ELUCIDATING THE SPACE OF ENZYMATIC REACTION: A UNIFIED BENCHMARK AND PRETRAINED MODEL

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing reaction models primarily learn molecular transformations, whereas enzymatic reactions depend jointly on molecular structure and catalytic function. We formulate this problem as learning an *enzymatic reaction space* linking reactants, products, and Enzyme Commission (EC) annotations. To characterize this space, we introduce VenusRX-Bench, a unified benchmark for forward reaction prediction, single-step retrosynthesis, and EC-number prediction. VenusRX-Bench integrates reactions from multiple biochemical databases with standardized curation, leakage-controlled splits, and consistent evaluation. Benchmarking representative chemical and enzymatic models reveals a clear chemical-to-enzymatic domain gap, driven by limited domain data, catalytic-context dependency, and the difficulty of modeling large biomolecular structures.

To bridge this gap, we develop VenusRX, a unified T5-style sequence-to-sequence model for enzymatic reactions. VenusRX jointly learns forward prediction, retrosynthesis, and reaction reconstruction, with two-stage training on millions of template-expanded reactions followed by real biochemical reactions. In addition, optional EC conditioning incorporates catalytic context, while Molecule Library-Constrained Decoding improves the generation of complex biomolecules. Across benchmark tasks and challenging generalization splits, VenusRX achieves the best or competitive performance on most evaluated settings over representative chemical and enzymatic baselines. Moreover, EC information consistently improves reaction prediction, while learned reaction representations support accurate EC prediction, revealing a bidirectional relationship between reaction structure and catalytic function. Together, VenusRX-Bench and VenusRX provide a unified framework for elucidating and modeling enzymatic reaction space.

1 INTRODUCTION

Enzymes are remarkable biocatalysts capable of accelerating complex chemical transformations with extraordinary regio- and stereoselectivity under mild physiological conditions. These superior catalytic properties position enzymatic reactions as a sustainable alternative to traditional chemical synthesis, driving transformative applications in synthetic biology, pharmaceutical manufacturing, and green chemistry (Turner & O'Reilly, 2013; Sheldon & Woodley, 2018; Nielsen & Keasling, 2016). With the burgeoning demand for enzymatic synthesis, deciphering and modeling the underlying principles of enzymatic reactivity has emerged as a fundamental endeavor. In the pursuit of automated biocatalytic pathway design, data-driven approaches capable of predicting reaction outcomes, proposing biosynthetic routes, and inferring catalytic functions directly from molecular representations hold great potential (Delépine et al., 2018; Finnigan et al., 2021).

Currently, machine learning and deep learning methods have achieved substantial progress in conventional organic reaction modeling, including reaction outcome prediction, retrosynthesis, and reaction classification. This progress has been largely driven by the availability of large-scale reaction datasets extracted from chemical patents (Lowe, 2012; Kearnes et al., 2021), which have enabled efficient benchmarking (Wan et al., 2022) and pretraining (Schwaller et al., 2021b; Irwin et al., 2022) of AI models. However, for enzymatic reaction, the scarcity of high-quality, large-scale datasets and benchmarks has become a major bottleneck. First, reported enzymatic reactions are limited in quantity and scattered across heterogeneous databases, requiring extensive standardization, correction,

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

and curation (Heid et al., 2023). Second, existing studies for modeling enzymatic reactions vary in data sources, preprocessing pipelines, task definitions, and data splitting strategies (Probst et al., 2022a; Zheng et al., 2022; Heid et al., 2023), making their reported performances difficult to compare directly.

Moreover, enzymatic reactions differ from organic reactions in a fundamental way: the enzyme catalyst is not merely a rate accelerator but an active determinant of reaction outcome. Organic reaction outcomes are primarily governed by substrate functional group chemistry, yielding relatively consistent pathways across different conditions. In contrast, the high functional diversity of enzymes introduces additional dimensions into enzymatic reactions—including substrate specificity, stereoselectivity, and cofactor dependence—such that reaction outcomes depend jointly on both substrate and enzyme. Since existing organic reaction models are not designed to incorporate enzymatic features, they are insufficient to capture the complexity of enzymatic transformations. In this work, we formulate the **enzymatic reaction space** as the joint relationships among *reactants*, *products*, and *catalytic functions* (Figure 1). Specifically, we employ Enzyme Commission (EC) numbers as a proxy for catalytic function, as their hierarchical taxonomy systematically encodes enzyme specificity while reaction-EC pairs are more abundant in existing databases than reaction-enzyme pairs.

To chart the enzymatic reaction space, we first construct a comprehensive benchmark named **VenusRX-Bench**. It comprises 126,361 unique enzymatic reactions curated from multiple biochemical databases, all annotated with EC numbers and atom-atom mapping. To the best of our knowledge, it represents the largest curated dataset of enzymatic reactions to date. VenusRX-Bench covers three core tasks: forward reaction prediction, single-step retrosynthesis, and EC-number prediction. It employs leakage-controlled data splits to evaluate out-of-distribution generalization across distinct biochemical dimensions. Our evaluation reveals that directly transferring organic reaction models to enzymatic reactions yields suboptimal performance, primarily due to the organic-to-enzymatic domain gap: (1) the presence of enzymatic catalysis, and (2) the larger molecular size of biomolecules. Given the limited scale of enzymatic reaction data, this gap cannot be easily bridged by simple supervised learning.

Accordingly, we propose **VenusRX**, a unified pretrained language model for enzymatic reactions. VenusRX adopts an encoder-decoder architecture and is pretrained under a T5-style (Raffel et al., 2020a) multi-task objective on enzymatic reaction SMILES and EC numbers, encompassing span corruption, forward reaction prediction, and retrosynthesis. To address the scarcity of enzymatic reaction data, we generate million-scale synthetic enzymatic reactions based on extracted reaction templates and conduct two-stage pretraining on both synthetic and real enzymatic reactions. To incorporate catalytic function context, we randomly append EC prompts to the input during pretraining, enabling the model to optionally condition its predictions on EC numbers at inference time. To improve the generation of large biomolecules, we further design a trie-based constrained sampling strategy that effectively reduces invalid outputs. Experimental results demonstrate that VenusRX achieves the best or competitive performance on most evaluated settings compared to both organic-chemistry and biochemical baselines, and that each of the above designs contributes critically to model performance.

2 VENUSRX-BENCH

Biochemical reaction databases differ in their formats, identifiers, and structural coverage. Study-specific tasks and processing procedures further limit direct comparison of reported results. VenusRX-Bench addresses these differences through a common reaction collection, standardized prediction tasks, and shared evaluation rules.

2.1 DATA CONSTRUCTION AND SPLITS

Data Construction We integrate EC-annotated reactions from BRENDA, PathBank, Rhea, and KEGG into a common representation while retaining source identifiers for traceability. Records are deduplicated by normalized reaction text and EC annotation, preserving associations between the same reaction and different catalytic annotations. We then resolve molecular structures, use EnzymeMap (Heid et al., 2023) for reaction balancing where possible, and generate atom mappings with RXNMapper (Grandjean et al., 2026). Following ECReact (Probst et al., 2022a), product-side curation retains a single principal product and excludes records without reactants.

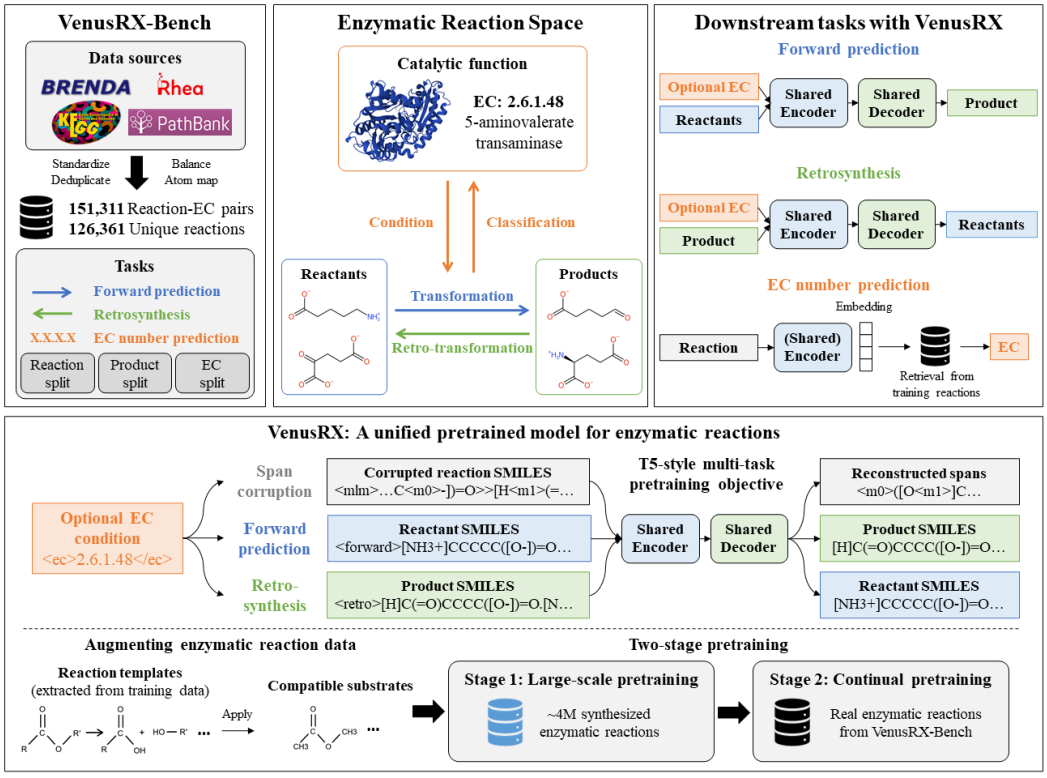


Figure 1: **Overview of the enzymatic reaction space and our framework.** VenusRX-Bench evaluates the relationships among reactants, products, and catalytic functions, while VenusRX learns a shared model of this space through domain-specific pretraining.

Table 1a compares the integrated collection with existing biochemical reaction datasets. VenusRX-Bench has the largest reported reaction count among the listed datasets and provides both EC annotations and atom mappings.

Splits We employ three split strategies for training, validation, and test sets. In each split, the grouping unit is disjoint across the training, validation, and test sets:

- **Reactant split.** Records are grouped by canonical reactant combinations to evaluate generalization to unseen combinations.
- **Product split.** Records are grouped by canonical principal products to evaluate generalization to unseen target products.
- **EC split.** Records are grouped by their original complete four-level EC annotations to evaluate generalization to unseen catalytic activities.

Product disjointness applies to the product side, while parent EC classes may overlap under the EC split. Detailed curation procedures and additional grouping constraints are provided in Appendix A.

2.2 Tasks and Evaluation

Each benchmark example is represented as (R, P, e), where R denotes the reactants, P the retained principal product, and e the associated EC annotation. We define three tasks:

$$\begin{aligned} \text{Forward prediction:} \quad & R \xrightarrow{e \text{ (optional)}} P, \\ \text{Single-step retrosynthesis:} \quad & P \xrightarrow{e \text{ (optional)}} R, \\ \text{EC-number prediction:} \quad & (R, P) \longmapsto e. \end{aligned}$$

(a) Dataset comparison.

Dataset	# Rxns	EC	Mapping
VenusRX-Bench (ours)	**126,361**	✓	✓
EnzymeMap (Heid et al., 2023)	47,974	✓	✓
ECReact (Probst et al., 2022a)	62,222	✓	×
CARE (Yang et al., 2024)	48,392	✓	×
EnzymeCage (Liu et al., 2026b)	29,608	✓	✓
BioNavi-NP (Zheng et al., 2022)	33,710	×	×

(b) Molecular statistics.

Metric	Dataset	Mean	Med.	Max
Sum of heavy atoms	VenusRX-Bench	**48.78**	**44**	**266**
	USPTO50K	28.29	28	97
SMILES length	VenusRX-Bench	**234.56**	**208**	**844**
	USPTO50K	189.83	186	707
molecule count	VenusRX-Bench	**2.11**	2	**5**
	USPTO50K	1.71	2	3
Max reactant heavy atoms	VenusRX-Bench	**36.23**	**33**	**265**
	USPTO50K	21.87	21	97

Table 1: Comparison of VenusRX-Bench with existing reaction datasets. (a) VenusRX-Bench provides the largest collection of unique enzymatic reactions together with EC annotations and atom–atom mappings. (b) Compared with USPTO50K, VenusRX-Bench contains larger and more complex reactant structures.All metrics are reactant-side.

For forward prediction and retrosynthesis, we report multi-reference Top-k exact-match accuracy for $k \in \{1, 3, 5, 10\}$. When multiple valid targets are associated with the same input in the test set, a prediction is counted as correct if any of the top-k candidates matches any recorded reference. Results with and without EC conditioning are reported separately.

For EC-number prediction, we evaluate all four hierarchy levels (EC1–EC4) using top-1 accuracy, weighted F1, and macro F1. At level ℓ, the target consists of the first ℓ fields of the EC annotation. The EC label is used only as the prediction target and is never provided as input.

2.3 Dataset Property

Table 1b compares VenusRX-Bench with USPTO50K. VenusRX-Bench contains more reactant-side heavy atoms on average. The largest individual reactant is also larger on average. Thus, the increase reflects larger molecular structures, not simply more reactant components.

These structural differences affect reactant encoding in forward prediction and target generation in retrosynthesis. In particular, longer, multi-component retrosynthetic targets require models to generate larger molecular structures while maintaining valid representations. This motivates training directly on enzymatic reactions and constraining molecular generation, rather than assuming reliable transfer from conventional chemical reaction data. The paired structure-only and EC-conditioned settings separately examine how catalytic information guides prediction. These considerations motivate VenusRX in Section 3; their effects on transfer, prediction accuracy, and generation validity are assessed in the experiments.

3 VenusRX

VenusRX is a unified model for learning enzymatic reaction space. It formulates reaction modeling as shared text-to-text learning over forward prediction, retrosynthesis, and reaction reconstruction. To address the limited amount of experimentally observed enzymatic reactions, we first train the model on a large template-expanded reaction corpus and then continue training on real VenusRX-Bench reactions. VenusRX also supports optional Enzyme Commission (EC) conditioning, allowing catalytic information to guide reaction prediction when it is available. For retrosynthesis and forward prediction, we further study constrained generation to reduce the difficulty of generating large and complex biochemical molecules.

3.1 Unified Multitask Modeling

VenusRX uses a T5-style encoder–decoder architecture (Raffel et al., 2020b), with all parameters shared across three text-to-text objectives:

$$\begin{aligned} \text{Forward:} \quad & \texttt{<forward>}R \to P, \\ \text{Retrosynthesis:} \quad & \texttt{<retro>}P \to R, \\ \text{Span corruption:} \quad & \texttt{<mlm>}(\widetilde{R \gg P}) \to \text{masked spans}. \end{aligned}$$

The first two objectives learn both directions of an enzymatic transformation, while span corruption encourages shared reaction representations from the complete reaction context.

EC conditioning. Because molecular structures alone may not uniquely determine the intended enzymatic transformation, we additionally treat the EC number as an optional condition. For EC-annotated reactions, <ec_number>e is appended to the input with probability 0.5 during training. This allows the same model to operate with or without catalytic information at inference time. Detailed task construction and optimization settings are provided in Appendix C.

3.2 Training Procedure

VenusRX is trained in two stages. Stage-1 learns general enzymatic reaction patterns from a large template-expanded reaction corpus. Stage-2 then continues training on the real reactions in the corresponding VenusRX-Bench training split, adapting the pretrained model to the benchmark distribution.

Both stages use the same model architecture and multitask objectives. For each reaction, we additionally use R-SMILES augmentation (Zhong et al., 2022) and randomly sample one representation during training. This exposes the model to multiple sequence views of the same chemical reaction without changing the number of unique reactions.

3.2.1 Large-Scale Enzymatic Reaction Pretraining

Experimentally observed enzymatic reactions are limited in both scale and substrate diversity. To expand the reaction space available for pretraining, we construct a Stage-1 corpus containing about four million reaction records.

The corpus is built in four steps:

1. **Template extraction.** We extract reaction templates from atom-mapped enzymatic reactions. Each template captures the local structural transformation around the reaction center and can be reused with new substrates.
2. **Candidate substrate collection.** We assemble a large molecule library from biochemical and natural product databases, standardize molecular structures, and remove duplicated or unsuitable candidates.
3. **Template-based reaction generation.** For each template, we identify structurally compatible substrates and apply the reaction rule to generate new reactant–product pairs. This expands known enzymatic transformations to a broader range of substrate contexts.
4. **Reaction filtering.** Generated reactions are checked for structural validity, chemical feasibility, and duplication. To prevent data leakage, we further remove reactions overlapping with the validation or test set of each benchmark split before pretraining.

This process expands the training data along the chemical dimension, while R-SMILES augmentation provides multiple sequence representations of each retained reaction. The two augmentations therefore play complementary roles: template-based generation increases reaction diversity, whereas R-SMILES augmentation increases representation diversity.

Full details of molecule sources, template generation, filtering, and split-specific processing are provided in Appendix D.

3.3 Molecule Library-Constrained Decoding

Enzymatic reactions in VenusRX-Bench often involve larger and more complex molecules than conventional organic reaction datasets (Table 1b), making autoregressive generation more prone to invalid outputs. To address this issue, we introduce **Molecule Library-Constrained Decoding (MLCD)**, which restricts molecular generation to a predefined library of valid molecules.

MLCD builds a prefix tree (trie) over tokenized molecules in the library. During beam search, only next tokens that remain consistent with at least one library molecule are allowed. For reactions containing multiple molecules, the constraint is applied independently to each generated molecule.

The trie constrains the molecular search space, while VenusRX still scores and selects candidates according to the input reaction context. Detailed decoding procedures and pseudocode are provided in Appendix G.

4 Experiments

4.1 Experimental Setup

We evaluate forward prediction, single-step retrosynthesis, and EC-number prediction using the tasks, splits, and metrics defined by VenusRX-Bench. All VenusRX experiments use the same 25.1M-parameter configuration. For reaction generation, we report multi-reference Top-k accuracy, while weighted F1 is the primary metric for EC prediction. Results with and without EC conditioning are reported separately. The best result is shown in **bold** and the second-best result is underlined.

Baselines. We compare VenusRX with representative methods for conventional chemical reaction modeling, enzymatic reaction modeling, and EC prediction. Trainable baselines are adapted to the corresponding VenusRX-Bench training splits, while zero-shot methods use their released pretrained checkpoints. Baseline-specific training and inference details are provided in the appendix.

EC prediction. We evaluate VenusRX representations in both frozen and task-specific settings. The frozen setting directly uses pretrained reaction representations for EC retrieval, while the task-specific setting replaces CLAIRE's reaction features with the same frozen VenusRX representations and retrains its downstream predictor.

Leakage control and round-trip evaluation. For each benchmark split, we remove reactions overlapping with its validation or test sets from the Stage-1 corpus before pretraining. For round-trip evaluation, we use a fixed VenusRX forward oracle trained on the complete benchmark data and measure whether retrosynthetic candidates can recover the target product. The oracle is used only for evaluation and does not affect retrosynthetic generation. Further implementation details are provided in the appendix.

4.2 Results

4.2.1 Generalization across Enzymatic Reaction Space

Tables 2 and 3 evaluate reaction prediction across the Reactant, Product, and EC Splits. For forward prediction, VenusRX achieves consistently strong performance across all three settings, with particularly clear gains on the Product and EC Splits. Similar improvements are observed in retrosynthesis, where VenusRX generalizes effectively across changes in molecular targets and catalytic functions. Molecule Library-Constrained Decoding (MLCD) provides further gains in both prediction directions.

These results show that domain-specific pretraining transfers across multiple dimensions of enzymatic reaction space rather than being tied to a particular reaction distribution. They also demonstrate the value of explicitly modeling the structural characteristics of enzymatic reactions during generation.

4.2.2 Catalytic Function and Reaction Structure Are Coupled

Comparing the EC-conditioned and structure-only settings in Tables 2 and 3 shows that catalytic information consistently improves reaction prediction. The benefit is especially clear in retrosynthesis,

Table 2: Forward reaction prediction results on three data splits. We report Top-k exact-match accuracy (%). Methods marked with * are evaluated using open-source checkpoints without fine-tuning.

Method	Reactant Split				Product Split				EC Split			
	T1	T3	T5	T10	T1	T3	T5	T10	T1	T3	T5	T10
Zero-shot												
R-SMILES* (Zhong et al., 2022)	0.000	0.711	0.979	1.841	0.000	0.683	1.354	2.313	0.000	0.324	0.923	1.594
Without EC Input												
R-SMILES	70.562	82.048	85.543	88.618	30.849	44.631	50.000	55.849	15.568	28.104	34.899	44.643
Graph2SMILES (Tu & Coley, 2022)	68.453	80.219	82.863	84.576	34.408	44.427	48.286	51.702	19.823	29.794	33.401	37.548
MolTransformer (Schwaller et al., 2019)	52.796	66.344	71.307	75.221	12.776	22.471	26.330	29.051	9.396	15.808	18.456	21.560
MEGAN (Sacha et al., 2021)	28.285	39.247	44.676	46.202	19.128	24.676	26.366	27.421	7.646	12.308	14.166	14.801
VenusRX	73.066	84.611	88.048	90.739	50.959	66.275	71.393	77.085	31.100	45.170	53.236	59.947
VenusRX (MLCD)	**77.074**	**88.350**	**91.414**	**94.210**	**58.785**	**74.413**	**79.123**	**84.552**	**39.274**	**56.328**	**64.753**	**74.161**
With EC Input												
ECReact (Probst et al., 2022a)	61.440	74.080	78.285	81.885	37.812	51.558	55.908	59.456	26.246	40.400	45.602	51.989
VenusRX	75.221	86.778	89.725	92.474	55.825	69.463	74.041	78.811	34.564	50.240	56.340	64.058
VenusRX (MLCD)	**80.056**	**91.135**	**93.977**	**95.515**	**62.596**	**77.193**	**81.556**	**85.918**	**42.917**	**61.182**	**69.128**	**77.661**

Table 3: Retrosynthesis results on three data splits. We report Top-k exact-match accuracy (%). Methods marked with * are evaluated using open-source checkpoints without fine-tuning.

Method	Reactant Split				Product Split				EC Split			
	T1	T3	T5	T10	T1	T3	T5	T10	T1	T3	T5	T10
Zero-shot												
SciReasoner* (LLM) (Wang et al., 2025)	1.445	1.957	2.528	3.355	0.947	1.570	2.121	2.876	0.204	0.611	0.839	1.282
R-SMILES*	0.000	0.687	2.004	2.656	0.000	1.007	2.073	2.601	0.000	0.228	0.527	0.899
RSGPT* (Deng et al., 2025b)	0.058	0.082	0.128	0.373	0.000	0.048	0.132	0.204	0.024	0.048	0.192	0.527
Without EC Input												
Graph2SMILES	7.025	11.708	15.040	18.802	**27.397**	34.192	36.925	39.022	3.320	4.626	5.693	6.388
Retroformer (Wan et al., 2022)	6.710	12.488	16.170	22.938	9.863	12.800	15.688	18.253	1.558	2.601	3.248	4.219
MEGAN	5.510	9.914	12.022	15.645	10.127	14.190	17.581	19.343	0.683	1.534	2.061	2.469
R-SMILES	5.359	9.425	12.127	16.146	4.147	7.299	9.084	11.314	1.174	3.667	4.986	6.364
BioNavi (Zheng et al., 2022)	2.901	5.440	8.889	13.292	9.767	15.484	18.552	21.632	1.007	1.642	1.930	2.900
G2Retro (Guo et al., 2023)	1.351	2.365	2.971	5.720	2.385	5.177	7.227	9.012	0.503	0.923	1.594	1.822
RSGPT	2.854	7.712	10.869	16.181	8.497	15.580	18.816	23.730	1.151	2.601	3.679	6.628
VenusRX	8.446	17.183	23.777	30.603	20.554	30.058	34.923	39.945	4.878	8.737	10.487	14.274
VenusRX (MLCD)	**9.448**	**19.070**	**26.584**	**34.355**	22.903	**34.636**	**40.244**	**46.093**	**7.407**	**14.657**	**18.384**	**23.849**
With EC Input												
ECReact	11.149	17.568	21.226	25.757	17.282	24.521	27.049	27.457	5.837	8.689	10.606	11.086
VenusRX	17.230	26.398	29.345	34.425	28.847	34.696	38.267	40.568	18.360	24.605	27.481	30.872
VenusRX (MLCD)	**17.300**	**27.400**	**32.258**	**38.828**	**29.302**	**39.525**	**43.672**	**48.466**	**20.374**	**27.505**	**30.609**	**34.552**

where EC annotations provide useful context for identifying plausible transformations from a target product.

The reverse relationship is also observed. As shown in Table 4, frozen VenusRX reaction representations encode sufficient catalytic information to support effective EC prediction, while using the same representations in a task-specific predictor further improves performance across the EC hierarchy.

Together, these results reveal a bidirectional relationship between reaction structure and catalytic function: catalytic information guides reaction prediction, while reaction representations recover catalytic information. This empirical coupling supports our formulation of enzymatic reaction space as a joint relationship among reactants, products, and catalytic functions.

4.2.3 Consistency and Constrained Generation

MLCD consistently improves exact-match performance across forward prediction and retrosynthesis, showing that constraining the molecular output space helps the model recover benchmark targets more reliably. The validity analysis in Appendix G.2 further shows that MLCD substantially reduces invalid molecular generation, supporting its effectiveness for decoding large and complex biochemical structures.

Table 4: EC-number prediction on the Reactant and Product splits. Weighted F1 (%) is reported at EC levels 1–4.

Method	Reactant Split				Product Split			
	EC1	EC2	EC3	EC4	EC1	EC2	EC3	EC4
Zero-Shot								
crxnfp (Zhang et al., 2025)	92.38	85.73	80.90	46.54	89.15	80.83	74.35	41.90
DRFP (Probst et al., 2022b)	92.57	85.75	81.21	49.40	91.48	**84.10**	**77.84**	44.61
rxnfp (Schwaller et al., 2021b)	92.11	85.59	80.52	47.11	88.88	79.80	73.03	41.09
VenusRX*	**94.76**	**87.76**	**82.81**	**51.17**	**92.11**	83.80	77.68	**47.27**
Task Specific Model								
Theia (Probst, 2023)	93.09 ± 0.28	87.72 ± 0.46	83.68 ± 0.24	**52.49** ± 0.26	92.48 ± 0.47	84.86 ± 0.38	79.33 ± 0.45	**47.44** ± 0.36
CLAIRE (Zeng et al., 2025)	94.88 ± 0.14	89.36 ± 0.27	84.14 ± 0.40	49.49 ± 0.96	92.41 ± 0.62	86.18 ± 0.30	78.86 ± 0.51	43.50 ± 0.60
BEC-Pred (Qian et al., 2024)	96.16 ± 0.33	89.78 ± 0.14	81.45 ± 0.50	46.79 ± 0.43	94.67 ± 0.36	87.02 ± 0.99	76.13 ± 0.60	42.14 ± 0.76
VenusRX	**96.45** ± 0.11	**90.99** ± 0.16	**86.03** ± 0.24	51.33 ± 0.59	**94.85** ± 0.33	**87.40** ± 0.22	**80.23** ± 0.64	47.02 ± 0.28

VenusRX denotes frozen VenusRX representations followed by a task-specific predictor. Task-specific results are mean ± std over three seeds.

Table 5: Ablation study on the Reactant Split. Reaction generation is evaluated with EC conditioning and without MLCD; EC prediction reports zero-shot weighted F1 (%).

Variant	Training Objectives			Stage-1	Stage-2	Forward (with EC)			Retrosynthesis (with EC)			EC Prediction			
	Fwd.	Retro.	MLM			Top-1	Top-5	Top-10	Top-1	Top-5	Top-10	EC1	EC2	EC3	EC4
Full	✓	✓	✓	✓	✓	**75.221**	**89.725**	**92.474**	**17.230**	**29.345**	**34.425**	**94.76**	**87.76**	**82.81**	**51.17**
w/o MLM	✓	✓	–	✓	✓	75.175	89.119	92.020	17.160	27.691	32.013	92.53	85.14	80.31	49.54
w/o Stage-1	✓	✓	✓	–	✓	47.192	71.086	77.178	7.130	17.882	22.682	89.86	80.78	75.49	45.17
w/o Stage-2	✓	✓	✓	✓	–	35.450	54.578	60.881	6.454	14.003	17.999	94.41	87.01	82.03	49.38

Round-trip evaluation in Appendix F.1 provides a complementary view of the generated candidate space. While unconstrained decoding retains a broader set of alternative hypotheses, MLCD focuses generation on molecules supported by the predefined library. This more targeted search improves recovery of recorded reaction targets and reduces structurally invalid outputs.

Overall, these results show that MLCD provides an effective inference-time strategy for navigating the large molecular output space of enzymatic reactions, improving both generation validity and target recovery.

4.2.4 Ablation Study

Table 5 examines the main components of VenusRX. The full two-stage training configuration provides the strongest overall performance, showing that large-scale reaction pretraining and benchmark-domain continued training play complementary roles. Stage-1 provides transferable enzymatic transformation knowledge, while Stage-2 adapts this knowledge to real benchmark reactions.

The span-corruption objective provides additional benefits, particularly for retrosynthesis and EC prediction. This suggests that reconstructive learning strengthens the shared reaction representation beyond the two directional generation tasks. Overall, the ablation results support the combination of large-scale pretraining, domain adaptation, and multitask learning used by VenusRX.

5 Related Work

5.1 Organic Reaction Modeling

Forward reaction prediction and retrosynthesis. A dominant line of work casts reaction modeling as SMILES-to-SMILES translation with Transformers. Molecular Transformer pioneered this paradigm for forward prediction (Schwaller et al., 2019), later extended to single-step retrosynthesis (Karpov et al., 2019); Retroformer further adds local attention over reactive topology (Wan et al., 2022), while R-SMILES aligns input-output roots via atom mapping to ease sequence learning (Zhong et al., 2022). In parallel, graph-based models encode molecular topology more explicitly: Graph2SMILES uses a permutation-invariant graph encoder with a SMILES decoder (Tu & Coley, 2022), whereas MEGAN and Graph2Edits model reactions as sequences of graph edits (Sacha et al., 2021; Zhong et al., 2023). These approaches provide useful architectures and baselines for our study.

Chemical reaction pretraining. To learn transferable reaction knowledge, recent work pretrains models on large chemical corpora before adapting them to downstream tasks. On the representation side, RXNFP learns reaction fingerprints via BERT-style masked language modeling for classification and reaction-space mapping (Schwaller et al., 2021b), while RxnRep uses chemically consistent contrastive pretraining for classification and regression (Wen et al., 2022). On the generative side, Chemformer applies BART-style pretraining for forward prediction and retrosynthesis (Irwin et al., 2022); ReactSeq introduces a reaction description language for retrosynthesis, retrieval, and yield prediction (Xiong et al., 2025); and RSGPT scales generative pretraining to ten-billion synthetic reactions for retrosynthesis planning (Deng et al., 2025b). Their strong performance motivates us to develop a pretrained model tailored specifically to enzymatic reactions.

5.2 Enzymatic Reaction Data and Modeling

Data resources and benchmarks. Existing efforts have constructed enzymatic reaction datasets and associated evaluation protocols. ECREACT aggregates EC-annotated reactions from Rhea, BRENDA, PathBank, and MetaNetX for forward prediction and retrosynthesis (Probst et al., 2022a). EnzymeMap provides a pipeline for correcting, validating, and atom-mapping enzymatic reactions, and evaluates the curated data on forward prediction, retrosynthesis, and regioselectivity (Heid et al., 2023). ReactZyme instead focuses on bidirectional enzyme-reaction retrieval, assessing generalization under enzyme and reaction similarity splits (Hua et al., 2024). CARE further introduces a benchmark for enzyme classification and retrieval, where Task 2 retrieves EC numbers from query reactions (Yang et al., 2024). While complementary, these resources differ substantially in task definitions, data processing, and curation quality, making them difficult to merge into a unified benchmark.

Bio-retrosynthesis and catalytic function prediction. Beyond dataset curation, several methods model enzymatic transformations by incorporating catalytic context into neural networks. Kreutter et al. adapt a Molecular Transformer via transfer learning, conditioning product prediction on textual enzyme descriptions (Kreutter et al., 2021). ECREACT instead encodes EC numbers as class tokens for forward prediction and jointly predicts reactants and EC classes in retrosynthesis (Probst et al., 2022a). BioNavi-NP trains Transformers on organic and biosynthetic reactions and couples single-step predictions with multi-step search for natural-product pathway planning (Zheng et al., 2022). For catalytic function annotation, CLAIRE employs contrastive learning to predict EC numbers for enzymatic reactions (Zeng et al., 2025). More recently, Enzyformer applies two-stage molecular and organic-reaction pretraining before fine-tuning for enzymatic retrosynthesis, together with a separate contrastive model for EC prediction (Liu et al., 2026a). Despite this progress, existing approaches are largely confined to task-specific supervised learning on individual datasets, and a unified pretraining framework with a comprehensive benchmark dedicated to enzymatic reactions remains lacking.

6 Conclusions and Future Work

We introduced VenusRX-Bench, a unified benchmark for enzymatic reaction modeling across forward prediction, retrosynthesis, and EC-number prediction, with controlled splits that evaluate generalization across reactants, products, and catalytic functions. Based on this benchmark, we developed VenusRX, a unified pretrained encoder–decoder model that combines large-scale template-expanded pretraining, continued training on real enzymatic reactions, optional EC conditioning, and molecule-library constrained decoding. Experiments across multiple tasks and splits show that domain-specific pretraining and catalytic context are important for modeling enzymatic reactions, while reaction representations themselves also encode useful catalytic information.

Several directions remain for future work. First, EC numbers provide only a coarse description of catalytic function; incorporating enzyme sequences, structures, cofactors, and reaction conditions may enable more precise modeling of enzyme-dependent reaction outcomes. Second, the current benchmark focuses on single-step reactions, and extending the framework to multi-step biosynthetic pathway planning would provide a more complete evaluation of biocatalytic synthesis. Finally, constrained decoding currently relies on a predefined molecular library. Developing generation methods that preserve chemical validity while allowing exploration beyond a fixed candidate space may further improve generalization to previously unseen biochemical molecules.

AI USE STATEMENT

AI was used to provide feedback on experimental design, assist with code implementation and checking, conduct related-work research, and support manuscript drafting. The correctness of the implementation and experimental results was independently verified by the authors. The authors take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT.

We provide detailed descriptions of the benchmark construction, data splits, model architecture, training procedure, inference settings, and evaluation protocols in the main text and appendix. Hyperparameters and implementation details necessary to reproduce the reported experiments are documented where applicable. Upon acceptance, we will publicly release the source code, benchmark processing pipeline, evaluation scripts, and model checkpoints to facilitate reproducibility and future research.

REFERENCES

Baudoin Delépine, Thomas Duigou, Pablo Carbonell, and Jean-Loup Faulon. Retropath2.0: A retrosynthesis workflow for metabolic engineers. *Metabolic Engineering*, 45:158–170, 2018. doi: 10.1016/j.ymben.2017.12.002.

Yafeng Deng, Xinda Zhao, Hanyu Sun, Yu Chen, Xiaorui Wang, Xi Xue, Liangning Li, Jianfei Song, Chang-Yu Hsieh, Tingjun Hou, Xiandao Pan, Taghrid Saad Alomar, Xiangyang Ji, and Xiaojian Wang. RSGPT: a generative transformer model for retrosynthesis planning pre-trained on ten billion datapoints. *Nature Communications*, 16:7012, 2025a. doi: 10.1038/s41467-025-62308-6.

Yafeng Deng, Xinda Zhao, Hanyu Sun, Yu Chen, Xiaorui Wang, Xi Xue, Liangning Li, Jianfei Song, Chang-Yu Hsieh, Tingjun Hou, Xiandao Pan, Taghrid Saad Alomar, Xiangyang Ji, and Xiaojian Wang. Rsgpt: a generative transformer model for retrosynthesis planning pre-trained on ten billion datapoints. *Nature Communications*, 16(1):7012, 2025b. doi: 10.1038/s41467-025-62308-6. URL https://doi.org/10.1038/s41467-025-62308-6.

William Finnigan, Lorna J. Hepworth, Sabine L. Flitsch, and Nicholas J. Turner. Retrobiocat as a computer-aided synthesis planning tool for biocatalytic reactions and cascades. *Nature Catalysis*, 4(2):98–104, 2021. doi: 10.1038/s41929-020-00556-z.

Yves Grandjean, Sacha Raffaud, Annie M. Westerlund, Philippe Schwaller, Samuel Genheden, and Jean-Louis Reymond. Rxnmapperv2: updated validation and evaluation on expanded chemical space. *ChemRxiv*, 2026(0626), 2026. doi: 10.26434/chemrxiv.15005247/v1. URL https://chemrxiv.org/doi/abs/10.26434/chemrxiv.15005247/v1.

Zhongliang Guo, Yan Wu, Ke Liu, Yuxin Sun, Yujing Yang, Peng Yang, Jun Wang, Yidan Zhang, Jian Wang, and Wen Du. G2Retro as a two-step graph generative models for retrosynthesis prediction. *Communications Chemistry*, 6:102, 2023. doi: 10.1038/s42004-023-00897-3.

Esther Heid, Daniel Probst, William H. Green, and Georg K. H. Madsen. Enzymemap: Curation, validation and data-driven prediction of enzymatic reactions. *Chemical Science*, 14(48):14229–14242, 2023. doi: 10.1039/D3SC02048G.

Chenqing Hua, Bozitao Zhong, Sitao Luan, Liang Hong, Guy Wolf, Doina Precup, and Shuangjia Zheng. ReactZyme: A benchmark for enzyme-reaction prediction. In *Advances in Neural Information Processing Systems*, volume 37, pp. 26415–26442, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/2e68b2367d2e0bc8dd6f0ff86e07c2eb-Abstract-Datasets_and_Benchmarks_Track.html.

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

Pavel Karpov, Guillaume Godin, and Igor V. Tetko. A transformer model for retrosynthesis. In *Artificial Neural Networks and Machine Learning – ICANN 2019: Workshop and Special Sessions*, volume 11731 of *Lecture Notes in Computer Science*, pp. 817–830, Cham, 2019. Springer. ISBN 978-3-030-30492-8. doi: 10.1007/978-3-030-30493-5_78.

Steven M. Kearnes, Michael R. Maser, Michael Wleklinski, Anton Kast, Abigail G. Doyle, Spencer D. Dreher, Joel M. Hawkins, Klavs F. Jensen, and Connor W. Coley. The open reaction database. *Journal of the American Chemical Society*, 143(45):18820–18826, 2021. doi: 10.1021/jacs.1c09820. URL https://doi.org/10.1021/jacs.1c09820.

David Kreutter, Philippe Schwaller, and Jean-Louis Reymond. Predicting enzymatic reactions with a molecular transformer. *Chemical Science*, 12(25):8648–8659, 2021. doi: 10.1039/D1SC02362D.

Tiantao Liu, Jiangcheng Xu, Xinke Zhan, Shaolong Lin, and Shirley W. I. Siu. Enzyformer: a two-stage pretrained model for enzymatic retrosynthesis. *Journal of Cheminformatics*, 18:35, 2026a. doi: 10.1186/s13321-026-01164-y.

Yong Liu, Chenqing Hua, Minghao Xu, Tao Zeng, Jiahua Rao, Zhaoyang Zhang, others, and Shuangjia Zheng. A geometric foundation model for enzyme retrieval with evolutionary insights. *Nature Catalysis*, 9(2):148–160, 2026b. doi: 10.1038/s41929-026-01478-y.

Daniel Mark Lowe. *Extraction of Chemical Structures and Reactions from the Literature*. PhD thesis, University of Cambridge, 2012.

Jens Nielsen and Jay D. Keasling. Engineering cellular metabolism. *Cell*, 164(6):1185–1197, 2016. doi: 10.1016/j.cell.2016.02.004.

Daniel Probst. Explainable prediction of catalysing enzymes from reactions using multilayer perceptrons. *bioRxiv*, 2023. doi: 10.1101/2023.01.28.526009.

Daniel Probst, Matteo Manica, Yves Gaetan Nana Teukam, Alessandro Castrogiovanni, Federico Paratore, and Teodoro Laino. Biocatalysed synthesis planning using data-driven learning. *Nature Communications*, 13(1):964, 2022a. doi: 10.1038/s41467-022-28536-w.

Daniel Probst, Philippe Schwaller, and Jean-Louis Reymond. Reaction classification and yield prediction using the differential reaction fingerprint DRFP. *Digital Discovery*, 1(2):91–97, 2022b. doi: 10.1039/D1DD00006C.

Wenjia Qian, Xiaorui Wang, Yu Kang, et al. A general model for predicting enzyme functions based on enzymatic reactions. *Journal of Cheminformatics*, 16(1):38, 2024. doi: 10.1186/s13321-024-00829-x.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020a.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020b. URL https://jmlr.org/papers/v21/20-074.html.

Mikołaj Sacha, Mikołaj Błaż, Piotr Byrski, Paweł Dabrowski-Tumański, Mikołaj Chromiński, Rafał Loska, Paweł Włodarczyk-Pruszyński, and Stanisław Jastrzebski. Molecule edit graph attention network: Modeling chemical reactions as sequences of graph edits. *Journal of Chemical Information and Modeling*, 61(7):3273–3284, 2021. doi: 10.1021/acs.jcim.1c00537.

Philippe Schwaller, Teodoro Laino, Théophile Gaudin, Peter Bolgar, Christopher A. Hunter, Costas Bekas, and Alpha A. Lee. Molecular transformer: A model for uncertainty-calibrated chemical reaction prediction. *ACS Central Science*, 5(9):1572–1583, 2019. doi: 10.1021/acscentsci.9b00576.

Philippe Schwaller, Benjamin Hoover, Jean-Louis Reymond, Hendrik Strobelt, and Teodoro Laino. Extraction of organic chemistry grammar from unsupervised learning of chemical reactions. *Science Advances*, 7(15):eabe4166, 2021a. doi: 10.1126/sciadv.abe4166.

Philippe Schwaller, Daniel Probst, Alain C. Vaucher, Vishnu H. Nair, David Kreutter, Teodoro Laino, and Jean-Louis Reymond. Mapping the space of chemical reactions using attention-based neural networks. *Nature Machine Intelligence*, 3(2):144–152, 2021b. doi: 10.1038/s42256-020-00284-w.

Roger A. Sheldon and John M. Woodley. Role of biocatalysis in sustainable chemistry. *Chemical Reviews*, 118(2):801–838, 2018. doi: 10.1021/acs.chemrev.7b00203.

Zhengkai Tu and Connor W. Coley. Permutation invariant graph-to-sequence model for template-free retrosynthesis and reaction prediction. *Journal of Chemical Information and Modeling*, 62(15): 3503–3513, 2022. doi: 10.1021/acs.jcim.2c00321.

Nicholas J. Turner and Elaine O'Reilly. Biocatalytic retrosynthesis. *Nature Chemical Biology*, 9(5): 285–288, 2013. doi: 10.1038/nchembio.1235.

Yue Wan, Chang-Yu Hsieh, Benben Liao, and Shengyu Zhang. Retroformer: Pushing the limits of end-to-end retrosynthesis transformer. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 22475–22490. PMLR, 2022. URL https://proceedings.mlr.press/v162/wan22a.html.

Yizhou Wang, Chen Tang, Han Deng, Jiabei Xiao, Jiaqi Liu, Jianyu Wu, Jun Yao, Pengze Li, Encheng Su, Lintao Wang, Guohang Zhuang, Yuchen Ren, Ben Fei, Mingmao Hu, Xin Chen, Dongzhan Zhou, Junjun He, Xiangyu Yue, Zhenfei Yin, Jiamin Wu, Qihao Zheng, Yuhao Zhou, Huihui Xu, Chenglong Ma, Yan Lu, Wenlong Zhang, Chunfeng Song, Philip H. S. Torr, Shixiang Tang, Xinzhu Ma, Wanli Ouyang, and Lei Bai. SciReasoner: Laying the scientific reasoning ground across disciplines. *arXiv preprint arXiv:2509.21320*, 2025. doi: 10.48550/arXiv.2509.21320.

Y. Wen, Z. Li, X. Xiang, D. Zhang, and S. Kim. Improving machine learning performance on small chemical reaction data with unsupervised contrastive pretraining. *Chemical Science*, 13 (5):1448–1459, 2022. doi: 10.1039/D1SC05469D. URL https://doi.org/10.1039/D1SC05469D.

Jiacheng Xiong, Wei Zhang, Yinquan Wang, Jiatao Huang, Yuqi Shi, Mingyan Xu, Manjia Li, Zunyun Fu, Xiangtai Kong, Yitian Wang, Zhaoping Xiong, and Mingyue Zheng. Bridging chemistry and artificial intelligence by a reaction description language. *Nature Machine Intelligence*, 7: 782–793, 2025. doi: 10.1038/s42256-025-01032-8. URL https://doi.org/10.1038/s42256-025-01032-8.

Jason Yang, Ariane Mora, Shengchao Liu, Bruce J. Wittmann, Anima Anandkumar, Frances H. Arnold, and Yisong Yue. CARE: A benchmark suite for the classification and retrieval of enzymes. In *Advances in Neural Information Processing Systems*, volume 37, pp. 3094–3121, 2024. doi: 10.52202/079017-0101. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/05a7ad45d75a3082d7a3a70de8743140-Abstract-Datasets_and_Benchmarks_Track.html.

Zishuo Zeng, Jin Guo, Jiao Jin, and Xiaozhou Luo. CLAIRE: a contrastive learning-based predictor for EC number of chemical reactions. *Journal of Cheminformatics*, 17:2, 2025. doi: 10.1186/s13321-024-00944-8.

Kunyang Zhang, Thierry D. Marti, Silke I. Probst, Serina L. Robinson, and Kathrin Fenner. Enzyme association for environmental biotransformation reactions through contrastive learning of reaction center-specific fingerprints. *Bioinformatics*, 2025. doi: 10.1093/bioinformatics/btag142.

Shuangjia Zheng, Tao Zeng, Chengtao Li, Binghong Chen, Connor W. Coley, Yuedong Yang, Ruibo Wu, et al. Deep learning driven biosynthetic pathways navigation for natural products with BioNavi-NP. *Nature Communications*, 13(1):3342, 2022. doi: 10.1038/s41467-022-30970-9.

Weihe Zhong, Ziduo Yang, and Calvin Yu-Chian Chen. Retrosynthesis prediction using an end-to-end graph generative architecture for molecular graph editing. *Nature Communications*, 14:3009, 2023. doi: 10.1038/s41467-023-38851-5.

Zipeng Zhong, Jie Song, Zunlei Feng, Tiantao Liu, Lingxiang Jia, Shaolun Yao, Min Wu, Tingjun Hou, and Mingli Song. Root-aligned SMILES: a tight representation for chemical reaction prediction. *Chemical Science*, 13(31):9023–9034, 2022. doi: 10.1039/D2SC02763A.

A Benchmark Construction Details

This section provides the construction details of VenusRX-Bench. The pipeline consists of four stages: multi-source harmonization, structural processing and deduplication, single-product curation, and leakage-controlled data partitioning.

A.1 Multi-source Harmonization

We collect enzymatic reactions with EC annotations from BRENDA, PathBank, Rhea, and KEGG. Because these databases differ in record format and reaction representation, all entries are first converted into a common schema containing the source identifier, EC annotation, reaction text, substrates, products, and molecular structures when available.

Source-specific representations are normalized before merging. Positive integer stoichiometric coefficients in PathBank are expanded into repeated molecular components, Rhea component separators are converted into the common reaction format, and the balanced reaction-level structures provided by KEGG are retained directly. Reaction text is normalized by standardizing arrow notation and whitespace. EC annotations are cleaned only at the representation level; missing EC numbers are removed rather than inferred or completed.

We first deduplicate records using the joint key of normalized reaction text and EC annotation. Thus, records describing the same reaction with different EC annotations remain distinct. Starting from 310,209 source records with valid EC annotations, this procedure yields 89,605 reaction–EC records, corresponding to 80,558 unique reaction texts.

A.2 Structural Processing and Deduplication

For records represented by compound names, molecular structures are resolved through a shared name-to-SMILES mapping and converted into structured reactions. Reaction balancing is then applied where possible. Records that already provide a balanced reaction-level representation, such as the retained KEGG reactions, are used directly.

Atom mapping is generated with RXNMapper. When a balanced reaction candidate is available, mapping is performed on that candidate; otherwise, a structurally valid reaction may be mapped as a fallback. Records for which no usable mapped reaction can be obtained are removed from subsequent construction.

A reaction can admit multiple mapped candidates. For candidate mapping m, we define the edit score as

$$S(m) = n_{\mathrm{remove}}(m) + n_{\mathrm{add}}(m) + n_{\mathrm{atom}}(m), \tag{1}$$

where n_{remove} and n_{add} denote the numbers of removed and added bonds, respectively, and n_{atom} denotes the number of atoms whose local environments change. Candidates with the minimum score are retained. If several candidates attain the same minimum score, all tied candidates are preserved rather than selecting one arbitrarily.

We subsequently canonicalize every molecule while retaining atom-map information, sort molecular components independently on the two sides of the reaction, and reconstruct the normalized mapped reaction. This removes differences caused only by component ordering. Structure-level deduplication is then performed using the EC annotation together with the normalized mapped reaction representation.

A.3 Single-product Curation

Many biochemical reactions contain multiple products, cofactors, or small by-products and therefore do not directly define a unique target for reaction prediction. Following the single-product curation protocol used by ECReact (Probst et al., 2022a), we construct one principal product for each retained reaction.

Specifically, we apply the following rules:

1. remove products that also occur on the reactant side of the same reaction;

2. for reactions containing multiple products, remove known coenzymes and common by-products;
3. remove product molecules containing fewer than four heavy atoms; and
4. discard reactions that retain zero or multiple products, or contain no reactants after curation.

The coenzyme and by-product filtering follows the ECReact curation lists, including common phosphate- and nucleotide-related species, NAD-related cofactors, coenzyme A, SAM, flavins, heme, iron–sulfur clusters, acetate, and pyruvate. This filtering is applied only when the original reaction contains multiple products, so that a reaction whose sole product belongs to one of these categories is not removed solely by this rule.

We additionally require consistency between the name-level, unmapped-structure, and mapped-structure representations. The mapped and unmapped versions therefore contain exactly the same examples in the same order and differ only in the presence of atom-map indices. After structural processing, product curation, and final single-product deduplication, the benchmark contains **83,439** reaction examples.

A.4 Leakage-controlled Data Splits

All three benchmark splits are constructed from the same 83,439 single-product reactions. Rather than independently assigning individual rows, we first define canonical structural keys. Atom maps are removed, each molecule is canonicalized, molecular components are sorted, and the two reaction sides are represented as

$$k_R = \mathrm{canon}(R), \tag{2}$$

$$k_P = \mathrm{canon}(P), \tag{3}$$

$$k_{\mathrm{rxn}} = k_R \parallel >> \parallel k_P. \tag{4}$$

We additionally use normalized reaction text and the hierarchical EC annotation as grouping keys.

Samples sharing a constrained key are grouped into indivisible connected components. Each component is assigned wholly to the training, validation, or test partition, targeting an approximately $80/10/10$ ratio. This prevents records connected by the corresponding leakage criterion from appearing in different partitions.

We construct three complementary splits:

Reactant Split. Samples sharing the same canonical reactant set or normalized reactant text are kept in the same partition. This prevents identical substrate combinations and identical reactant descriptions from crossing partitions.

Product Split. In addition to the Reactant Split constraints, samples sharing the same canonical product are kept in the same partition. The test set therefore evaluates prediction on target products that are not present in the training partition.

EC Split. Samples are grouped according to their complete four-level EC annotation together with the canonical full reaction. Consequently, an EC class assigned to the validation or test partition does not occur in the training partition, while identical complete reactions are also prevented from crossing partitions.

Because complete connected components are assigned jointly, the final partition sizes can deviate slightly from the nominal $80/10/10$ ratio. The resulting split sizes are shown in Table 6.

A.5 Mapped and Unmapped Representations

For each partition, we provide both mapped and unmapped reaction representations. The two versions share the same examples, filtering decisions, and row ordering. The unmapped representation is used by models that operate directly on molecular strings, whereas the mapped representation supports methods that require atom correspondence for reaction-center identification or graph-edit supervision. This paired construction ensures that models requiring different reaction representations are evaluated on exactly the same underlying examples. “‘

Table 6: Statistics of the three leakage-controlled benchmark splits. All splits are constructed from the same 83,439 curated reactions.

Split	Train	Validation	Test
Reactant Split	66,672	8,183	8,584
Product Split	66,751	8,344	8,344
EC Split	66,751	8,344	8,344

B EC PREDICTION

B.1 EC PREDICTION: IMPLEMENTATION DETAILS

Data and label spaces. Reaction SMILES are standardized with RDKit, while preserving one complete reaction per record and retaining all products. The supplied validation partitions are used without resplitting. After label normalization, the training vocabularies contain 7, 67, 230, and 4,074 classes at EC1–EC4 for the Reactant Split, and 7, 67, 231, and 3,943 classes for the product split. Labels appearing only in validation or test data are not added to the prediction candidates. Test examples with unseen labels are retained in the reported evaluation.

Pretrained checkpoints. The VenusRX model has 25.1M parameters, hidden size 384, and six layers in each of the encoder and decoder. EC prediction uses the encoder only. We use the final stage-two checkpoints after 80,100 and 72,100 steps for the reaction and product splits, respectively, rather than validation-best checkpoints. Stage-two pretraining uses the corresponding training partition with 20-fold R-SMILES augmentation. The sampling weights for forward prediction, retrosynthesis, and span corruption are 0.35, 0.35, and 0.30, respectively. EC prompting is enabled with probability 0.5, and the EC contrastive objective is disabled. Thus, the downstream zero-shot setting does not imply that EC annotations were absent during pretraining.

Reaction representations and zero-shot retrieval. For a reaction r, let $x_r^{(v)}$ be its v-th complete R-SMILES view with the <mlm> prefix. The frozen encoder E_θ produces the reaction representation

$$\mathbf{z}(r) = \frac{1}{10}\sum_{v=1}^{10} \mathrm{MeanPool}\Big(E_\theta(x_r^{(v)})\Big). \tag{5}$$

Pooling first averages token embeddings within a view and then averages the 10 view-level vectors. The same precomputed vectors are used by VenusRX * and VenusRX. The 10-view representation extraction is distinct from the 20-fold augmentation used during pretraining. VenusRX * uses cosine k-nearest-neighbor retrieval with labeled training reactions as its reference database. The value of k is selected on validation data, without using test scores. DRFP and rxnfp also use cosine k-nearest-neighbor retrieval without downstream parameter training, with k selected on validation data. crxnfp uses its released encoder weights without downstream training and performs 1-nearest-neighbor retrieval against the training set.

CLAIRE and VenusRX. CLAIRE uses its original concatenated DRFP and rxnfp features, with DRFP folded to 256 dimensions. VenusRX replaces these inputs with $\mathbf{z}(r)$ while keeping the VenusRX encoder frozen. The original CLAIRE implementation is reused for projection training; only feature construction is replaced. Both methods use the same threefold permutation-augmented training set and train a separate projection network for each EC level. The five-layer projection network has hidden width 1,280, output dimension 128, LayerNorm, and dropout 0.1. Training uses a learning rate of 5×10^{-4}, batch size 6,000, and triplet margin 1.0. Both methods train for a fixed 2,000 epochs and select the checkpoint with the lowest training loss within the final 20% of epochs. This checkpoint-selection rule does not use validation scores or validation-based early stopping. Retrieval uses Euclidean distances in the learned projection space.

Other trained baselines. Theia's classifier and CLAIRE's projection network are randomly initialized and trained on the corresponding training partition, rather than initialized from their released task-trained weights. Theia uses 10,240-bit DRFP features with radius 2 and a single hidden layer of

1,664 units with Tanh activation. It is trained with Adam, learning rate 10^{-3}, batch size 64, and an exponential learning-rate decay factor of 0.9. Multi-label training records are expanded for this baseline. Its official stopping rule requires the current validation loss to improve by at least 10^{-3} relative to the mean of the preceding five epochs, with a maximum of 50 epochs. BEC-Pred is initialized from the official rxnfp `bert_pretrained` weights and trained for 48 epochs, using batch size 8, learning rate 10^{-5}, maximum sequence length 512, and no warmup. Where paper descriptions and released code differ, we follow the official code configurations. CLAIRE is additionally evaluated at EC4, and BEC-Pred at EC1, EC2, and EC4, by adapting the label space while keeping their training configurations unchanged. BEC-Pred retains its raw-string label indexing; for reaction-split EC4, this produces 4,081 output classes before normalization, compared with 4,074 normalized benchmark labels.

Evaluation and strict-subset analysis. All methods are scored with the same `score_ec.py` implementation and scoring parameters. Let $\mathcal{Y}_{i,\ell}$ be the set of reference labels for reaction i at EC level ℓ, and $\hat{y}_{i,\ell}$ the top-ranked prediction. The hit-any top-1 accuracy is

$$\mathrm{Acc}_\ell = \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \mathbf{1}[\hat{y}_{i,\ell} \in \mathcal{Y}_{i,\ell}] \,, \tag{6}$$

where N_ℓ is the number of records evaluated at level ℓ. Weighted F1 is the primary metric, with macro F1 and top-1 accuracy also reported. The supplementary strict-subset scores exclude 278 reaction-split test records and 18 product-split test records flagged as input-structure-equivalent to training reactions. These records remain in the main test evaluation. Strict-subset filtering uses the `test_leaked_ids_input_only` flags in `leakage.json`. Theia, CLAIRE, BEC-Pred, and VenusRX are run with seeds 13, 21, and 42, and their results are summarized by the mean and standard deviation. crxnfp, DRFP, rxnfp, and VenusRX * are evaluated as deterministic methods and have no across-seed standard deviation.

C VenusRX Implementation Details

C.1 Architecture and Reaction Representation

Architecture and initialization. The T5-style model shares input embeddings across the encoder and decoder and ties the output embedding to them. Table 7 lists the configurations used for reaction-split scaling; Tiny is used for cross-split comparisons. All configurations use ReLU feed-forward layers, dropout 0.1, 32 relative-attention buckets, a maximum relative-distance parameter of 128, and layer-normalization epsilon 10^{-6}. The relative-distance parameter is not a maximum sequence length. The decoder starts with padding token ID 0; EOS has ID 1.

Table 7: VenusRX configurations. Parameter counts are recorded wrapper totals, including the inactive projection head.

Model	Enc./Dec.	d_{model}	d_{ff}	Heads	d_{kv}	Parameters
Micro	4/4	256	1,024	4	64	7,554,816
Compact	6/6	320	1,280	5	64	17,495,296
Tiny	6/6	384	1,536	6	64	25,147,648
Medium	6/6	768	3,072	12	64	100,135,168

The wrapper also instantiates a two-layer projection head, $d_{\mathrm{model}} \to d_{\mathrm{model}} \to 256$, with a tanh activation. Its contrastive-loss coefficient is zero, so it does not contribute to the sequence-training objective. For Tiny, this head contains 246,400 parameters and the T5 backbone contains 24,901,248 parameters. No enzyme-sequence encoder or additional EC-prediction loss is used in this training protocol.

Tokenizer and length limits. Both stages and standard inference load the same fixed 302-token vocabulary. A custom regular expression identifies SMILES tokens, including bracketed atoms, multi-character elements, bonds, branches, ring indices, stereochemical symbols, and >>. If the

expression does not cover an entire string, tokenization falls back to individual characters; unseen tokens map to <unk>. Input whitespace is removed. Maximum source and target lengths are 1,200 and 768 tokens, including special tokens and EOS. Truncation forces the last retained token to EOS. Batches are dynamically right-padded; source padding is attention-masked and target padding is assigned label -100 and excluded from the loss.

Molecules, cofactors, and stereochemistry. Molecules on the same side are separated by `.`, and reaction sides by >>. An original R>agents>P record contributes its reactant and product sides; the middle agents field is not modeled. The reactant side retains all components present in the prepared view, including recorded cofactors. Forward targets and retrosynthesis inputs use one principal product, whereas span corruption retains the full product side. Principal-product selection first prefers components not recognized as common cofactors and then maximizes a SMILES-based size heuristic. The heuristic counts uppercase non-hydrogen letters rather than computing an exact heavy-atom count. If every component is identified as a cofactor, the highest-scoring component is retained; ties follow input order. Cofactor matching uses an explicit list and canonical or uncharged forms.

Atom-map identifiers are removed from model inputs and targets. Existing chirality and bond-direction symbols are retained; this does not imply complete stereochemical annotation of every record. Forward and retrosynthesis use their respective aligned views and need not be literal string inverses.

Task serialization. Table 8 gives the with-EC formats before EOS is appended. Removing the complete <ec_number>e suffix gives the without-EC format. Here, C replaces masked spans with sentinel tokens while protecting special tokens and the EC suffix.

Table 8: Task-specific source and target formats. EOS is appended to both sequences.

Task	Source	Target
Forward	<forward>R>><mask_product><ec_number>e	$p^\star$
Retro	<retro>$p^\star$<ec_number>e	R
Span	C(<mlm>R>>P)<ec_number>e	Masked spans

C.2 Task Sampling and Span Corruption

Sampling and loss reduction. Training samples records randomly, with distributed sharding across devices. For each presentation, the collator samples one available view uniformly by list position and one task with probabilities $(0.35, 0.35, 0.30)$. The task, view, EC inclusion, and corruption mask can change between presentations. Duplicate stored views are retained, so repeated strings can have greater sampling probability. Equation equation **??** is computed for each device-local batch; distributed training averages the resulting gradients across devices. This is not necessarily a globally token-weighted average when devices have different target-token counts. EOS and sentinel tokens contribute to the loss, and longer targets contribute more supervised positions. No additional task-loss weights, label smoothing, or EC contrastive loss are applied.

Corruption procedure. The span task uses a forward R-SMILES view of the full reaction, with an original-reaction fallback when no valid view is available. After tokenization, the source is initially restricted to 1,199 tokens. Special tokens and all tokens from <ec_number> onward are protected; molecular tokens, `.`, and >> remain maskable. For $M > 0$ maskable positions, the target number of masked tokens is

$$K = \max(1, \lceil 0.15M \rceil)\,. \tag{7}$$

Span lengths are sampled as

$$\ell = \max(1, \mathrm{round}(z))\,, \qquad z \sim \mathrm{Exponential}(\mathrm{rate} = 1/3). \tag{8}$$

Thus, 3 is the scale of the underlying exponential distribution, not a measured mean length of the final merged spans. Sampling extends rightward from a selected start and may revisit previously selected positions. It stops when the token budget is reached, no eligible positions remain, or the number of attempts reaches ten times the sequence length. Adjacent masked positions are merged, ordered by their original positions, and restricted to at most 100 spans.

Each retained span is replaced with <extra_id_i> in the source. The target concatenates each sentinel and its missing span in source order, followed by EOS. No additional terminal sentinel is appended after the final span, and the target does not reconstruct the full reaction. An empty maskable set yields an EOS-only target. The final source and target obey the 1,200/768-token limits.

Reaction-center-dependent start weights require atom-map identifiers in the selected view. The bundled R-SMILES generator removes these identifiers, so views produced through that path use base weights. The configured center/neighbor/base weights of 4/2/1 do not multiply the decoder loss.

C.3 EC Serialization and Label Handling

When available, a label is sampled uniformly from the cleaned, deduplicated, sorted ec4 list; otherwise, selection falls back to the record's ec_number field. The chosen string is appended as <ec_number>e before source EOS with probability 0.5. The tokenizer represents EC digits and periods using its existing tokens, rather than assigning one token to each EC class. Missing annotations always yield unconditioned inputs, so the corpus-wide conditioned fraction need not equal 50%. Because the suffix is appended before truncation, it may be partially or entirely removed from an overlength source.

The implementation does not impose a complete numerical-validity check on all four-part labels; strings containing an unspecified component such as 1.1.1.– are not categorically excluded. Stage-two preparation deduplicates canonical substrate/product pairs without EC in the key and retains the first record, rather than necessarily preserving every EC annotation for that pair. In grouped retrosynthesis inference, the first sorted available EC annotation is selected. Inference removes an optional EC prefix and uses the first four components; labels with fewer than four components do not produce a suffix. The conditions come from benchmark annotations, not from a separately predicted EC label.

C.4 Pretraining Corpus and Split-Specific Filtering

Common preprocessing. The available pipeline starts from merged upstream reaction records and cluster assignments. It selects the designated cluster-center records, applies RXNMapper (Schwaller et al., 2021a), and constructs up to 20 task-specific R-SMILES views with seed 13 and mixed forward mode. Mapping checks examine the presence of atom maps, uniqueness of map identifiers within each side, and agreement between the two sides' map-ID sets. These checks are not a complete elemental or charge-balance test. Records with mapping or R-SMILES failures are retained and can fall back to their original reaction representation. One stored view contains four task-specific strings; these are not four distinct reactions. The common pre-filtering corpus contains 4,168,335 records.

Structural overlap detection. For each benchmark split, comparisons remove atom maps and sort components within each reaction side. Five matching levels are used: canonical isomeric SMILES; canonical SMILES without stereochemistry; uncharged SMILES without stereochemistry; the preceding form after removal of small inorganic components; and that form with repeated components collapsed to a set. The small-component rule removes molecules with at most one heavy atom, or carbon-free molecules with at most five heavy atoms. Matching considers both reaction directions and supports matching a complete reactant side with an individual product. The last two levels additionally require matching EC annotations when the pretraining record has an EC annotation; otherwise, matching is structural only. Uncharging failures retain the original molecule, and there is no explicit tautomer-canonicalization level. These are overlap-detection rules, not the upstream definition of a unique reaction.

Template-associated exclusion. Let T_{train}, T_{val}, and T_{test} be the template-ID sets collected from upstream records that match the corresponding benchmark partitions. Define

$$E = (T_{\text{val}} \cup T_{\text{test}}) \setminus T_{\text{train}}. \tag{9}$$

We remove records that either directly match a validation/test reaction or carry any template ID in E. A scalar template field is used when the template-ID list is unavailable. Templates shared with training matches are not removed merely because they also occur in held-out matches. This is post-generation filtering based on matched-record provenance, not training-only template extraction.

Table 9: Split-specific stage-one corpora. Stored views include duplicates and exclude dynamically constructed fallback views. Records are not audited counts of unique synthetic reactions.

Split	Retained records	20-view records	Zero-view records	Stored views
Reaction	4,044,455	3,660,831	383,624	73,216,620
EC	3,853,604	3,471,243	382,361	69,424,860
Product	4,047,718	3,665,111	382,607	73,302,220

Audit scope and counting. The final audits report zero validation and test reaction overlaps under the specified matching procedure for all three splits. Validation and test structures are used for exclusion and auditing, not for standard model gradient updates. This does not establish that the upstream substrate pool excludes held-out molecules, or that all pretraining products and scaffolds are disjoint from the benchmark held-out sets. The reaction-split filter removes 123,880 records: 9,398 direct reaction matches and 114,482 additional template-associated records. Table 9 distinguishes retained records from stored views. Unique real/synthetic reaction counts and their intersection remain unaudited; no separate real/synthetic sampling ratio is configured.

C.5 Optimization and Checkpoint Selection

Table 10 reports the standard Tiny training settings. Stage two loads the selected stage-one weights but resets optimizer, scheduler, step counter, and early-stopping state. The training seed is 13; internal-validation and fixed benchmark-validation sampling seeds are 10013 and 20013, respectively. Stage-two preprocessing deduplicates canonical substrate/product pairs before an internal training/validation subdivision. These prepared-record counts are not substituted for the benchmark scorer's original test-row denominators.

Table 10: Standard Tiny optimization settings. Step limits are maximum budgets, not actual training durations.

Setting	Stage one	Stage two
Initialization	Random	Stage-one best weights
Optimizer	AdamW	AdamW
Adam betas	$(0.9, 0.999)$	$(0.9, 0.999)$
Weight decay	0	0
Peak learning rate	2×10^{-4}	5×10^{-5}
Warmup steps	1,000	1,000
Schedule	Cosine decay to zero	Cosine decay to zero
Gradient clipping	Norm 1.0	Norm 1.0
Per-device batch size	32	32
Devices / accumulation	8 / 1	8 / 1
Nominal global batch	256 records	256 records
Precision	BF16 autocast	BF16 autocast
Maximum steps	700,000	100,000
Validation interval	16,000 steps	1,000 steps
Early-stopping patience	5 validation checks	5 validation checks

Internal validation. The historical training configuration groups records connected by shared template IDs and assigns groups to training or validation using a seeded stable hash. Records without template IDs use a reaction-key fallback. Hash thresholds are 0.01 in stage one and 0.05 in stage two; these are not guarantees of exact validation-record fractions. Validation uses batch size 32 and a cap of 200 batches per validation loader. Reported loss averages batch-level scalar losses rather than computing one corpus-wide token-weighted mean. Distributed sampler padding can increase the reported sample count above the number of distinct records.

Checkpoint selection. For the standard Tiny runs, checkpoint selection uses internal `val_loss`; the fixed benchmark validation set is monitored separately as `stage2_val_loss` in stage two. For example, the Tiny reaction-split checkpoint selected at step 75,000 records internal loss 0.098622 and

fixed benchmark validation loss 0.116906. These are different metrics and must not be interchanged. Table 11 distinguishes selected and final training steps. The no-span ablation samples forward/retro tasks with probabilities $0.5/0.5$; the stage-two-only ablation starts from random initialization without stage-one training. These are single-seed experiments with different realized training budgets, not strictly compute-matched ablations.

Table 11: Selected checkpoint step / final training step for each run.

Run	Stage one	Stage two
Micro, reaction	544,000 / 630,080	86,000 / 91,136
Compact, reaction	400,000 / 488,312	72,000 / 77,074
Tiny, reaction	288,000 / 378,048	75,000 / 80,100
Medium, reaction	240,000 / 330,792	38,000 / 43,076
Tiny, EC	336,000 / 420,196	72,000 / 77,000
Tiny, product	176,000 / 268,022	67,000 / 72,100
Tiny, no span corruption	176,000 / 267,784	47,000 / 52,154
Tiny, stage two only	Not applicable	93,000 / 98,078

C.6 Inference and Separately Reported Diagnostics

Direct inference. Forward and retrosynthesis evaluation use the selected stage-two checkpoint with and without supplied EC annotations. The configuration requests 20 R-SMILES views, mixed mode, seed 13, beam size 10, and 10 returned candidates per view, retaining at most 10 final candidates. Source/target limits remain 1,200/768 tokens and inference batch size is 2. The candidate count is not a top-k sampling parameter. There is no additional task-specific fine-tuning after stage two; this is not zero-shot evaluation with respect to benchmark training data.

Retrosynthesis fusion. Candidates are canonicalized and deduplicated within each view, so a structure contributes at most once per view. For a candidate a, let $\mathcal{V}(a)$ contain views that generate it, and let $r_j(a)$ be its best rank in view j. Define

$$b(a) = \min_{j \in \mathcal{V}(a)} r_j(a), \qquad q(a) = \sum_{j \in \mathcal{V}(a)} \frac{1}{r_j(a)}. \tag{10}$$

The implementation sorts by decreasing

$$s_{\text{fusion}}(a) = q(a) - 10^8\big(b(a) - 1\big). \tag{11}$$

This prioritizes the best rank obtained in any view, then the reciprocal-rank sum; remaining ties retain first-occurrence order. It combines representations of one model, not independent models or averaged sequence log probabilities.

Molecule-library constraints. The constrained-decoding diagnostic applies token-level constraints from a split-specific molecule library containing train, validation, and test R-SMILES molecules, preserving their SMILES strings. It measures performance with access to a benchmark-wide molecular vocabulary and is kept separate from unrestricted held-out evaluation. Library membership does not establish reaction feasibility or enzyme activity, and this setting is not a guarantee of an accuracy upper bound.

Forward rescoring. Forward rescoring uses the forward task of the same Tiny stage-two checkpoint, with length-normalized teacher-forced scoring of the target product given a candidate reactant mixture. It does not use a separately trained forward model. Two diagnostics are distinguished: the earlier raw-score setting with $\lambda = 0.5$, and a rank-blend setting with

$$s_{\text{blend}}(a) = -\big(r_{\text{fusion}}(a) - 1\big) - \lambda\big(r_{\text{forward}}(a) - 1\big), \tag{12}$$

where larger scores rank first and the fixed main rank-blend setting is $\lambda = 0.25$. The test-set sweep over λ is a sensitivity diagnostic, not validation-based parameter selection. Neither rescoring variant is assumed to improve every condition.

D SYNTHETIC ENZYMATIC REACTION GENERATION

Stage-1 pretraining uses a large synthetic reaction corpus constructed with a template-based generation framework following RSGPT (Deng et al., 2025a). The overall pipeline is

$$\begin{aligned} &\text{enzymatic reactions} \rightarrow \text{reaction templates}, \\ &\text{molecular collection} \rightarrow \text{BRICS fragments}, \\ &\text{templates} + \text{fragments} \rightarrow \text{candidate reactions} \rightarrow \text{filtered synthetic corpus}. \end{aligned} \tag{13}$$

The following sections describe each stage in detail.

D.1 REACTION TEMPLATE CONSTRUCTION

A reaction template represents a transferable local transformation rather than a complete reaction instance. Starting from an atom-mapped reaction, we compare the reactant and product sides to identify atoms and bonds whose local environments change. The transformation together with the required neighboring structural context defines a reaction template.

Conceptually, a template T with K reactant patterns can be written as

$$T : (Q_1, Q_2, \ldots, Q_K) \rightarrow Q_P, \tag{14}$$

where Q_i denotes a reactant-side substructure query and Q_P encodes the corresponding product-side transformation.

The query patterns preserve the local chemical constraints required for template application, including relevant atom and bond properties. Templates are normalized and duplicate transformation rules are merged while retaining their provenance to the source enzymatic reactions.

Each template additionally retains its association with the catalytic annotations of its source reactions when available. Such an inherited EC annotation describes the reaction category from which the template was obtained. It should not be interpreted as experimental evidence that a particular enzyme catalyzes every synthetic reaction generated from that template.

D.2 CANDIDATE SUBSTRATE POOL

To expose each transformation to a larger molecular context, we construct a candidate substrate pool following the fragment-based strategy used in RSGPT (Deng et al., 2025a). Molecules from the upstream molecular collection are decomposed using the BRICS fragmentation algorithm (**?**).

BRICS identifies chemically meaningful bonds that can be disconnected to produce smaller molecular fragments. After fragmentation, attachment points are converted into chemically valid terminated structures, yielding independently parseable fragments. The resulting structures are standardized and deduplicated before template matching.

Let

$$\mathcal{F} = \{f_1, f_2, \ldots, f_N\} \tag{15}$$

denote the resulting fragment pool. These fragments provide candidate substrate contexts; BRICS connection points themselves do not define the enzymatic reaction center.

D.3 TEMPLATE–FRAGMENT MATCHING

For every reactant pattern Q_i of template T, we identify fragments whose molecular structures satisfy the corresponding substructure query. We define the candidate set

$$\mathcal{F}(T, i) = \{f \in \mathcal{F} \mid f \models Q_i\}, \tag{16}$$

where $f \models Q_i$ indicates that fragment f contains a valid match to the template's i-th reactant pattern.

For a template containing K reactant components, the candidate combination space is therefore

$$\mathcal{C}(T) = \mathcal{F}(T, 1) \times \mathcal{F}(T, 2) \times \cdots \times \mathcal{F}(T, K). \tag{17}$$

This combinatorial construction is the main source of data scaling: a single experimentally observed transformation can be instantiated with many compatible substrate combinations. Importantly, matching is based on the structural constraints encoded by the reaction template rather than molecular similarity alone.

D.4 Template-Based Reaction Generation

For each compatible reactant combination

$$\mathbf{f} = (f_1, \ldots, f_K) \in \mathcal{C}(T), \tag{18}$$

we apply template T in the forward direction to obtain one or more candidate products:

$$\mathcal{P}_T(\mathbf{f}) = T(f_1, \ldots, f_K). \tag{19}$$

Each successful template application produces a candidate reaction

$$f_1.f_2.\cdots.f_K \;\longrightarrow\; p, \qquad p \in \mathcal{P}_T(\mathbf{f}). \tag{20}$$

A reactant combination may fail to produce a valid product, and a single combination may yield several products because of multiple matching sites. Such outcomes are handled before the reaction is admitted to the pretraining corpus.

For traceability, generated reactions retain their template provenance and, where available, the catalytic annotation inherited from the source enzymatic transformation.

D.5 Reaction Filtering

Template application enforces a local reaction rule, but it does not guarantee that every generated instance represents a plausible biochemical reaction. We therefore apply several filtering stages before pretraining.

Structural validity. Reactants and products must be parseable molecular structures and the template application must produce a non-empty product. Invalid molecular structures and failed template applications are discarded.

Reaction consistency. We remove degenerate transformations and normalize component ordering and molecular representations before reaction-level comparison. Generated reactions are deduplicated after normalization while retaining template and EC provenance when multiple templates produce the same transformation.

Biochemical and thermodynamic filtering. We further filter the generated candidates using the biochemical constraints and reaction-energy information available in our upstream processing pipeline. This step removes candidates that are inconsistent with the target enzymatic reaction space and prioritizes thermodynamically plausible transformations.

Benchmark isolation. Finally, synthetic pretraining reactions are filtered independently for each VenusRX-Benchsplit. We remove direct overlaps with held-out reactions and additionally exclude records associated with templates that occur exclusively in held-out matches. Let T_{train}, T_{val}, and T_{test} denote template sets associated with reactions matched to each benchmark partition. The excluded held-out template set is

$$T_{\mathrm{held}} = (T_{\mathrm{val}} \cup T_{\mathrm{test}}) \setminus T_{\mathrm{train}}. \tag{21}$$

Synthetic records directly matching validation or test reactions, or associated with a template in T_{held}, are removed from the corresponding pretraining corpus.

D.6 From Synthetic Reactions to Model Inputs

After generation and filtering, the retained synthetic reactions enter the same representation-processing pipeline used by VenusRX.

We first obtain atom mappings with RXNMapper when a valid mapping is not already available (Schwaller et al., 2021a). We then construct up to 20 root-aligned SMILES views for each reaction using R-SMILES (Zhong et al., 2022). Forward and retrosynthetic views are generated according to their respective prediction directions and stored together for downstream multitask training.

This stage should be distinguished from synthetic reaction generation. Template application increases the number of distinct reaction instances,

$$|\mathcal{D}_{\text{reaction}}|, \tag{22}$$

whereas R-SMILES increases the number of sequence realizations of each instance,

$$r \longrightarrow \{s_r^{(1)}, \ldots, s_r^{(M)}\}, \qquad M \leq 20. \tag{23}$$

Therefore, the approximately four million synthetic reactions used for Stage-1 pretraining refer to reaction instances before counting their alternative R-SMILES representations.

D.7 Interpretation of Synthetic Reactions

The generated corpus is intended as large-scale pretraining data rather than a database of experimentally verified enzymatic reactions. Each generated example satisfies an extracted local transformation rule and the implemented filtering criteria, but template compatibility alone does not establish experimental feasibility.

Similarly, an EC label inherited from a template identifies the catalytic category associated with the source transformation. It does not establish that a specific enzyme carrying that EC annotation accepts the newly generated substrates.

The synthetic corpus is therefore used to expose VenusRX to a broader enzymatic reaction space before the model is adapted to experimentally observed VenusRX-Benchreactions during Stage-2 continued training.

E VenusRX-Tiny results without constrained generation

Table 12: VenusRX-Tiny results without constrained generation on the three data splits. We report Top-k exact-match accuracy (%). These results use the standard unconstrained decoding protocol.

Task	EC Input	Reactant Split				Product Split				EC Split			
		T1	T3	T5	T10	T1	T3	T5	T10	T1	T3	T5	T10
Forward	Without EC	73.066	84.611	88.048	90.739	50.959	66.275	71.393	77.085	31.100	45.170	53.236	59.947
	With EC	75.221	86.778	89.725	92.474	55.825	69.463	74.041	78.811	34.564	50.240	56.340	64.058
Retrosynthesis	Without EC	8.446	17.183	23.777	30.603	20.554	30.058	34.923	39.945	4.878	8.737	10.487	14.274
	With EC	17.230	26.398	29.345	34.425	28.847	34.696	38.267	40.568	18.360	24.605	27.481	30.872

F Additional Experimental Results

F.1 Retrosynthesis

Table 13: Round-trip coverage on three data splits. We report RT Coverage@k (%), defined as the percentage of test reactions for which at least one of the top-k retrosynthesis candidates can be mapped back to the target product by the forward oracle. Methods marked with * are zero-shot methods and are evaluated directly using open-source checkpoints. Methods are grouped by whether EC information is provided to the retrosynthesis model. Within each group and column, the best result is bold and the second-best result is underlined.

Method	Reactant Split				Product Split				EC Split			
	C@1	C@3	C@5	C@10	C@1	C@3	C@5	C@10	C@1	C@3	C@5	C@10
Zero-shot												
SciReasoner* (LLM)	**31.163**	**45.853**	**52.260**	**61.125**	**20.206**	**30.489**	**36.266**	**44.487**	**17.953**	**29.782**	**36.937**	**46.045**
RSGPT*	14.247	25.524	29.555	34.098	8.881	14.861	16.982	19.427	8.198	17.797	21.201	24.904
R-SMILES*	3.856	9.879	13.898	19.385	2.721	6.807	10.139	14.849	4.830	8.401	12.548	19.691
Without EC Input												
Graph2SMILES	38.269	52.551	58.504	63.781	28.667	34.552	37.140	42.174	25.443	34.744	38.698	43.313
Retroformer	24.814	34.098	41.344	51.083	10.331	14.669	17.965	22.651	15.292	21.453	25.791	31.771
MEGAN	37.710	53.763	60.252	66.577	30.513	44.451	50.803	57.083	21.213	35.031	41.587	49.269
R-SMILES	29.928	42.661	47.029	52.575	15.832	28.164	35.031	42.581	10.930	22.016	28.092	37.452
BioNavi	39.108	52.365	58.388	64.189	11.397	17.414	20.686	24.724	20.973	29.279	33.030	38.219
G2Retro	31.139	51.549	58.854	67.754	27.205	40.448	49.029	58.293	22.699	39.070	46.069	56.843
RSGPT	22.623	42.428	49.219	60.217	12.021	21.908	26.282	33.869	13.627	24.065	30.717	39.250
VenusRX	**43.383**	**60.648**	**66.787**	**75.128**	**41.107**	**54.602**	**60.954**	**69.607**	**28.883**	**48.394**	**55.081**	**64.753**
VenusRX (MLCD)	43.057	58.586	63.269	70.282	38.255	49.940	56.520	62.105	28.655	47.711	53.152	60.990
With EC Input												
ECReact	44.851	60.135	65.284	70.701	28.152	36.601	41.407	47.291	32.862	43.349	48.370	55.633
VenusRX	**53.308**	**66.880**	**72.064**	**76.887**	**47.915**	**56.544**	**63.015**	**69.307**	**42.006**	**53.428**	**58.221**	**65.964**
VenusRX (MLCD)	50.233	63.723	67.719	74.348	43.996	53.248	57.251	62.668	38.231	50.096	53.955	60.822

Table 14: Round-trip accuracy on three data splits. We report RT Accuracy@k (%), defined as the fraction of the first k retrosynthesis candidates that can be mapped back to the target product by the forward oracle. Methods marked with * are zero-shot methods and are evaluated directly using open-source checkpoints. Methods are grouped by whether EC information is provided to the retrosynthesis model. Within each group and column, the best result is bold and the second-best result is underlined.

Method	Reactant Split				Product Split				EC Split			
	A@1	A@3	A@5	A@10	A@1	A@3	A@5	A@10	A@1	A@3	A@5	A@10
Zero-shot												
SciReasoner* (LLM)	**31.163**	**24.282**	**20.503**	**17.672**	**20.206**	**14.533**	**12.081**	**9.340**	**17.953**	**13.774**	**12.157**	**9.939**
RSGPT*	14.247	12.950	10.848	7.780	8.881	6.935	5.297	3.496	8.198	7.674	6.764	4.715
R-SMILES*	3.856	4.042	4.268	3.929	2.721	2.621	2.812	2.641	4.830	3.356	3.490	3.476
Without EC Input												
Graph2SMILES	38.269	33.652	30.594	25.495	28.667	22.064	18.502	14.403	25.443	22.108	19.578	15.511
Retroformer	24.814	22.930	22.260	21.553	10.331	9.416	8.758	7.296	15.292	14.446	13.320	11.956
MEGAN	37.710	35.089	32.968	29.065	30.513	25.559	23.418	20.150	21.213	19.451	18.068	15.690
R-SMILES	29.928	26.130	24.117	21.262	15.832	12.568	11.644	9.950	10.930	10.067	9.267	8.201
BioNavi	39.108	35.702	33.979	30.325	11.397	9.368	8.205	6.510	20.973	18.333	16.285	13.424
G2Retro	31.139	27.512	24.497	21.454	27.205	20.877	19.456	16.419	22.699	20.254	17.543	14.880
RSGPT	22.623	20.857	20.315	18.113	12.021	9.907	8.411	6.874	13.627	12.672	11.663	9.884
VenusRX	**43.383**	**37.508**	**34.434**	**30.494**	**41.107**	**31.516**	**28.346**	**25.345**	**28.883**	**25.803**	**23.027**	**20.152**
VenusRX (MLCD)	43.057	35.038	31.379	27.340	38.255	27.744	24.295	18.948	28.655	24.549	21.026	16.710
With EC Input												
ECReact	44.851	42.280	**39.814**	**35.216**	28.152	27.201	26.515	24.895	32.862	**32.882**	**31.853**	**30.128**
VenusRX	**53.308**	**43.480**	38.844	34.287	**47.915**	**36.326**	**32.728**	**27.096**	**42.006**	31.819	28.032	23.688
VenusRX (MLCD)	50.233	39.418	33.684	28.879	43.996	30.661	25.686	20.147	38.231	26.127	22.090	17.338

F.2 EC PREDICTION

Table 15: EC-number prediction on the Reactant and Product splits. Macro F1 (%) is reported at EC levels 1–4.

Method	Reactant Split				Product Split			
	EC1	EC2	EC3	EC4	EC1	EC2	EC3	EC4
Zero-Shot								
crxnfp (Zhang et al., 2025)	86.64	71.17	57.28	27.35	68.45	55.85	46.12	21.12
DRFP (Probst et al., 2022b)	85.75	72.80	57.53	31.94	71.11	60.03	51.23	26.15
rxnfp (Schwaller et al., 2021b)	83.71	70.56	56.36	28.46	66.90	54.32	45.66	21.11
VenusRX*	**87.38**	**73.05**	**61.05**	**33.31**	**73.46**	**61.14**	**51.96**	**26.90**
Task Specific Model								
Theia (Probst, 2023)	83.72 ± 1.70	71.34 ± 0.95	$\mathbf{62.15} \pm 1.75$	$\mathbf{36.29} \pm 0.53$	73.72 ± 2.03	$\mathbf{63.79} \pm 1.23$	52.92 ± 1.07	$\mathbf{27.81} \pm 0.02$
CLAIRE (Zeng et al., 2025)	$\mathbf{89.80} \pm 0.66$	$\mathbf{72.32} \pm 2.93$	59.51 ± 1.57	31.74 ± 0.74	74.50 ± 0.22	57.82 ± 0.68	49.99 ± 0.82	23.96 ± 0.56
BEC-Pred (Qian et al., 2024)	85.35 ± 6.25	61.58 ± 1.10	42.64 ± 0.59	28.13 ± 0.43	$\mathbf{76.70} \pm 1.49$	60.23 ± 1.03	43.30 ± 0.89	21.18 ± 0.27
VenusRX	88.35 ± 1.54	71.65 ± 0.64	62.10 ± 0.63	33.93 ± 0.93	76.01 ± 0.69	62.12 ± 1.88	$\mathbf{54.95} \pm 0.57$	26.09 ± 0.21

VenusRX denotes frozen VenusRX representations followed by a task-specific predictor. Task-specific results are mean $\pm$ std over three seeds.

G Details of Molecule Library-Constrained Decoding

G.1 Implementation

MLCD constrains autoregressive decoding with a prefix tree constructed from the tokenized molecular representations in the predefined molecule library. Each trie node corresponds to a molecular token prefix, and its outgoing edges specify the valid next tokens. A molecule can terminate only when the current prefix reaches a complete library entry.

For multi-molecule outputs, the trie is reset after each molecule separator so that the constraint is applied independently to every generated molecule. Reaction-level control tokens and separators are generated normally. MLCD therefore changes only the set of admissible molecular tokens; sequence probabilities and beam ranking remain determined by VenusRX.

Algorithm 1 summarizes the decoding procedure.

G.2 Analysis of Molecule Library-Constrained Decoding

We provide additional analysis of Molecule Library-Constrained Decoding (MLCD) for retrosynthesis.

Generation validity. We report the raw invalid rate, defined as the percentage of generated molecular candidates that cannot be parsed as valid SMILES before final filtering. Missing Top-k slots are not counted as invalid molecules.

As shown in Table 16, MLCD eliminates invalid raw molecular candidates in all evaluated retrosynthesis settings. This supports its role in constraining generation to structurally valid molecular outputs.

Round-trip evaluation. For each retrosynthesis candidate, we apply a fixed forward oracle and test whether the predicted product matches the target product. We report RT Accuracy@k, the fraction of the first k candidates that map back to the target, and RT Coverage@k, the fraction of test reactions for which at least one of the first k candidates maps back to the target.

Tables 14 and 13 show that unconstrained decoding generally achieves higher round-trip scores than MLCD. Combined with the higher exact-match accuracy of MLCD, this indicates that constrained decoding improves recovery of recorded benchmark targets while reducing the diversity of alternative candidates retained by unconstrained decoding.

Algorithm 1 Molecule Library-Constrained Decoding (MLCD)

Require: Input x, model p_θ, molecule trie $\mathcal{T}$, beam size B
Ensure: Ranked output sequences

```
Initialize beam set B with the start token
while not all beams are finished do
    B_new ← ∅
    for each partial sequence y ∈ B do
        if y is currently generating a molecule then
            h ← current molecular prefix
            A ← valid next tokens from T given h
        else
            A ← valid reaction-level tokens
        end if
        Expand y using tokens in A
        Score expansions with p_θ(· | x, y)
    end for
    Keep the top-B scored expansions
    Update B
end while
return completed sequences ranked by model score
```

Table 16: Raw invalid SMILES rate (%) for retrosynthesis with and without MLCD. Lower is better.

Split	EC	Unconstrained	MLCD
Reactant	✓	2.277	**0.000**
	–	1.599	**0.000**
Product	✓	3.641	**0.000**
	–	2.756	**0.000**
EC	✓	3.086	**0.000**
	–	2.210	**0.000**