	Addition Thiol ⇒ Alkene.Thiol	
SMILES: [*]CCCNC1CC(=O)N(c2ccc(S(=O)(=O)c3ccc(N4C(=O)CC(NCCCN5CCN([*])CC5)C4=O)cc3)cc2)C1=O 	Addition Amine ⇒ Alkene.Amine	

Table 7 | Examples of predicted polymerization class, reaction templates, and monomers obtained from polymer SMILES

Polymer	Language Template	Monomers
SMILES: `[*][*]CC(C)S([*])(=O)=O` 	**Addition** Sulfone ⇒ Alkane.Sulfone	
SMILES: `[*]c1cccc(NC=CC(=O)c2ccc(C(=O)C=CNc3cccc(S([*])(=O)=O)c3)cc2)c1` 	**Addition** Nitrile ⇒ Alkyne.Amine	 
SMILES: `[*]C=C[Si]([*])(c1ccccc1)c1ccccc1` 	**Addition** Silane ⇒ Alkyne.Silane	 

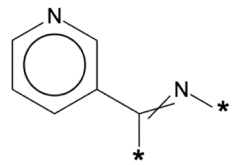
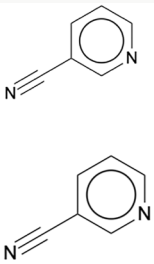
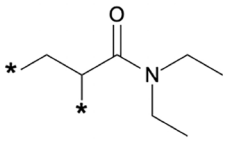
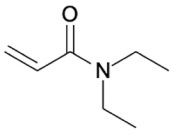
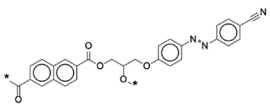
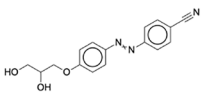
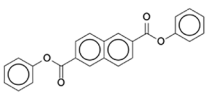
the dataset in order to add more polymer classes and different chemistries to the dataset. The addition and condensation reaction templates capture the underlying functional group transformations that drive polymer formation and are used to map the polymer structure to its likely synthetic route. It is important to note that the number of SMARTS templates corresponding to the language templates is significantly high, as shown in Table 2.

A further breakdown of these language templates by polymer type is provided in Tables 3 and 4 for addition and condensation reactions, respectively. For example, the majority of addition polymers are alkanes, which are associated with two distinct reaction templates: Alkane → Alkane.Alkane and Alkane → Alkene.Alkene. These templates represent different retrosynthetic possibilities for forming alkane-based polymers, depending on the specific reactant structures involved. The most prevalent class in this category is polyamides, which are predominantly derived from two common reaction templates: Amide → Amine.Carboxylic acid and Amide → Amine.Halide. These templates reflect the two major synthetic routes for amide bond formation via carboxylic acid activation or halide substitution. A detailed description of each template for all the polymer classes is provided in Tables S2 and S3.

As illustrated in Fig. 3, we trained separate models for the addition and condensation polymerization classes (Model 1 and Model 2). To initially assess the model's ability to understand both types of template formats (SMARTS and natural language), we fine-tuned GPT 3.5 using a subset of 1000 data points for each template format, resulting in four independent models corresponding to the two polymerization classes and the two template formats. Model performance was evaluated using accuracy, defined as the number of correctly predicted templates divided by the total number of data points. As shown in Fig. S9, GPT 3.5 trained on natural language templates achieved significantly higher accuracy for both addition and condensation compared to the models trained on SMARTS templates. These results suggest that LLMs interpret human-readable chemical descriptions more effectively than symbolic SMARTS patterns. Based on this insight, we proceeded to develop the final predictive models by fine-tuning GPT-3.5 and LLaMA 3B with the complete dataset of addition and condensation using natural language templates.

Addition templates: Model 1. For the addition polymerization model (Model 1), the complete addition dataset was divided into five different

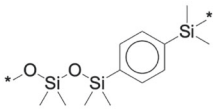
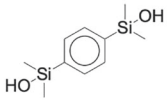
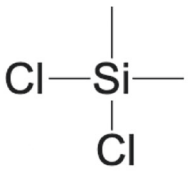
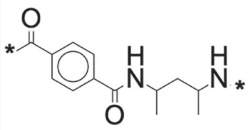
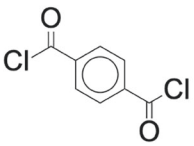
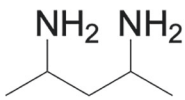
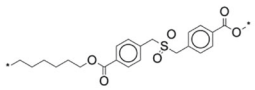
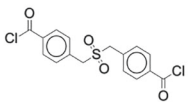
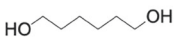
Table 8 | Examples of predicted polymerization class, reaction templates, and monomers obtained from polymer SMILES

Polymer	Language Template	Monomers
SMILES: `[*]/C(C1=CC=CN=C1)=N[*]` 	**Addition** Imine ⇒ Imine.Imine	
SMILES: `[*]CC([*])C(N(CC)CC)=O` 	**Addition** Alkane ⇒ Alkene.Alkene	
SMILES: `O=C([*])C1=CC=C2C(C=CC(C(OCC(COC3=CC=C(/N=N/C4=CC=C(C#N)C=C4)C=C3)O[*])=O)=C2)=C1` 	**Addition** Ester ⇒ Alcohol.Carboxylic acid	 

folds using a 90/10 split, where the training set included 2002 unique polymers and the test set comprised 271 unique polymers with their corresponding language templates. This split was intentionally chosen because Objective 2(a) is inherently more complex than Objective 1, as it involves predicting reaction templates and monomer structures rather than only the polymerization class. Consequently, a larger training dataset is required to adequately capture the diversity of polymerization mechanisms and reaction templates. The 90/10 split enables the training set to achieve broad coverage of these templates while still preserving a sufficiently large held-out test set for robust evaluation. The train and test sets contain various polymer types; however, there is an imbalance between the different polymer types as shown in Table 3. These splits were used to fine-tune both GPT 3.5 and LLaMA 3B models. Hyperparameter optimization for the GPT and LLaMA 3B model is detailed in Figs. S9, S10, and S11. Inference temperature of 0.5 was found to be optimal for GPT-3.5. The best-performing configuration for LLaMA fine-tuning was found to be a rank of 32, α of 64, and 20 training epochs.

Figure 4A presents the performance of the optimized fine-tuned models, demonstrating that both LLMs are capable of accurately predicting addition reaction templates for previously unseen polymers across different folds. The GPT model achieved a high overall accuracy of 0.93, indicating strong predictive capability and generalization. Across multiple folds, the GPT model also exhibits low variability as shown in Fig. 4A, highlighting the consistency and robustness of its predictions. LLaMA also performed well, though slightly below GPT, confirming its potential as a viable alternative. These results suggest that LLMs, particularly GPT, are highly effective at capturing the underlying reaction logic from polymer structures. Although the overall accuracy of LLaMA is lower than that of GPT, it offers several important advantages. Figure 4B provides a class-wise breakdown of accuracy for the GPT model across different polymer types. The results indicate consistently high accuracy for the major polymer classes, which have a strong representation of their corresponding reaction template in the training dataset. However, for underrepresented polymer types with limited data per reaction template, the model performance decreases slightly, reflecting

Table 9 | Examples of the predicted polymerization class, reaction templates, and the monomers using the polymer SMILES

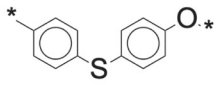
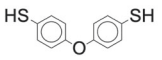
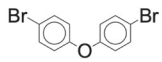
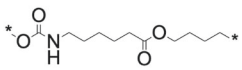
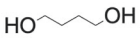
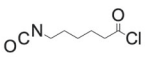
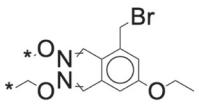
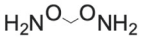
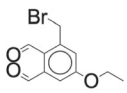
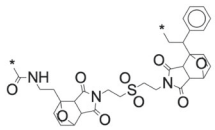
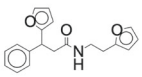
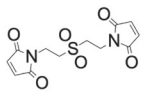
Polymer	Language Template	Monomers
SMILES: C[Si](O[Si](C)(C)O[*])(C)C1=CC=C([Si](C)([*])C)C=C1 	**Condensation** Alcohol ⇒ Alcohol.Halide	 
SMILES: [*]C(C1=CC=C(C(NC(CC(N[*])C)C)=O)C=C1)=O 	**Condensation** Amide ⇒ Amine.Carboxylic acid	 
SMILES: [*]CCCCCCOC(=O)c1ccc(CS(=O)(=O)Cc2ccc(C(=O)O[*])cc2)cc1 	**Condensation** Ester ⇒ Alcohol.Carboxylic acid	 

the influence of insufficient training examples on predictive accuracy. For instance, silylether constitutes only 0.25% of the training dataset and is associated with two distinct reaction templates, while aldehyde represents about 0.9% of the dataset with four different reaction templates. Consequently, these sparsely represented polymer types, each associated with multiple reaction pathways but limited data coverage, exhibit poorer predictive performance compared to polymer classes with broader and more balanced representation in the dataset.

Condensation templates: Model 2. Similarly, for the condensation polymerization model (Model 2), we fine-tuned LLMs using the condensation polymer dataset paired with the corresponding natural language reaction templates. The dataset comprised two components: a curated experimental known polymer set of approximately 7000 condensation polymers corresponding to 68 reaction templates, and a second set of 8000 virtual polymers generated in-house using common condensation templates and click chemistry strategies (Table S3). The distribution of polymer types and their associated templates across the experimental and virtual datasets is summarized in Table 4. For fine-tuning, the experimentally known polymer dataset was divided into five folds of a 90/10 train-test split, ensuring that all unique reaction templates were represented within the training set across various polymer classes. Furthermore, the same 90% experimental training set was augmented with 500 additional virtual data points from each of the eight virtual polymer classes, incorporated into both the training and test subsets (Table 4). This augmentation step expanded the chemical and structural diversity of the training corpus, enabling the model to generalize more effectively across novel polymer types and reaction chemistries, thereby improving the robustness of the condensation polymerization predictions.

Both GPT 3.5 and LLaMA 3B were fine-tuned using this dataset. The comparative performance of the models is shown in Fig. 4C, which clearly indicates that GPT outperforms LLaMA. GPT has an accuracy of 0.97 with a standard deviation of ±0.02 across multiple folds. In contrast, LLaMA has a lower accuracy of 0.73. Class-wise performance for the GPT models is shown in Fig. 4D. GPT achieves high accuracy in identifying the correct reaction templates. The model performs exceptionally well for classes having balanced 500 data points per class in both training and testing. This balanced representation ensures that the fine-tuned models are able to generalize effectively and accurately predict the reaction templates for the dominant classes of condensation polymers. Furthermore, for other classes like ester, amide, ethersulfone, sulfonic acid, alkyne, ether, etc, the model is able to effectively select the reaction templates.

Table 10 | Examples of the predicted polymerization class, reaction templates, and the monomers using the polymer SMILES

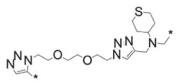
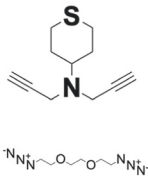
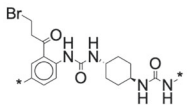
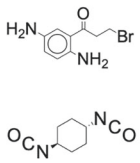
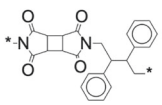
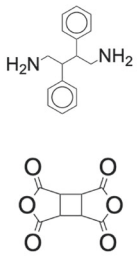
Polymer	Language Template	Monomers
SMILES: `[*]C1=CC=C(SC2=CC=C(O[*])C=C2)C=C1` 	**Condensation** Thioether ⇒ Thiol.Halide	 
SMILES: `COC(NCCCCCC(OCCCC[*])=O)=O` 	**Condensation** Urethane ⇒ Alcohol.Isocyanate	 
SMILES: `BrCC1=CC(OCC)=CC(/C=N/OCC)=C1/C=N/O[*]` 	**Condensation** Oxime ⇒ Hydroxylamine.Ketone(aldehyde)	 
SMILES: `C=S(CCN1C(C2C3C=CC(O3)(CCNC([*])=O)C2C1=O)=O)(CCN4C(C(C5C=CC6(O5)C(C7=CC=CC=C7)C[*])C6C4=C)=C)=C` 	**Condensation** Furan maleimide ⇒ Furan.Maleimide	 

Objective 2(b): Monomer prediction

Using the fine-tuned models (Model 1 and Model 2), we are now able to determine the reaction template corresponding to a given polymer for both addition and condensation, which is a critical step toward predicting the monomers required for its synthesis. The final step in polyRETRO involves linking the polymer SMILES and the predicted reaction template to the actual monomer structures. This process is illustrated in Fig. 5. Starting from the polymer SMILES, as an initial preprocessing step, we cyclically connect the head and tail atoms of the repeat unit to simulate the polymer's continuous backbone. This ensures that all bond environments are treated symmetrically and enables uniform cleavage logic across different polymer types. Based on the predicted template, we identify the specific bond(s) that need to be cleaved to derive the corresponding synthons. A synthon is an idealized fragment of a target molecule that represents a potential synthetic building blocks[36]. These fragments will have open ends, for instance, after breaking a C–O bond in an ether, we get a carbon atom and an oxygen atom, each with a free valence (C*, O*). The atomic environment around the break points is preserved, so we can attach the right end group. The reaction logic encoded in the template is then reapplied to reconstruct the full monomer structures from these fragments, i.e., synthons. For example, in the case of a polyester, a cyclic structure is first generated from the polymer SMILES. The ester linkage is then identified as the bond to be broken, as dictated by the corresponding Ester → Alcohol.Carboxylic acid template. Upon cleavage, the resulting fragments, an alcohol and a carboxylic acid functional group are interpreted as the monomers that would undergo condensation to form the original polymer.

The overall retrosynthesis accuracy for the prediction of monomers for the condensation polymerization dataset was 0.87. Here, the accuracy is defined as the correctly predicted monomers divided by the total number of data points. Furthermore, the polyRETRO pipeline achieved a high retrosynthesis accuracy of 0.89 on the addition polymerization dataset. The accuracies for each objective of the polyRETRO workflow are given in Table 5. The slight reduction in accuracy compared to template prediction arises from the strict criteria of an exact match between the predicted and ground truth monomers. In several cases, the

Table 11 | Examples of the predicted polymerization class, reaction templates, and the monomers using the polymer SMILES

Polymer	Language Template		Monomers
SMILES: [*]CN(C1CCSCC1)CC(N=N2)=CN2C COCCOCCN3C([*])=CN=N3 	Condensation Triazole Alkyne.Azide	⇒	
SMILES: [*]C1=CC(C(CCBr)=O)=C(NC(NC2CCC(NC(N[*]) =O)CC2)=O)C=C1 	Condensation Urea Amine.Isocyanate	⇒	
SMILES: O=C(N([*])C1=O)C2C1C(C(N3CC(C(C[*]) C4=CC=CC=C4)C5=CC=CC=C5)=O)C2C3=O 	Condensation Imide Amine.Anhydride	⇒	

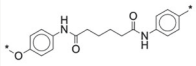
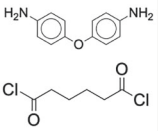
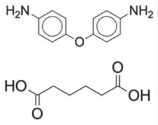
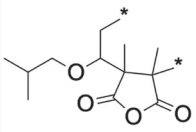
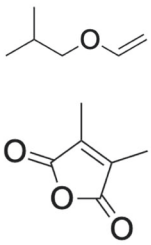
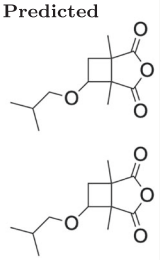
predicted monomers differed only in their terminal groups. Therefore, such predictions should not be regarded as entirely incorrect. The accuracy does not capture these slightly different but acceptable predictions. Some examples of the predicted monomers for both addition and condensation are provided in Tables 6–11. The overall monomer prediction, as shown in Tables 7–11, is quite robust, suggesting the effectiveness of the template-guided polyRETRO approach.

Discussion

In this work, we introduce an LLM-based framework called polyRETRO for predicting the monomers required to synthesize a target polymer. polyRETRO first determines the polymerization class (condensation, addition, ROP, or ROMP) of a target polymer using a fine-tuned GPT-3.5 model, achieving an accuracy of 0.98. In the next step, for the ROP and ROMP polymers, the monomer is directly inferred through ring-closure. In contrast, for addition and condensation, the LLMs infer the underlying functional group transformations through natural language reaction templates that provide an interpretable description of the underlying reaction chemistry. GPT-3.5 gave the most reliable template predictions for both addition and condensation polymerizations. These natural-language reaction templates are finally mapped to their corresponding monomers, yielding a complete monomer assignment for the target polymer. The polyRETRO pipeline serves as an initial step towards a scalable, accurate, and interpretable approach to monomer prediction that could accelerate polymer design and help close the gap between in silico polymer discovery and experimental realization.

Furthermore, for the complete evaluation of the whole pipeline, we performed an end-to-end assessment on 100 completely unseen polymer samples (25 from each polymerization class), tracking performance across all three stages: polymerization class prediction, reaction template selection, and monomer generation. The pipeline achieved an overall accuracy of 88%, with 3 failures at the polymerization classification stage (Objective 1), 4 failures at the reaction template stage (Objective 2(a)), and 5 additional failures at the monomer prediction stage (Objective 2(b)). These results highlight that errors in earlier stages tend to propagate downstream, as incorrect polymerization class predictions often lead to incompatible reaction templates and consequently incorrect monomer predictions. However, in certain cases, we observed that the final monomer prediction remained correct despite misclassifications in earlier steps, suggesting that different polymerization pathways can sometimes converge to the same underlying monomer structure. A detailed analysis of failure cases is presented in Tables 12–14. Table 12 highlights cases where only the monomer prediction fails, which may offer opportunities to explore alternative synthetic routes for the target polymer. Table 13 presents examples where the reaction template prediction fails; in some

Table 12 | Representative examples of monomer prediction failures in Objective 2(b) of the PolyRETRO pipeline

Polymer	Language Template	Monomers
SMILES: [*]Oc1ccc(NC(=O)CCCCC(=O)Nc2ccc([*])cc2)cc1 	**True : Condensation** Amide ⇒ Amine.Carboxylic acid **Predicted : Condensation** Amide ⇒ Amine.Carboxylic acid	True  Predicted 
SMILES: [*]CC(OCC(C)C)C1(C)C(=O)OC(=O)C1([*])C 	**True : Addition** Alkane ⇒ Alkane.Alkane **Predicted : Addition** Alkane ⇒ Alkane.Alkane	True  Predicted 

instances, the model hallucinates templates not present in the training or test data, making subsequent monomer prediction challenging. Table 14 summarizes cases where the pipeline fails at the initial polymerization classification stage, leading to entirely incorrect downstream predictions of both templates and monomers. Additionally, even among the failures at the final stage, some predictions were partially correct (e.g., only one monomer was incorrect), indicating that the model still captures meaningful retrosynthetic patterns. Despite 12 failures out of 100 cases, the pipeline often provides viable alternative predictions that can be explored as synthetic routes to the target polymer. Overall, these results highlight the robustness and practical utility of the polyRETRO pipeline as an efficient framework for monomer prediction and polymer retrosynthesis.

While the present work focuses on the fundamental challenge of predicting plausible monomer structures from a target polymer, achieving experimentally actionable polymer synthesis requires a more comprehensive framework that also accounts for reaction conditions. Key parameters such as catalysts, solvents, temperature, and processing conditions play a critical role in determining the feasibility and success of polymerization reactions. As a next step, the polyRETRO framework can be extended to incorporate reaction condition prediction.

Another important future direction is the extension of the PolyRETRO framework to copolymer systems, which would require incorporating copolymer-specific reaction templates and addressing additional challenges such as identifying multiple monomer compositions and sequences, thereby enabling retrosynthesis of more complex polymer architectures.

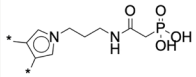
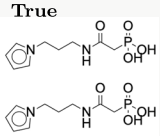
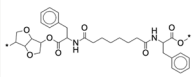
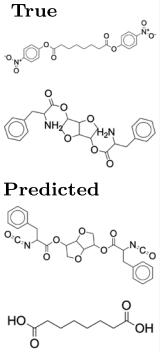
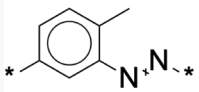
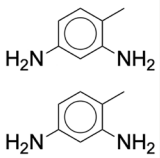
Methods

Objective 1: Finetuning

For the classification task of predicting the polymerization class, we fine-tuned LLMs, including OpenAI's proprietary GPT-3.5[37,38] and Meta's open-source LLaMA-3B[39]. The fine-tuning process involved training these base models on a curated dataset comprising prompt-response pairs specifically designed for the polymerization domain. Each data point was formatted as a question-answer pair consistent with the input structure expected by these LLMs. As illustrated in Fig. 2A, the fine-tuning followed a structured message format consisting of three roles: the system role, which defines the model's expertise and sets the context for the task; the user role, which poses the question "What is the reaction type required to synthesize polymer with SMILES {polymer SMILES}?", where the placeholder {polymer SMILES} is replaced with the actual SMILES (Simplified Molecular- Input Line-Entry System)[40,41] representation of the polymer; and the assistant role, which provides the correct answer. The model was trained to generate a response from a predefined set of four possible polymerization classes: "Condensation", "Addition", "Ring-opening", and "ROMP". This structured approach enabled the model to effectively learn the classification task and accurately predict the polymerization reaction class for unseen examples.

While both GPT-3.5 and LLaMA-3B were fine-tuned for the polymerization classification task using the structured prompt-response format described above, the two models differ significantly in their fine-tuning flexibility and transparency. GPT-3.5, fine-tuned via the OpenAI API, offers ease of use for training and inference but provides limited

Table 13 | Representative examples of reaction template prediction failures in Objective 2(a) of the PolyRETRO pipeline

Polymer	Language Template	Monomers
SMILES: `[*]c1cn(CCCNC(=O)CP(=O)(O)O)cc1[*]` 	**True : Addition** Cycloalkane⇒ Cycloalkane.Cycloalkane **Predicted : Addition** Phosphate⇒ Phosphate(Interactant) + Monomer(Product)	**True**  **Predicted**
SMILES: `[*]OC(=O)C(Cc1ccccc1)NC(=O)CCCCCCC(=O)NC(Cc1ccccc1)C(=O)OC1COC2C([*])COC12` 	**True : Condensation** Amide⇒ Amine.Carboxylic acid **Predicted : Condensation** Amide ⇒ Carboxylic acid.Isocyanate	**True**  **Predicted**
SMILES: `[*]N=Nc1cc([*])ccc1C` 	**True : Condensation** Azo⇒ Amine.Amine **Predicted : Condensation** Azo ⇒ Nitrosoamine	**True**  **Predicted**

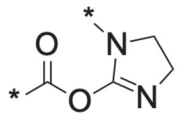
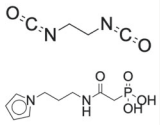
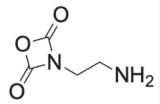
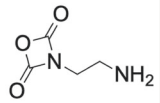
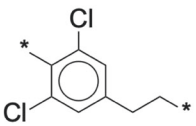
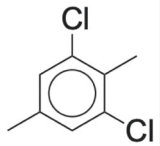
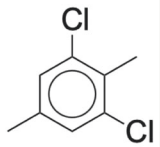
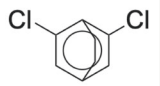
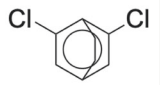
control over internal hyperparameters. As a result, GPT-3.5 functions largely as a black-box model, with fine-tuning restricted to a few configurable parameters such as the number of training epochs and the softmax inference temperature (T), which is a measure that controls the degree of randomness in the model's token selection process. Moreover, the details of the underlying fine-tuning mechanisms and model architecture remain proprietary and are not publicly disclosed. In order to optimize the number of epochs, the model was fine-tuned on several epochs using the 2000 train points and tested on the remaining data. Epoch 5 was found to be optimal. Furthermore, for inference temperature, the GPT fine-tuned model on 2000 data points with epoch 5 was tested on 100 randomly selected data points with varying temperature. The optimum value of T was found to be 1, as shown in Fig. S3.

In contrast, fine-tuning the open-source LLaMA-3B model provided greater flexibility and control over the training process. The model was fine-tuned on in-house servers using Low-Rank Adaptation (LoRA)[42], a parameter-efficient approach that injects trainable rank-decomposed matrices into the pre-trained weight space. This technique substantially reduced computational overhead, accelerated training time, and lowered memory usage without compromising performance. We conducted systematic hyperparameter optimization for LLaMA-3B, tuning the rank (r), scaling factor (α), number of training epochs, and softmax temperature (T) during inference to maximize model accuracy. Detailed results of the optimization are provided in the Supporting Information. Specifically, Fig. S4 presents the selected values of rank and α, while Fig. S5 highlights that 10 training epochs produced the best performance. This level of control and transparency afforded by LLaMA-3B fine-tuning was advantageous for tailoring the model to the specific demands of the task in-hand. Performance of all these models was assessed using the accuracy metrics, which is defined as the total number of correct predictions divided by the total number of predictions.

Objcetive 1: ML models

We trained traditional machine learning (ML) classifiers to predict the polymerization class using the same dataset. This workflow began by computing Polymer Genome (PG) fingerprints[2]. These fingerprints are hierarchical descriptors that capture chemically meaningful structural and electronic features of polymer repeat units. They capture polymer information at multiple scales, including atomic fragment motifs, molecular QSPR-based descriptors, and larger-scale morphological characteristics such as side-chain structure and ring topology. This multiscale representation enables the model to account for the diverse chemical and structural factors that influence polymer properties. PG fingerprints were used as input features for the classification models. The full ML pipeline, including data preprocessing, model training, hyperparameter tuning, cross-validation, and evaluation, was implemented using scikit-learn. A thorough comparison of multiple algorithms was performed, and the top five models based on performance metrics were XGBoost, Random Forest, Gradient Boosting, Extra Trees, and Decision Tree. All models were evaluated using stratified five-fold cross-validation to ensure balanced representation across polymerization classes and to enhance the robustness of performance estimates. Details of the models and their corresponding hyperparameters are provided in the Supporting Information.

Table 14 | Representative examples of end-to-end pipeline failures spanning Objective 1 to Objective 2(b) in the PolyRETRO framework

Polymer	Language Template	Monomers
SMILES: [*]c1cn(CCCNC(=O)CP(=O)(O)O)cc1[*] 	**True : Addition** Ester⇒ Alcohol.Aldehyde **Predicted : Condensation** Imine⇒ Amine.Aldehyde	True  Predicted  
SMILES: [*]CCc1cc(Cl)c([*])c(Cl)c1 	**True : Condensation** Cycloalkane⇒ Cycloalkane.Cycloalkane **Predicted : Condensation** Alkane ⇒ Alkane.Alkane	True   Predicted  

Objcetive 2(a): Fine-tuning

To train the models for reaction template prediction, we fine-tuned the LLMs using these two types of structured templates as target outputs. This task builds upon the earlier classification of polymerization classes (objective 1) and represents a more fine-grained objective that enables the reconstruction of complete polymer synthesis routes. To fine-tune the LLM for reaction template prediction, the polymer-template dataset was reformatted into question-answer pairs, as shown in Fig. 3. Each question was phrased as "What is the reaction template required to synthesize the polymer with SMILES <Polymer SMILES> by addition/condensation?", where <Polymer SMILES> represents the specific polymer structure. The corresponding reaction template was provided as the assistant's response. During fine-tuning, the model was trained to associate these polymer SMILES inputs with the correct template from the curated set.

Data availability

The dataset generated and/or analysed during the current study is not publicly available due to IP protection being considered at the authors' institution. Requests for Code should be addressed to R.R.

Code availability

Requests for Code should be addressed to R.R.

Received: 2 December 2025; Accepted: 26 April 2026;
Published online: 08 May 2026

References

1. Batra, R., Song, L. & Ramprasad, R. Emerging materials intelligence ecosystems propelled by machine learning. *Nat. Rev. Mater.* **6**, 655–678 (2021).
2. Doan Tran, H. et al. Machine-learning predictions of polymer properties with polymer genome. *J. Appl. Phys.* **128**, 171104 (2020).
3. Kim, C., Batra, R., Chen, L., Tran, H. & Ramprasad, R. Polymer design using genetic algorithm and machine learning. *Comput. Mater. Sci.* **186**, 110067 (2021).
4. Kuenneth, C. & Ramprasad, R. polybert: a chemical language model to enable fully machine-driven ultrafast polymer informatics. *Nat. Commun.* **14**, 4099 (2023).
5. Kern, J., Su, Y.-L., Gutekunst, W. & Ramprasad, R. An informatics framework for the design of sustainable, chemically recyclable,

synthetically accessible, and durable polymers. *npj Comput. Mater.* **11**, 182 (2025).
6. Agarwal, S., Mahmood, A. & Ramprasad, R. Polymer solubility prediction using large language models. *ACS Mater. Lett.* **7**, 2017–2023 (2025).
7. Chen, L. et al. Polymer informatics: current status and critical next steps. *Mater. Sci. Eng.: R: Rep.* **144**, 100595 (2021).
8. Peerless, J. S., Milliken, N. J., Oweida, T. J., Manning, M. D. & Yingling, Y. G. Soft matter informatics: current progress and challenges. *Adv. Theory Simul.* **2**, 1800129 (2019).
9. Adams, N. & Murray-Rust, P. Engineering polymer informatics: towards the computer-aided design of polymers. *Macromol. Rapid Commun.* **29**, 615–632 (2008).
10. Otsuka, S., Kuwajima, I., Hosoya, J., Xu, Y. & Yamazaki, M. Polyinfo: Polymer database for polymeric materials design. In *Proc. International Conference on Emerging Intelligent Data and Web Technologies*, 22–29 (IEEE, 2011).
11. Kim, C., Chandrasekaran, A., Huan, T. D., Das, D. & Ramprasad, R. Polymer genome: a data-powered polymer informatics platform for property predictions. *J. Phys. Chem. C.* **122**, 17575–17585 (2018).
12. Audus, D. J. & de Pablo, J. J. Polymer informatics: opportunities and challenges. *ACS Macro Lett.* **6**, 1078–1082 (2017).
13. Mannodi-Kanakkithodi, A. et al. Rational co-design of polymer dielectrics for energy storage. *Adv. Mater.* **28**, 6277–6291 (2016).
14. Wu, Y., Guo, J., Sun, R. & Min, J. Machine learning for accelerating the discovery of high-performance donor/acceptor pairs in non-fullerene organic solar cells. *npj Comput. Mater.* **6**, 120 (2020).
15. Agarwal, S. & Singh, A. Recommendation system to predict the d-band center of core-shell bimetallic nanoparticles catalysts. *Adv. Theory Simul.* **8**, 2401460 (2025).
16. Tran, H. et al. Design of functional and sustainable polymers assisted by artificial intelligence. *Nat. Rev. Mater.* **9**, 866–886 (2024).
17. Chen, L., Kern, J., Lightstone, J. P. & Ramprasad, R. Data-assisted polymer retrosynthesis planning. *Appl. Phys. Rev.* **8**, 031405 (2021).
18. Segler, M. H., Preuss, M. & Waller, M. P. Planning chemical syntheses with deep neural networks and symbolic AI. *Nature* **555**, 604–610 (2018).
19. Schwaller, P. et al. Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. *ACS Cent. Sci.* **5**, 1572–1583 (2019).
20. Gao, H. et al. Using machine learning to predict suitable conditions for organic reactions. *ACS Cent. Sci.* **4**, 1465–1476 (2018).
21. Dai, H., Li, C., Coley, C., Dai, B. & Song, L. Retrosynthesis prediction with conditional graph logic network. *Adv. Neural Inf. Process. Syst.* **32** (2019).
22. Coley, C. W., Rogers, L., Green, W. H. & Jensen, K. F. Computer-assisted retrosynthesis based on molecular similarity. *ACS Cent. Sci.* **3**, 1237–1245 (2017).
23. Sacha, M. et al. Molecule edit graph attention network: modeling chemical reactions as sequences of graph edits. *J. Chem. Inf. Model.* **61**, 3273–3284 (2021).
24. Chen, Z., Ayinde, O. R., Fuchs, J. R., Sun, H. & Ning, X. G2Retro as a two-step graph generative models for retrosynthesis prediction. *Commun. Chem.* **6**, 102 (2023).
25. Yao, L. et al. Node-aligned graph-to-graph: elevating template-free deep learning approaches in single-step retrosynthesis. *JACS Au* **4**, 992–1003 (2024).
26. Ferrari, B. S., Manica, M., Giro, R., Laino, T. & Steiner, M. B. Predicting polymerization reactions via transfer learning using chemical language models. *npj Comput. Mater.* **10**, 119 (2024).
27. Liu, X. et al. Retrocaptioner: beyond attention in end-to-end retrosynthesis transformer via contrastively captioned learnable graph representation. *Bioinformatics* **40**, btae561 (2024).
28. Zheng, S., Rao, J., Zhang, Z., Xu, J. & Yang, Y. Predicting retrosynthetic reactions using self-corrected transformer neural networks. *J. Chem. Inf. Model.* **60**, 47–55 (2019).
29. Sun, R., Dai, H., Li, L., Kearnes, S. & Dai, B. Towards understanding retrosynthesis by energy-based models. *Adv. Neural Inf. Process. Syst.* **34**, 10186–10194 (2021).
30. Xiong, J. et al. Bridging chemistry and artificial intelligence by a reaction description language. *Nat. Mach. Intell* **7**, 782–793 (2025).
31. Ohno, M., Hayashi, Y., Zhang, Q., Kaneko, Y. & Yoshida, R. Smipoly: generation of a synthesizable polymer virtual library using rule-based polymerization reactions. *J. Chem. Inf. Model.* **63**, 5539–5548 (2023).
32. Gurnani, R. et al. Ai-assisted discovery of high-temperature dielectrics for energy storage. *Nat. Commun.* **15**, 6107 (2024).
33. Ma, R. et al. Rationally designed polyimides for high-energy density capacitor applications. *ACS Appl. Mater. Interfaces* **6**, 10445–10451 (2014).
34. Li, Z. et al. High energy density and high efficiency all-organic polymers with enhanced dipolar polarization. *J. Mater. Chem. A* **7**, 15026–15030 (2019).
35. Wu, C. et al. Flexible temperature-invariant polymer dielectrics with large bandgap. *Adv. Mater.* **32**, 2000499 (2020).
36. Corey, E. J. & Wipke, W. T. Computer-assisted design of complex organic syntheses: Pathways for molecular synthesis can be devised with a computer and equipment for graphical communication. *Science* **166**, 178–192 (1969).
37. Lee, J.-S. & Hsiang, J. Patent claim generation by fine-tuning openai gpt-2. *World Pat. Inf.* **62**, 101983 (2020).
38. Brown, T. et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **33**, 1877–1901 (2020).
39. Grattafiori, A. et al. The llama 3 herd of models. Preprint at *arXiv* https://doi.org/10.48550/arXiv.2407.21783 (2024).
40. Weininger, D. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* **28**, 31–36 (1988).
41. Lin, T.-S. et al. Bigsmiles: a structurally-based line notation for describing macromolecules. *ACS Cent. Sci.* **5**, 1523–1531 (2019).
42. Hu, E. J. et al. Lora: Low-rank adaptation of large language models. *ICLR* **1**, 3 (2022).

Acknowledgements

The authors acknowledge the Office of Naval Research for supporting this research. This work was supported by the Office of Naval Research through grants N00014-19-1-2103 and N00014-20-1-2175.

Author contributions

S.A. conceptualized the idea, performed the model training, data analysis, wrote the manuscript, and prepared the figures. W.X. performed the model training, data analysis, and wrote the manuscript. R.R. conceptualized the idea, supervised the overall research direction, and oversaw the writing and final approval of the manuscript

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s44387-026-00113-2.

Correspondence and requests for materials should be addressed to Rampi Ramprasad.