nature portfolio

https://doi.org/10.1038/s41586-025-09426-9

Supplementary information

Electron flow matching for generative reaction mechanism prediction

In the format provided by the authors and unedited

Supplementary information for Electron flow matching for generative reaction mechanism prediction

Joonyoung F. Joung[1,2†], Mun Hong Fong[1†], Nicholas Casetti[1], Jordan P. Liles[1], Ne S. Dassanayake[3], Connor W. Coley[1,4*]

[1]Department of Chemical Engineering, Massachusetts Institute of Technology, 77 Massachusetts Ave., Cambridge, 02139, Massachusetts, United States.
[2]Department of Chemistry, Kookmin University, 77 Jeongneung-ro, Seongbuk-gu, Seoul, 02707, Republic of Korea.
[3]Department of Chemistry, Massachusetts Institute of Technology, 77 Massachusetts Ave., Cambridge, 02139, Massachusetts, United States.
[4]Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 77 Massachusetts Ave., Cambridge, 02139, Massachusetts, United States.

*Corresponding author(s). E-mail(s): ccoley@mit.edu;
Contributing authors: jjoung@mit.edu; fong410@mit.edu; ncasetti@mit.edu; jliles@mit.edu; nedassa@mit.edu;
†These authors contributed equally to this work.

Contents

S1 Code and Data availability

All data used in this study, along with pretrained single-step models, are publicly available on Figshare: https://doi.org/10.6084/m9.figshare.28359407.v2. The dataset is provided in a text format, including reaction SMILES.

The source code for dataset curation can be found in the following GitHub repository: https://github.com/jfjoung/Mechanistic_dataset. This repository includes scripts for preparing mechanistic dataset.

The implementation of FlowER, including model training, inference, and evaluation, is available at: https://github.com/FongMunHong/FlowER. This repository provides training scripts and inference pipelines necessary to reproduce the results presented in this manuscript.

Both repositories include a detailed README with instructions for installation, usage, and result reproduction.

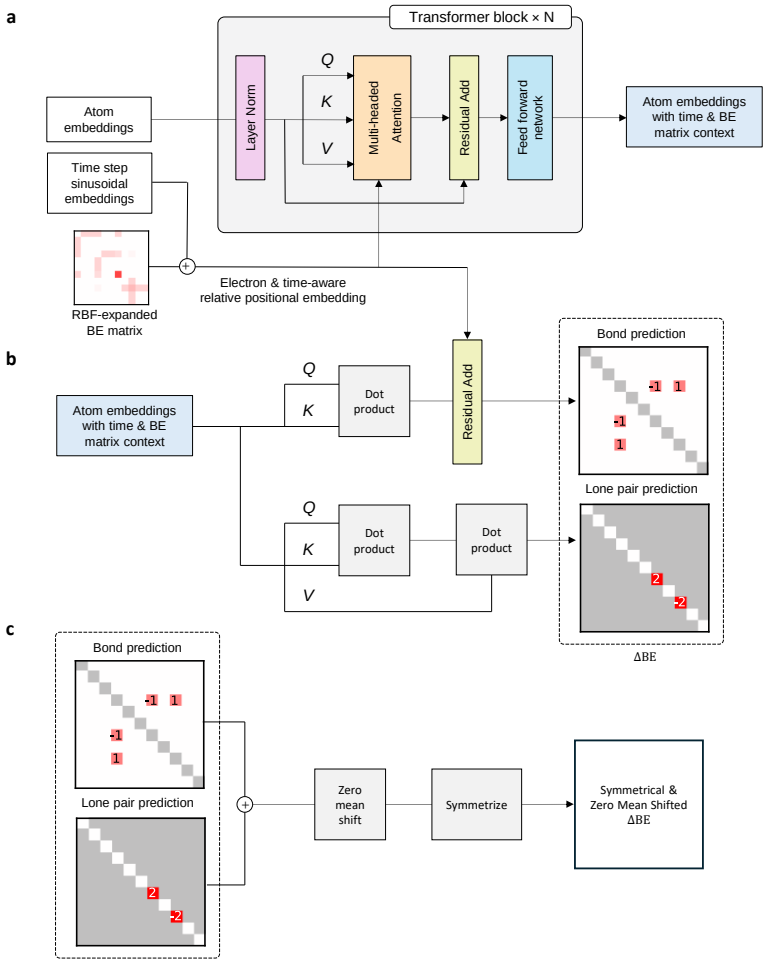

Fig. S1 Detailed architecture of the core processing block in FlowER. **a** The model receives one hot encoded atom embeddings as input. The RBF-expanded bond-electron (BE) matrix and sinusoidal timestep embedding are concatenated to provide an electron- & time-aware relative positional embedding (RPE), fed into a multi-head attention module along with the input representation. The output represents atom embeddings contextualized by both time and BE matrix information. **b** These contextualized embeddings are processed by separate linear layers to compute dot products, yielding the bond and lone pair prediction matrices. The bond prediction receives the graph & time-aware RPE as additional residual connection. Together, these form the predicted change in the bond-electron matrix (ΔBE). **c** The ΔBE matrix is zero-mean shifted and symmetrized to ensure electron conservation, representing the final output of the FlowER processing block.

S2 Additional quantitative results

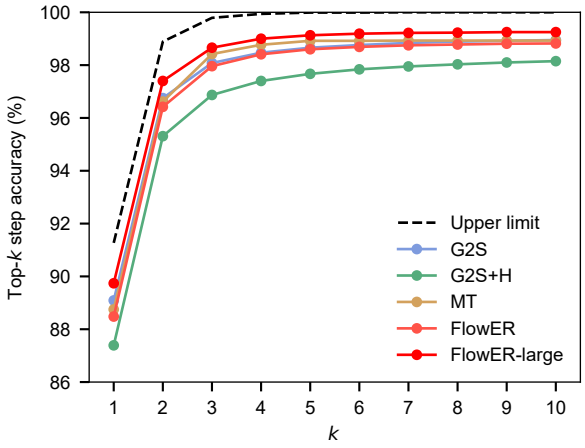

Fig. S2 Top-k step accuracy for predicting the product of a single elementary step. Dashed line represents the upper limit.

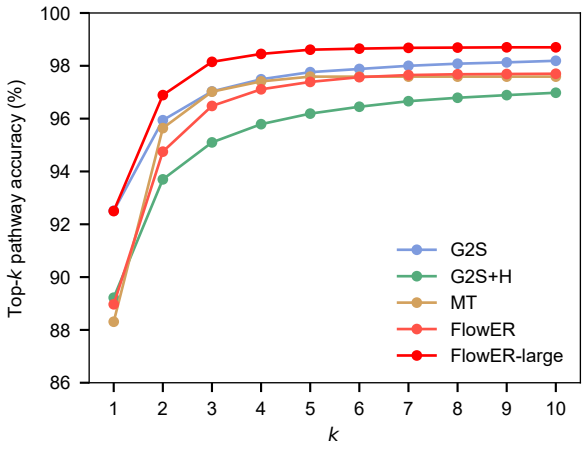

Fig. S3 Top-k pathway accuracy for predicting the full reaction pathway during beam search, showing the minimum beam width (k) required to capture the correct sequence of elementary steps.

S3 Mechanistic dataset preparation

We utilized the USPTO-Full dataset [1], which contains 1,100,105 reactions, for curating an expanded mechanistic dataset. Building on our previous method for constructing mechanistic datasets [2], we significantly extended the scope of the dataset. This involved creating 1,220 unique reaction templates, covering 252 reaction classes and 185 distinct mechanisms. By adding more reaction classes and mechanisms, we ensured broader coverage and diversity. Since our method of data imputation relies on reaction class labels, we classified reactions using NextMove's NameRxn software [3].

S3.1 Expert-curated reaction templates

Expert-curated reaction templates were manually constructed by two researchers involved in the project and reviewed by a PhD-level organic chemist to ensure mechanistic plausibility. Each template encodes a single elementary transformation and is written in SMARTS format for consistency and interoperability.

The design of templates was guided by well-accepted reaction mechanisms described in standard organic chemistry textbooks. We included mechanistic motifs commonly observed in both classical transformations and widely recognized named reactions (e.g., aldol condensation, Beckmann rearrangement, Claisen rearrangement, olefin metathesis), extending the scope beyond the most frequent reaction types identified in the USPTO-Full dataset.

Templates were constructed to obey conservation of heavy atoms, hydrogen atoms, and electrons. For proton transfer steps, compatible acid/base species were dynamically selected from the reaction pool based on these annotated pK_a thresholds, avoiding the need to hard-code specific species while maintaining chemical plausibility.

To promote the accuracy of the constructed templates, three chemists reviewed each mechanism. A series of templates corresponding to a given reaction class was then sequentially applied to a set of reactants, generating possible intermediates and ultimately the products.

To ensure consistency with the experimentally observed products of the overall reactions, we algorithmically pruned unproductive intermediates and retained only the mechanistic steps leading to the reported products. Through this process, we were able to impute intermediates not explicitly recorded in the overall reactions and identify often-overlooked byproducts.

A full set of the expert-encoded templates used in this work is available as part of our dataset release. We also provide an interactive visualization interface through a public Google Colab notebook[1]. The same link is included in the `README` of the GitHub repository for convenient access. We expect the collection of templates to both grow and be refined over time, and welcome community contributions and pull requests to the repository.

[1] https://colab.research.google.com/github/jfjoung/Mechanistic_dataset/blob/main/Mechanistic_dataset_visualization.ipynb

S3.2 Stoichiometry and acid-base reactions

A significant number of elementary steps involve acid-base reactions. In our previous work [2], hydrogen conservation was not fully addressed due to the challenge of explicitly specifying all possible acid-base partners for each template. To overcome this limitation, we redesigned the templates to ensure hydrogen conservation and accurately model acid-base reactions. Each template was constructed as a unimolecular reaction, where the intermediate either gains or loses a proton. To account for the proton donor or acceptor, we annotated the p$K_a$of the required partner directly in the template.

We compiled p$K_a$values for 88 distinct acid-base pairs. During the mechanistic dataset generation process, whenever a template requiring a proton transfer was applied, the following procedure was implemented:

- If the reaction required an acid, the system checked whether an acid with a p$K_a$lower than the specified value was present in the current set of chemicals. If such an acid was found, it was added to the reactant side of the reaction, while the corresponding base was added to the product side, transforming the unimolecular template into a bimolecular one.
- Conversely, if a base was required, the system checked for a base with a p$K_a$higher than the specified value and adjusted the reaction in the same manner.

While our approach ensures hydrogen conservation and general acid-base reactivity via pK_a-threshold annotations, the implementation is inherently local to the reaction center. As a result, compatibility between acidic and basic species is not enforced globally across intermediates, which can occasionally result in chemically implausible scenarios, such as the coexistence of strong acids (e.g., HCl) with neutral (e.g., water) or basic species (e.g., alkoxides or amines). These species are added dynamically from the reactant pool based on their pK_a values, but no explicit neutralization is imposed unless dictated by a separate elementary step. In complex mechanisms with multiple proton transfers, this may yield transient zwitterions or unphysical charge-separated species.

In practice, resolving this limitation is complicated by the nature of datasets like USPTO, where acids and bases used during workup are recorded in the same reactant pool as those used for the core reaction. As a result, it is often unclear at which point in the mechanism a given proton donor or acceptor was introduced. This ambiguity makes it difficult to enforce global proton-neutralization constraints without inadvertently removing species essential for the intended transformation. For example, prematurely neutralizing a nucleophile with a workup-intended acid may prevent the actual reaction from occurring. To address this, future iterations of our dataset generation pipeline may incorporate temporal labeling of reactive species or context-aware constraints to distinguish between reaction-phase and workup-phase roles of acid-base species. Such refinement would help eliminate chemically implausible intermediates and improve the fidelity of the generated mechanisms.

Additionally, while applying reaction templates, we addressed scenarios where multiple equivalents of a reactant were required, a common occurrence in acid-base and catalytic reactions. For example, the USPTO-Full dataset often records only one equivalent of a reactant, even when two or more equivalents are necessary for the reaction

to proceed. If a template required the same reactant that had already been consumed in a previous elementary step, the mechanistic data generation process was adjusted as follows:

1. All earlier elementary steps were revised to duplicate the required reactant, ensuring sufficient quantities were present throughout the reaction sequence.
2. The current template was then applied to the reaction, utilizing the additional equivalent of the reactant.

This approach ensured that the mechanistic pathways generated from the templates remained chemically accurate, consistent with stoichiometric requirements, and reflecting the realistic reaction conditions.

S3.3 Kekulé representation of molecules

In constructing the BE matrix, we adopted the Kekulé representation for aromatic rings instead of using the classical BE matrix form proposed by Ugi in the 1970s [4, 5]. In Ugi's BE matrix, aromatic bonds are represented with a bond order of 1.5. However, in fused ring systems, this approach leads to an issue where certain carbon atoms, shared between two aromatic rings, end up with three bonds of order 1.5. As shown in the BE matrix of the canonical form in Fig. S4, this results in a total electron count of 4.5 for such atoms, requiring lone pairs to be assigned a non-physical value of -0.5 electrons to maintain overall electron conservation. While the total electron count remains consistent, the introduction of negative fractional electrons is unrealistic and undesirable.

To address this, we adopted the Kekulé form, explicitly representing aromatic bonds as alternating single and double bonds. As seen in the BE matrix of the Kekulé form in Fig. S4, this approach ensures that all bond orders are represented as non-negative integers, making the BE matrix more physically meaningful. Kekulization was applied systematically using RDKit's `Chem.Kekulize(mol)` function.

S3.4 Data cleaning and validation

To ensure the reliability and consistency of the mechanistic dataset, we applied rigorous cleaning procedures.

- **Atom Mapping Consistency**: All atom mappings were verified to be unique and consistent between reactants and products.
- **BE Matrix Validation**: The BE matrix was checked to be diagonally symmetric, and all entries were ensured to be either 0 or positive integers. Conservation of electrons was verified by confirming that the sum of the BE matrix values remained constant before and after each reaction.
- **Chemical Validity Check**: Structures were screened using RDKit to remove chemically invalid molecules, such as those with incorrect valency or problematic resonance structures.
- **Kekulization Filtering**: If Kekulization failed for a given molecule, all steps involving that molecule were removed from the dataset.

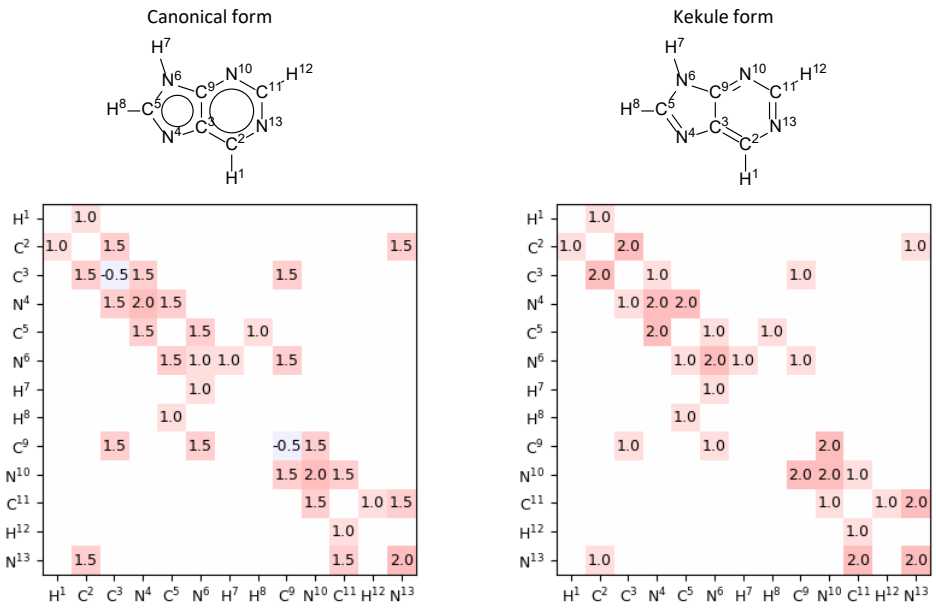

Fig. S4 Comparison of BE matrices for the canonical and Kekulé forms. The canonical form represents aromatic bonds with bond order 1.5, leading to non-physical fractional electron counts. In contrast, the Kekulé form explicitly represents alternating single and double bonds, ensuring integer number of electrons.

- **Pathway Integrity Check**: If the removal of invalid steps resulted in the loss of a complete reaction pathway from reactants to the observed product, the entire sequence was discarded to maintain dataset consistency.

These validation steps ensured that the final dataset contained only chemically meaningful and computationally robust reaction sequences, improving its reliability for reaction modeling.

S3.5 Integration with external databases

To enhance dataset diversity, we incorporated data from RMechDB [6] and PMechDB [7]. Since these external datasets provided atom mappings only for reaction centers, we used RXNMapper [8] to extend atom mappings to the entire reaction while ensuring consistency with our existing mapping approach.

Before integration, we applied the same cleaning protocols used for our mechanistic dataset, including atom mapping validation, BE matrix verification, and stoichiometry checks.

S3.6 Final dataset

The resulting dataset consisted of:

- Training Set: 250,782 overall reactions (1,445,189 elementary steps)

- Validation Set: 2,801 overall reactions (15,744 elementary steps)
- Test Set: 28,049 overall reactions (162,002 elementary steps)

The dataset spans 240 reaction classes and 185 distinct mechanisms, covering a broad range of organic transformations. The dataset was split into training, validation, and test sets using an 89:1:10 ratio.

S3.7 Future direction

Another important future direction is to refine how proton transfer steps are handled. The current mechanistic templates encode proton transfers locally based on pK_a thresholds, without enforcing global acid-base consistency across intermediates. While this simplifies template construction, it may occasionally yield chemically implausible intermediates. These inconsistencies stem from the dataset construction process, particularly the lack of distinction between reaction-phase and workup-phase components in sources like USPTO. In future work, we aim to improve mechanistic fidelity by introducing temporal labeling of reactive species and context-aware constraints to better distinguish the roles of acids and bases across multi-step pathways. Future extensions may incorporate synthetic examples of failed or competing mechanisms, such as premature acid-base neutralizations or unintended side reactions involving organometallic reagents, to further expand the model's predictive scope.

S4 Runtime and model scalability

To assess the inference performance and scalability of our model, we conducted a comparative runtime analysis between FlowER, Graph2SMILES with implicit hydrogens (G2S), and Graph2SMILES with explicit hydrogens (G2S+H). Each model was evaluated on a single NVIDIA RTX A5000 GPU. The benchmark involved providing a single set of reactants of varying total heavy atom counts and generating 16 output samples per input.

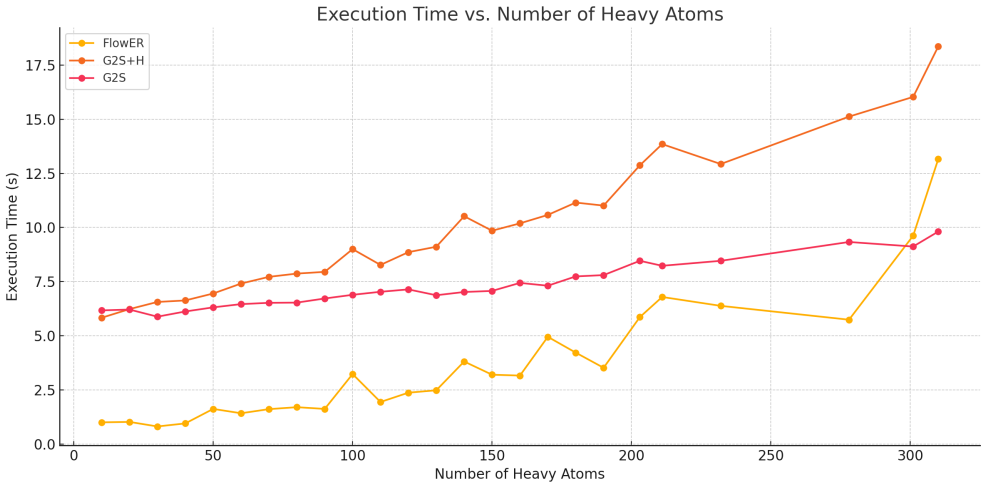

Fig. S5 Execution time for 16 inference passes as a function of molecule size (number of heavy atoms) for FlowER, G2S, and G2S+H. FlowER offers faster inference time over G2S for all but the very largest systems. This experiment is conducted using randomly sampled reactant sets varying in total heavy atom count from 10 to 310.

As shown in Figure S5, FlowER demonstrates improved inference time compared to G2S+H for single reactants set inputs where hydrogens are modeled explicitly. In contrast, G2S achieves comparable inference time when omitting explicit hydrogen representation when reactants' heavy atom count approaches or exceeds 300.

While FlowER performs faster in single-system inference, the situation reverses in the context of batched inference. When evaluating inference time over the full test (top-k step prediction and pathway prediction), G2S+H completes in approximately 12 hours, whereas FlowER takes around 28 hours. This discrepancy arises from less efficient memory-efficient batching of FlowER's dense BE matrix representation compared to integer-encoded tokens, leading to smaller batch sizes and memory bottlenecks.

Despite the faster speed for a single inference pass, the need to explicitly represent all atoms including hydrogens in FlowER may pose a significant memory challenge for large molecular systems, such as full proteins. In such cases, potential strategies to mitigate resource requirements include: coarse-graining the input representation to

heavy atoms or backbone-only fragments or employing hybrid modeling strategies, such as QM/MM-like embeddings. Such strategies could effectively reduce the dimensionality of the BE matrix and facilitate more tractable inference without sacrificing model applicability. Further efficiency gains may be possible by leveraging low-rank or sub-quadratic approximations in transformer attention mechanisms. Exploring these directions forms an exciting avenue for future work in extending FlowER to larger biomolecular systems.

S5 Post processing of model output

S5.1 Sum-safe rounding

The sum-safe rounding is performed through a reimplementation of the python package `iteround` [9] for better parallelization using gpu-based implementation. It provides a safe-sum rounding, which ensures the preservation of electron counts after rounding from floating point number back to representable integer bond order on the BE matrix, allowing us to reconstruct molecules back using RDKit.

$$\text{diff} = x - round(x)$$

This equation represents how each value is incremented / decreased sequentially according to the sorted difference. The algorithm alternates between the smallest surplus / deficit to alter each floating point back to it's rounded integer.

S5.2 Validity fix

The validity fix stems from the notion that the product atoms' valence electron count should be the same as the reactant atoms' valence electron count. E.g., if the input reactant valence electron count is [2, 8, 8, 8, 8, 2], predicted product valence electron count is [2, 6, 8, 8, 10, 2], difference would be [0, -2, 0, 0, +2, 0]. If the sum of difference is zero, we apply this validity fix through "redistributing" the electrons back to match it's reactant valence electron count. Empirically, this fix improves BE matrix to SMILES conversion validity by 3-4% on the test set, and typically reactants with HCNOF elements that strictly obey the octet rule benefits from it.

S5.3 Failure mode analysis

Despite the post-processing techniques described above, FlowER still produces 5.06% invalid SMILES during sampling, as mentioned in the main text. To better understand these failure cases, we analyzed the invalid outputs and categorized them into three major types. Fig. S6 illustrates the distribution of the three failure cases identified in FlowER's sampled predictions.

- **Electron over-redistribution in prediction**
  In some cases, FlowER predicts an excessive electron redistribution, moving more electrons from the bonds/lone pairs than what is originally available in the reactants. This results in negative electron counts in the product BE matrix, making it physically invalid. This issue originates from errors in the ΔBE matrix prediction, where the model incorrectly estimates the number of electrons involved in rearrangement.
- **Diagonal symmetry violation after sum-safe rounding**
  While sum-safe rounding ensures that the total number of electrons is preserved, it does not guarantee that the product BE matrix remains diagonally symmetric. If the rounding process distorts symmetry, the number of bonding electrons can become odd-valued, which leads to errors when reconstructing molecules.

In BE matrix-based molecule reconstruction, single, double, and triple bonds are defined as having 2, 4, and 6 bonding electrons, respectively. If a bond has an odd number of electrons due to symmetry breaking, it cannot be correctly mapped back to a valid molecular structure, resulting in an invalid SMILES.

- **Chemically invalid molecule despite valid BE matrix**

  In some cases, the sampled product BE matrix maintains mass conservation and diagonal symmetry but still fails to generate a chemically valid molecule. These cases occur when the predicted electron distribution leads to an unstable molecular structure, violating chemical stability rules. Examples of such chemically invalid molecules are illustrated in Fig. S7.

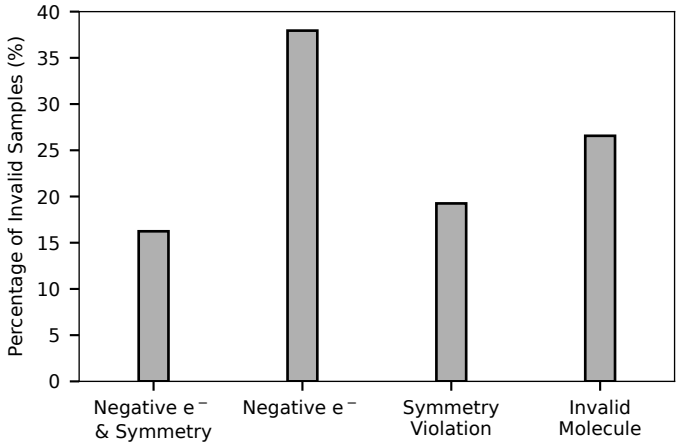

Fig. S6 Histogram of failure modes in FlowER's sampled predictions. The four categories represent different causes of invalid SMILES: (1) negative electron counts and symmetry violation, (2) negative electron counts only, (3) symmetry violation only, and (4) chemically invalid molecules.

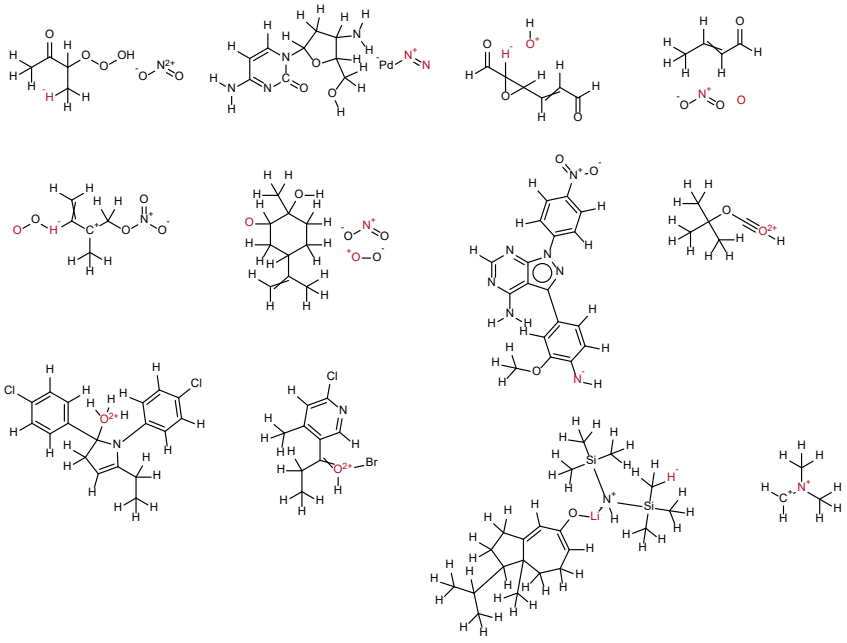

Fig. S7 Examples of chemically invalid molecules generated by FlowER. Problematic regions within the molecules are highlighted in red.

S6 Ambiguity around representations of ions

Ions such as LiOH can be represented in two major ways: as an ionic form, where the components are separated into charged species (e.g., `[Li+].[OH-]` in SMILES), or as a neutral form, where the molecule is written without explicit charges (e.g., `[Li]O` in SMILES). These two representations correspond to the same chemical compound in essence, but they result in different BE matrices. Variations in the BE matrix may impact the model's predictions. While salts included in the workup do not participate in the reaction, acids are typically recorded in their neutral form, whereas bases can appear in either ionic or neutral representations.

To better understand this variation, we analyzed how bases are recorded in both the USPTO dataset (all reactions) and the reactions, in Pistachio, from patents reported in 2024 [10] that were not assigned to any of the 2,639 predefined reaction types in NameRxn [3]. The distribution between ionic and neutral forms is summarized in Fig. S8. In the USPTO dataset, strong organolithium bases that contain C–Li bonds are recorded in their neutral form, whereas nearly all other bases are represented in their ionic form. In contrast, bases in the Pistachio dataset are recorded in both ionic and neutral forms, with no consistent convention.

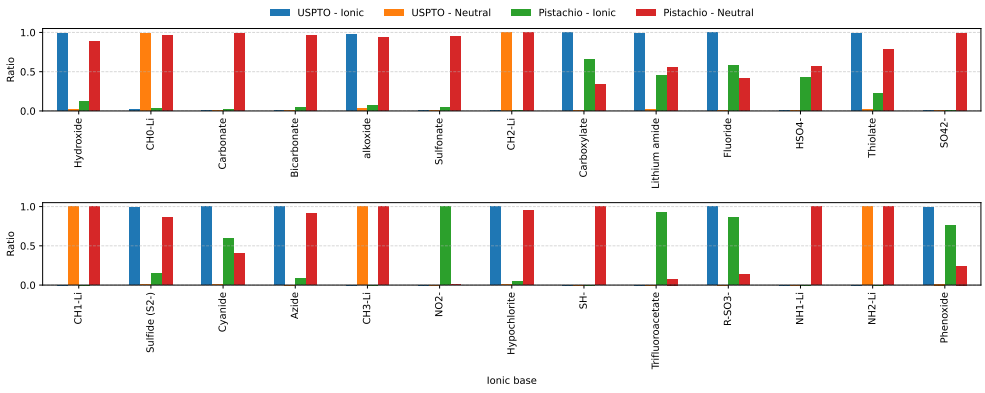

Fig. S8 Comparison of ionic and neutral forms of 26 representative bases across two datasets. All reaction in the USPTO dataset (blue/orange) were used, while the Pistachio dataset (green/red) includes only reactions reported in 2024 that were not categorized to any predefined reaction types. For each base, the proportion of occurrences in ionic versus neutral form is shown as a ratio within each dataset. This histogram shows which form dominates in each datasets and how bases are represented in each datasets.

To explore how these representational differences affect model behavior in practice, we conducted a focused analysis of 29 bases that appear in both neutral and ionic forms. An expanded analysis of 29 bases with both neutral and ionic representations is summarized in Extended Table 1. Some bases exhibited differing reactivity between their neutral and ionic forms, while others showed consistent behavior across forms. For example, in the case of sodium ethoxide, the neutral form exclusively predicted keto alpha-alkylation, whereas the ionic form predicted both keto alpha-alkylation and

S_N2 pathways. For sodium ethanethiolate (EtSNa, p$K_a \approx 10$), neither the neutral nor ionic form successfully predicted α-alkylation; instead both predicted the S_N2 reaction. These nuances in behavior result from dataset trends (e.g., underrepresentation of thiols used as bases rather than nucleophiles) and representation conventions (e.g., of neutral sodium salts versus ionized fragments).

While representation issues may be addressed through standardized preprocessing to unify conventions across datasets, both of these issues may also be corrected through data augmentation. As it stands, FlowER is not robust to these bookkeeping differences and so queries made using different representations of ionic species may lead to substantially different predictions.

S7 Hyperparameter tuning

Table S1 Hyperparameter setting used in the experiments for different datasets. Best settings selected based on validation are highlighted in **bold** if multiple values have been experimented.

Model	Dataset	Parameter	Value(s)
FlowER	All	Filter size in Transformer	2048
		Attention encoder layers	6, 8, **12**
		Attention encoder heads	8, **32**
		Sigma	0.1, 0.13, **0.15**, 0.16, 0.17, 2.0
		Learning rate	**0.0001**, 0.001
		Sample size	16, 32, **64**
		Scheduler	**NoamLR**, StepLR, None
FlowER	Full	Embedding size	**128**, 256
		Hidden size (same among all modules)	**128**, 256
		RBF low, high, step	**{0, 8, 0.1}**, {0, 18, 0.1}
FlowER-large	Full	Embedding size	128, **256**
		Hidden size (same among all modules)	128, **256**
		RBF low, high, step	{0, 8, 0.1} ,**{0, 18, 0.1}**
		ODE solver method	**Dopri5**, Euler
		ODE solver absolute error tolerance	$\mathbf{10^{-4}}$
		ODE solver relative error tolerance	$\mathbf{10^{-4}}$
FlowER	Full	Total number of steps	3000000
FlowER	50k, 5k, 1500, 500	Total number of steps	500000
FlowER	Unrecognized Pistachio	Beam size	2
		N best	3
		Max depth	9
		Chunk size	50
Graph2SMILES	All	Embedding size	**256**, 512
		Hidden size (same among all modules)	**256**, 512
		Filter size in Transformer	2048
		Number of D-MPNN layers	2, **4**, 6
		Attention encoder layers	4, **6**
		Attention encoder heads	8
		Decoder layers	4, **6**
		Decoder heads	8
		Number of accumulation steps	4
		Batch size	4096
Graph2SMILES	Full	Total Steps	1200000
		Noam learning rate factor	2
		Dropout	0.1
Graph2SMILES	50k, 5k, 1500, 500	Total Steps	300000
		Noam learning rate factor	2, **4**
		Dropout	0.1, **0.3**

S7.1 Beam search hyperparameters

To predict reaction pathways, FlowER operates in an expand and search manner, where expansion refers to providing plausible outcomes (intermediates or products) through ODE integration, and search refers to selection of which outcome to pick next for expansion.

Below are the list of hyperparameters which would guide users to tune FlowER's search capabilities.

Beam Size - Refers to the size of top-k selection of candidates (based on cumulative probability) to be further expanded. Increasing this would make the overall search more comprehensive, but at the cost of slower runtime.

N-Best - Refers to the cut-off size of the top-k outcomes generated by FlowER after the expansion. This cutoff can filter out unlikely outcomes to be part of the selection.

Sample Size - Refers to the amount of outcomes FlowER would sample. It refers to the raw (& duplicated) sample size count.

Max Depth - Refers to the maximum depth the beam search should explore.

Chunk Size - Refers to the number of reactants to be run beam search concurrently.

S8 Low data regime experiments

A data-constrained evaluation is carried out to demonstrate the effectiveness of FlowER as compared to sequence translation based approach in predicting reaction outcome with only very small amount of data. Training data is randomly subsampled into smaller partitions of 50k, 5k, 1500, and 500. Using a training set size of 1500 roughly corresponds to 1 example per reaction class & condition pair from the full training set. Despite this very small amount of data, even when training from scratch, FlowER's problem formulation and architecture provides impressive evidence of generalization on the reaction outcome prediction task.

S9 Recovering reaction pathways with FlowER

We analyzed reactions from patents reported in 2024 [10] that were not assigned to any of the 2,639 predefined reaction types in NameRxn [3]. These cases include reactions that may involve unexpected product formations, combinations of known transformations, or mechanistic pathways that are not explicitly categorized within existing classification schemes.

We performed a narrow beam search (width 2, depth 9) on 22,000 unrecognized reactions and successfully recovered 351 products.

The five reaction pathways successfully recovered by FlowER, originating from reactions reported in nine patents from 2024 [11–21], are illustrated in Fig. S9 through Fig. S17.

It is important to note that the reactants shown here correspond to those recorded in Pistachio [10], including both reactants and reagents. However, certain chemicals listed in the patents may be missing, or additional compounds introduced during work-up might be included. Furthermore, the overall neutralization reaction was not explicitly considered when curating the mechanistic dataset, meaning that some molecules may remain unneutralized. Additionally, since the stronger acid/base species were not prioritized for use in FlowER's design, the model may not predict these cases accurately.

In some cases, species introduced only during work-up are extracted and recorded as reactants in the Pistachio dataset. For example, the patent [21] for the reaction in Fig. S17 describes the addition of saturated aqueous sodium bicarbonate to quench the Grignard reaction. However, the Pistachio records sodium bicarbonate as a primary reactant, rather than as a late-stage quenching agent. As a result, the reaction record appears to contain both a Grignard reagent and a carboxylic acid under the same conditions, which is chemically implausible. Similarly, the presence of acidic and basic species in the same mixture can create mechanistic contradictions, such as attempting Fischer esterification under basic conditions. These inconsistencies arise from how procedural actions in patent text are mapped to reaction inputs and are not a consequence of FlowER's prediction mechanism. While the current dataset retains all recorded components from the patent source, including species introduced during work-up, this approach can result in chemically incompatible input mixtures. In future work, this limitation may be mitigated by using step-aware annotations or curated datasets with more explicit temporal role assignment, such as those available in commercial tools like SciFinder.

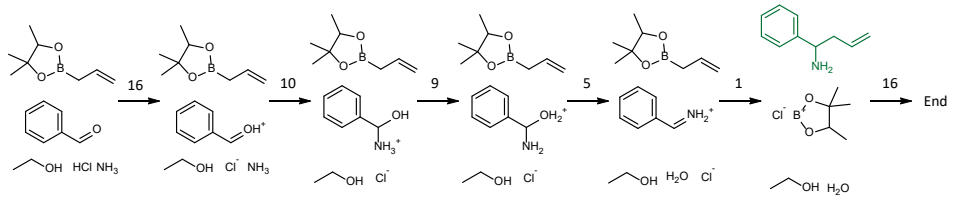

Fig. S9 An example reaction reported in 2024 [20], which was not assigned to a specific reaction class in our dataset [3, 10]. FlowER successfully reproduces the experimentally-recorded product in green. Noteworthy in this example is the proper handling of intermediate protonation states throughout the acid-mediated amination (i.e., formation of an oxonium prior to nucleophilic attack); formation of the iminium in strongly acidic conditions is well-precedented [22]. Direct α-allylation of imines has been described by Alam et al. [23]; a Zimmerman-Traxler six-membered transition state is proposed [24]. The numbers above each arrow represent the number of times that reaction was proposed during 16 independent sampling steps.

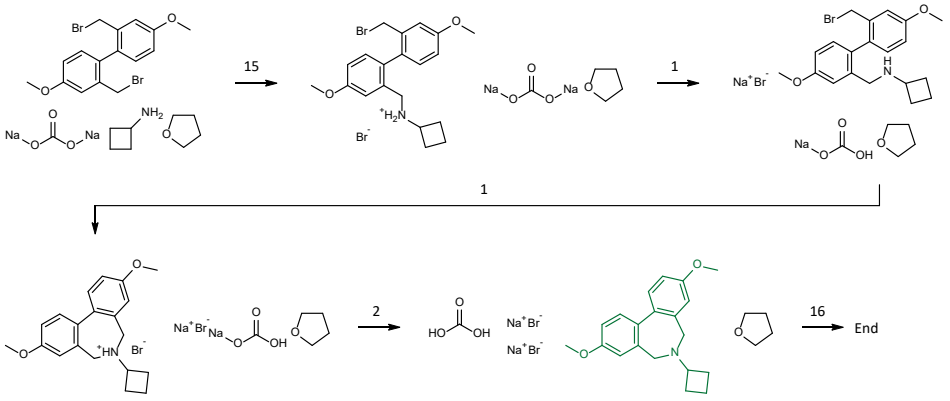

Fig. S10 An example reaction reported in 2024 [11], which was not assigned to a specific reaction class in our dataset [3, 10]. FlowER successfully reproduces the experimentally-recorded product in green. FlowER demonstrates strong performance when handling the first pK_a-dependent substitution (ammonium pK_a= ∼9-11, carbonate pK_a= ∼10.33). However, the second deprotonation (en route to dihydro-dibenzo[c,e]azepine) requires a second equivalent of carbonate to proceed (bicarbonate pK_a= ∼ 6.35). While FlowER has not seen examples of bicarbonate-mediated deprotonations, it selects the best available base—in this case, bicarbonate—to proceed toward the product. This leads to formation of carbonic acid rather than direct generation of CO_2. Since CO_2 is typically produced via the decomposition of carbonic acid, FlowER implicitly models this event as a possible subsequent mechanistic step, but not one en route to the core product structure. This example showcases the need for detailed procedural record-keeping to build next-generation models that are mass-balance- and mechanism-aware. The numbers above each arrow represent the number of times that reaction was proposed during 16 independent sampling steps.

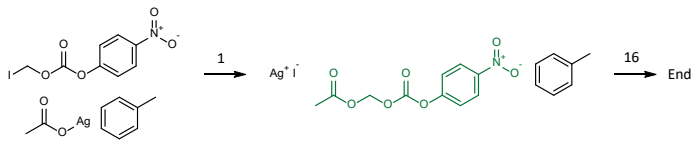

Fig. S11 An example reaction reported in 2024 [12], which was not assigned to a specific reaction class in our dataset [3, 10]. FlowER successfully reproduces the experimentally-recorded product in green. However, while FlowER suggests this transformation follows an S_N2 mechanism, silver salts are well-known to perform halide abstractions [25], [26], driven by the formation of insoluble AgX [27]. Although this mechanism is more likely in this case, FlowER did not receive examples in training that incorporated Ag, yet effectively predicted the outcome of this out-of-distribution transformation. The numbers above each arrow represent the number of times that reaction was proposed during 16 independent sampling steps.

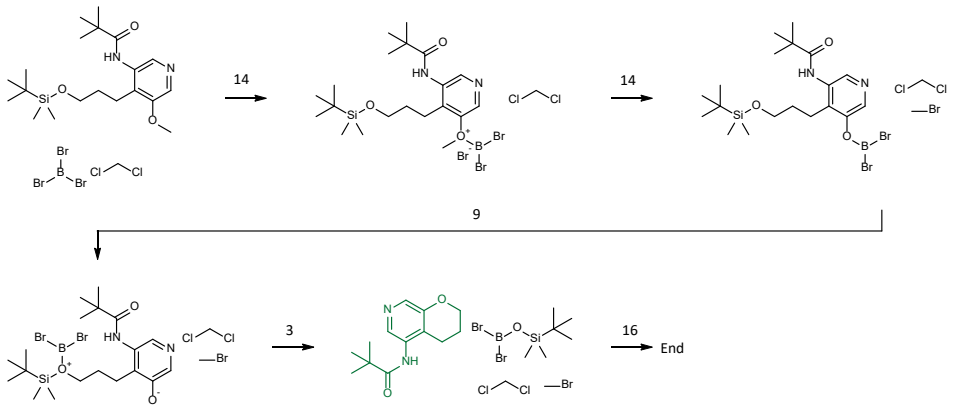

Fig. S12 An example reaction reported in 2024 [13, 14]. FlowER successfully reproduces the experimentally-recorded product in green. Importantly, FlowER captures the commonly accepted unimolecular mechanism for demethylation [28], formalized by McOmie et al. [29] in 1968. Closer inspection of the patent conditions reveals the ground truth conditions use of 3 equivalents of BBr_3. This offers a potential alternative mechanism, wherein the *tert*-butyldimethylsilyl (TBS) ether activation is accomplished with a second equivalent of BBr_3. We note this nuanced mechanism is still actively being researched [30], and several alternative mechanisms have been proposed [31]. The numbers above each arrow represent the number of times that reaction was proposed during 16 independent sampling steps.

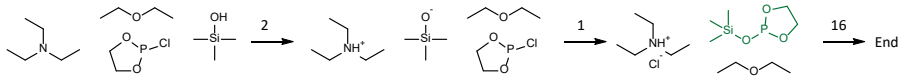

Fig. S13 An example reaction reported in 2024 [15]. FlowER successfully reproduces the experimentally-recorded product in green. We select this particular example to highlight FlowER's handling of a substitution reaction at a non-carbon atom (i.e., the phosphorus of a phosphochloridite). Interestingly, FlowER opts for an anionic mechanism with the trimethylhydroxysilane ($pK_a = \sim 11$) in the presence of TEA (10.8). This suggests the neutral hydroxysilane would operate as the primary nucleophile under these conditions, though the similar relative acidities of both substrates provide an avenue for both mechanistic pathways. The numbers above each arrow represent the number of times that reaction was proposed during 16 independent sampling steps.

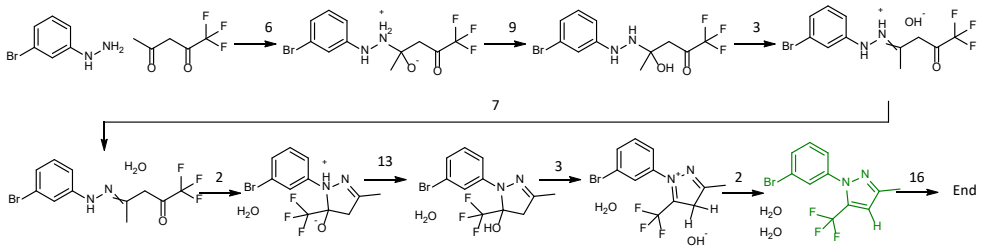

Fig. S14 An example reaction reported in 2024 [16, 17]. FlowER successfully reproduces the experimentally-recorded product in green. We highlight this example to showcase a named reaction that was not identified in Pistachio due to two recorded main products (See the Extended Fig. S2a); Specifically, the acid-mediated Knorr pyrazole synthesis [32]. While the commonly accepted mechanism in strongly acidic solution begins with formation of an oxonoium intermediate [33], we find FlowER exhibits a preference for formation of a zwitterionic intermediate. Nonetheless, FlowER accurately generates the required oxonium en route to both dehydration steps. We note that proton transfers in this mechanism may appear intramolecular, but this is an artifact of our local template encoding: proton transfer steps are modeled locally at the reaction center using pK_a-based rules, without spatial awareness of the intermediate geometry, as discussed in the Supplementary Section S2.2. As such, intermolecular proton shuttling is implied but not explicitly rendered. This is noteworthy, as the formation of hydroxide in strongly acidic conditions can be problematic for other models. Additionally, FlowER accurately predicts the formation of both possible pyrazoles, showcasing the utility of FlowER for exploring mechanistic pathways leading to multiple possible products. The numbers above each arrow represent the number of times that reaction was proposed during 16 independent sampling steps.

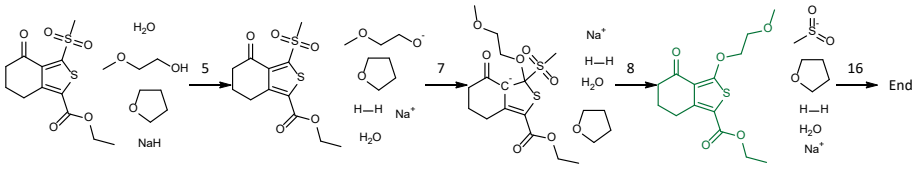

Fig. S15 An example reaction reported in 2024 [18]. FlowER successfully reproduces the experimentally-recorded product in green. In this example, we showcase the ability for FlowER to identify and apply named reaction mechanisms (i.e., sulfone-mediated S_NAr reaction [34]). We highlight the correct handling of terminal alcohols in the presence of NaH, and the subsequent release of methanesulfinate to prepare the final fused (2-methoxyethoxy)thiophene. The numbers above each arrow represent the number of times that reaction was proposed during 16 independent sampling steps.

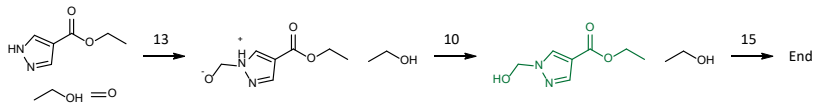

Fig. S16 An example reaction reported in 2024 [19]. FlowER successfully reproduces the experimentally-recorded product in green. Importantly, without the presence of an exogenous base, the pyrazole is correctly proposed to form a zwitterionic intermediate upon nucleophilic attack of formaldehyde (rather than first undergoing deprotonation). As a general note, due to the use of Kekulé representations during preprocessing, the major tautomer is not always correctly identified in instances such as this. This can cause reactivity to occur from an incorrect tautomeric structure. Since tautomerization pathways were not included in the training set, FlowER does not learn to generate or prioritize alternative tautomers, and instead predicts reactivity directly from the input Kekulé structure. The numbers above each arrow represent the number of times that reaction was proposed during 16 independent sampling steps.

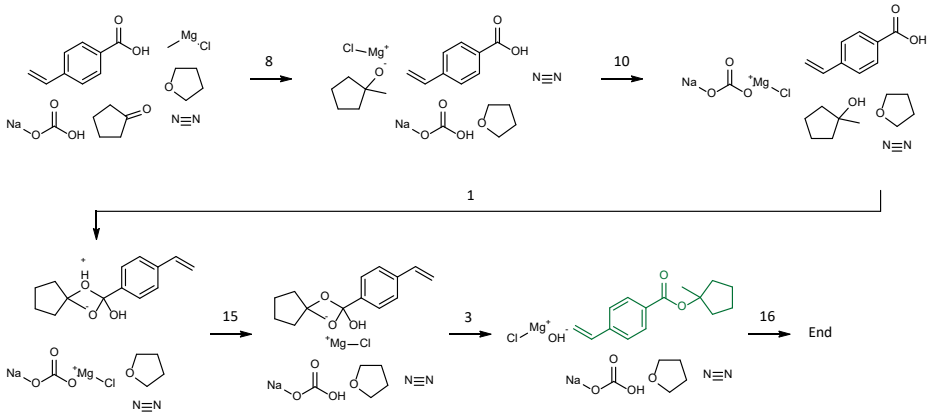

Fig. S17 An example reaction reported in 2024 [21]. FlowER successfully reproduces the experimentally-recorded product in green. This entry was extracted from the Pistachio [10], which consolidates multi-step procedures into a single reaction record. As a result, chemically incompatible reagents (e.g., Grignard reagent and sodium bicarbonate) may appear together, even though they would not coexist experimentally. We highlight this example to showcase how FlowER navigates the entire complex reagent pool—with chemically incompatible reagents—to identify the correct mechanism. While order of addition is undoubtedly of utmost importance in this reaction (i.e., Grignard reagents will rapidly act as a strong base in the presence of a carboxylic acid), FlowER is able to identify a reasonable mechanistic pathway to the observed product. In this particular case, FlowER implicitly identifies the requirement for two subsequent reactions: a Grignard addition and base-mediated esterification. The numbers above each arrow represent the number of times that reaction was proposed during 16 independent sampling steps.

S10 Fine-tuning on previously-unseen reaction types

S10.1 Performance on out-of-distribution reaction types

To evaluate FlowER's ability to generalize to new reaction types, we selected 12 reaction types that were not included in the original training set: Transamidation, Prilezhaev epoxidation, Diels-Alder cycloaddition, Staudinger reduction, Hydroxy to bromo, N-Bn deprotection, Negishi coupling, Carboxylic acid sulfonamide condensation, Menshutkin reaction, Urea Schotten-Baumann, Appel bromination, and SEM protection. These reaction types were curated using the same method for curating our mechanistic dataset as described in Section S3.

For fine-tuning, only 32 overall reactions were used as the training set. The validation set contained at least two overall reactions per reaction type, while the remaining reactions were assigned to the test set.

The best-performing model, selected based on validation set accuracy, was evaluated on the test set. The top-k step and pathway accuracies on the test set are presented in Fig. S19.

The elementary steps of the 12 reaction types and their top-1 step accuracy before and after fine-tuning can be found in Fig. S22 through Fig. S29.

S10.2 Assessing catastrophic forgetting after fine-tuning

To investigate whether fine-tuning caused the model to catastrophically forget reactions learned during the original training phase, we evaluated FlowER and its fine-tuned variants on a subset of the original test set. Specifically, we randomly sampled 10% of the original test set and compared the performance of the original FlowER model with the fine-tuned models.

The top-k step accuracy for FlowER and the fine-tuned models is presented in Fig. S20, while the top-k pathway accuracy is shown in Fig. S21.

Among the 12 fine-tuned models, 11 showed no substantial drop in accuracy compared to the original model, indicating minimal catastrophic forgetting. The exception was the Transamidation fine-tuned model, which exhibited a noticeable performance drop. This suggests that, in most cases, FlowER retains its ability to predict previously learned reactions even after fine-tuning on unseen reaction types.

S10.3 Comparison with Graph2SMILES

To assess how FlowER compares to existing models under low-data settings, we applied the same fine-tuning procedure to G2S on the same 12 reaction types.

The top-1 step and pathway accuracies of G2S before and after fine-tuning are shown in Fig. S18. Overall, G2S exhibited similar levels of accuracy and accuracy retention to FlowER across the 12 reaction types. We observed that G2S produced a lower fraction of chemically valid SMILES and frequently violated heavy atom, hydrogen, or electron conservation—both before and after fine-tuning—as shown in Fig. S18c. These issues persist despite fine-tuning, demonstrating the limitations of sequence-based models when applied to mechanistically structured tasks.

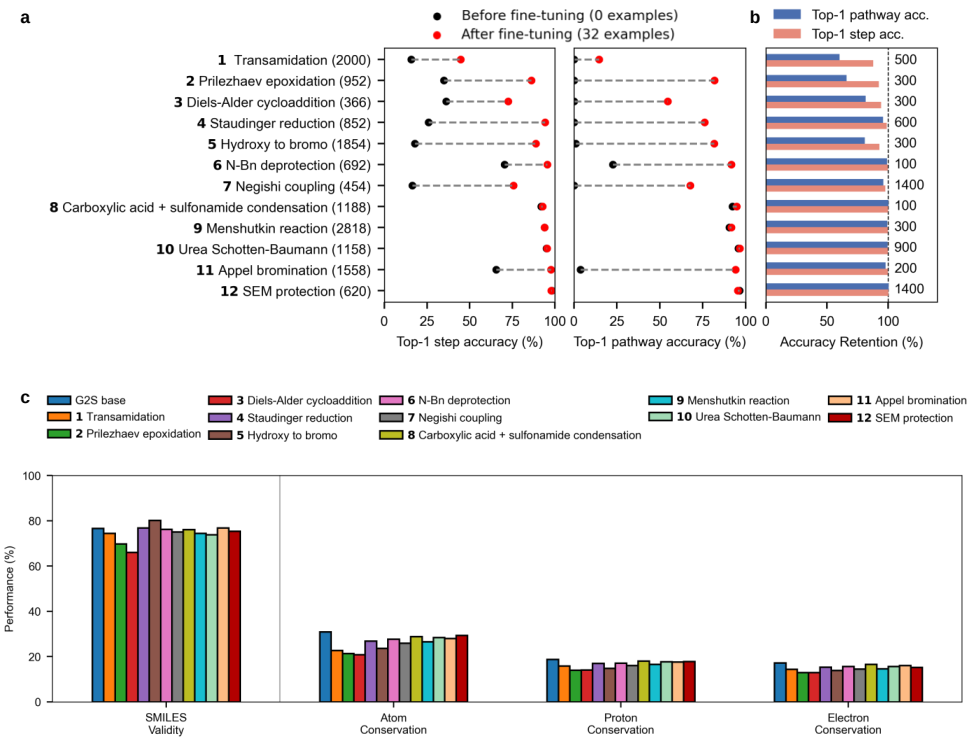

Fig. S18 **a**. Comparison of top-1 step accuracy and top-1 pathway accuracy before and after fine-tuning Graph2SMILES on reaction types absent from the training set. Fine-tuning was conducted using the same method as FlowER, with only 32 overall reactions per reaction type. Each reaction type is annotated with the number of reactions used for evaluation. **b**. Accuracy retention on a 10% random subset of the original test set after Graph2SMILES was fine-tuned on unseen reaction types, normalized to the accuracy of the pretrained model. **c** Validity and conservation performance of Graph2SMILES on a 10% subset of the original test set, before and after fine-tuning.

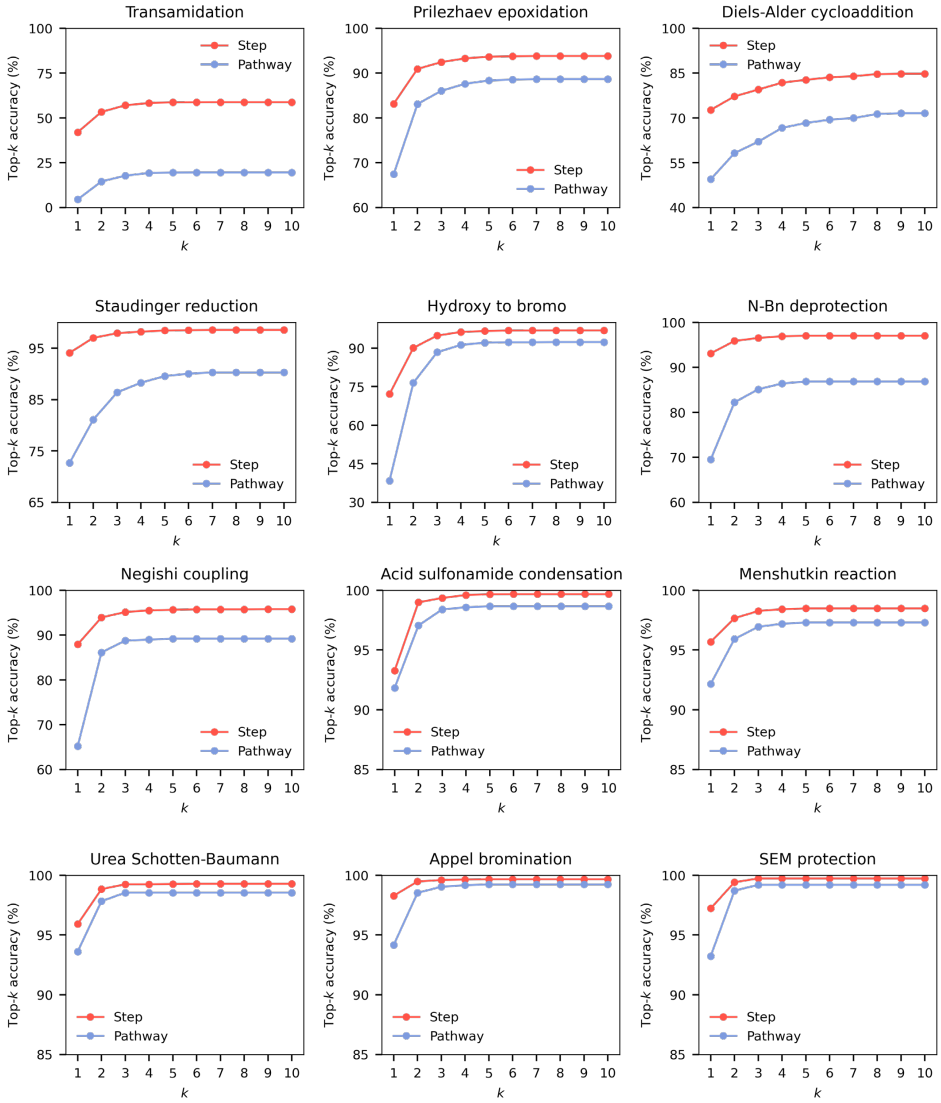

Fig. S19 Top-k accuracy of FlowER on previously unseen reaction types after fine-tuning on 32 examples. Each subplot represents the performance on one of the 12 reaction classes that were excluded from the initial training set. Red and blue curves indicate top-k step accuracy and top-k pathway accuracy, respectively.

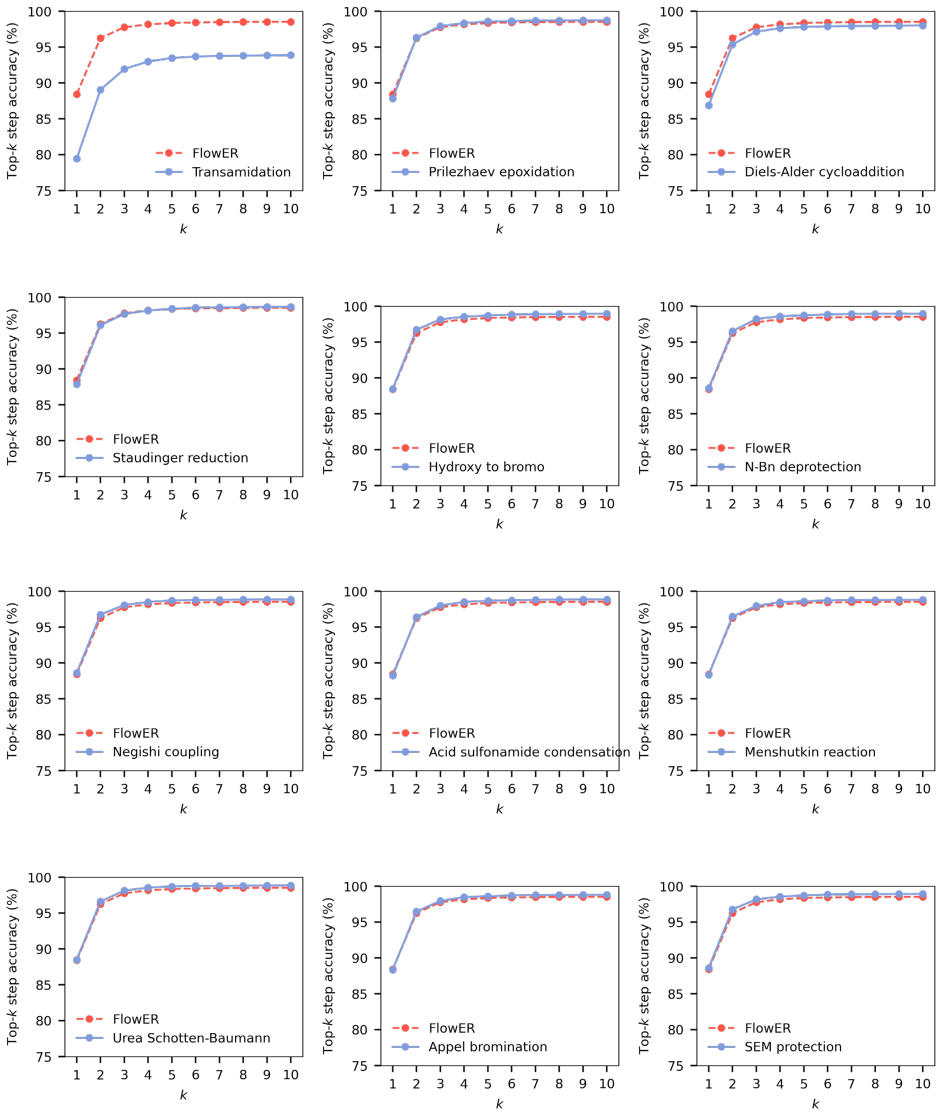

Fig. S20 Performance of fine-tuned models on a random 10% subset of the original test set compared to the original FlowER without fine-tuning, in terms of top-k step accuracy. This evaluation was conducted to assess catastrophic forgetting, where a model forgets previously learned knowledge after fine-tuning on new reaction types. For each fine-tuned model, we recomputed the top-k step accuracy on the sampled test subset and compared it against the original FlowER model.

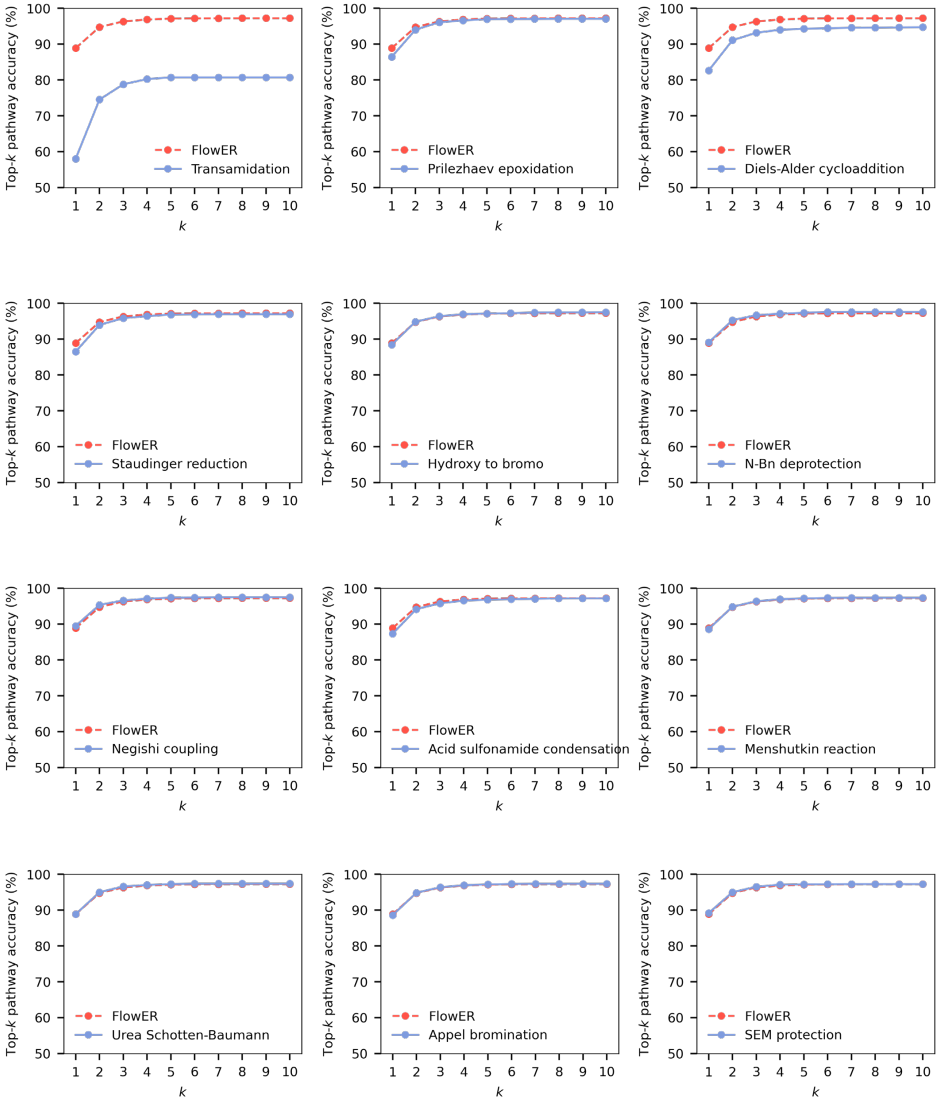

Fig. S21 Performance of fine-tuned models on a random 10% subset of the original test set compared to the original FlowER without fine-tuning, in terms of top-k pathway accuracy. This evaluation was conducted to assess catastrophic forgetting, where a model forgets previously learned knowledge after fine-tuning on new reaction types. For each fine-tuned model, we recomputed the top-k pathway accuracy on the sampled test subset and compared it against the original FlowER model.

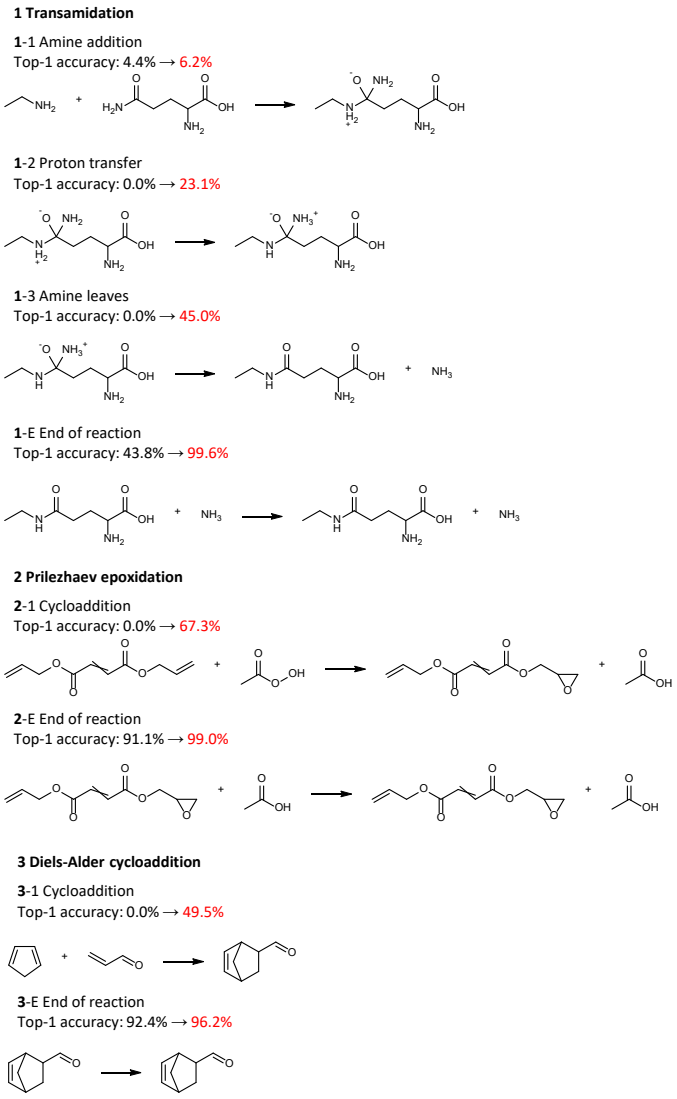

Fig. S22 Changes in top-1 step accuracy for elementary steps of **1** transamidation, **2** Prilezhaev epoxidation, and **3** Diels-Alder cycloaddition before and after fine-tuning on 32 reaction examples. *Note*: Transamidation is known to proceed very slowly under standard conditions [35], [36]. In the case of transamidation of primary amides, the driving force is thought to be loss of ammonia gas [37].

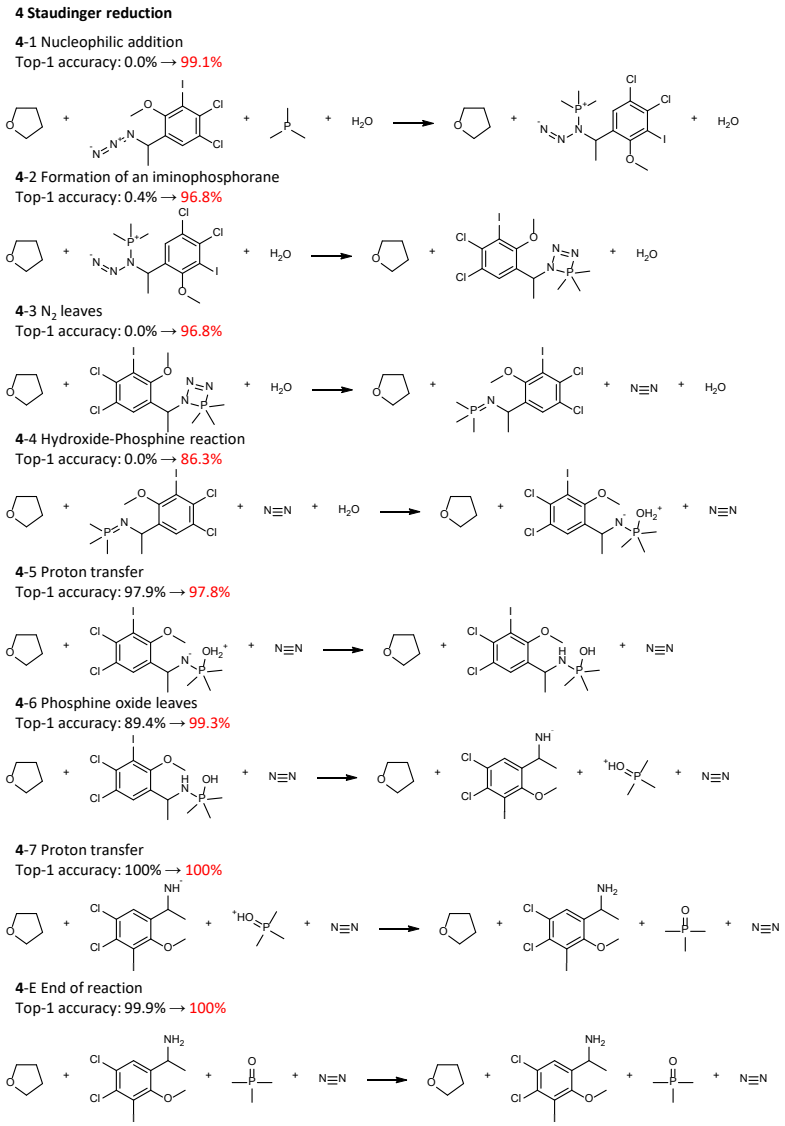

Fig. S23 Changes in top-1 step accuracy for elementary steps of **4** Staudinger reduction before and after fine-tuning on 32 examples. Note: in step 4, the addition of water is thought to be preceeded by deprotonation by the iminophosphorane. See [38] for further reading.

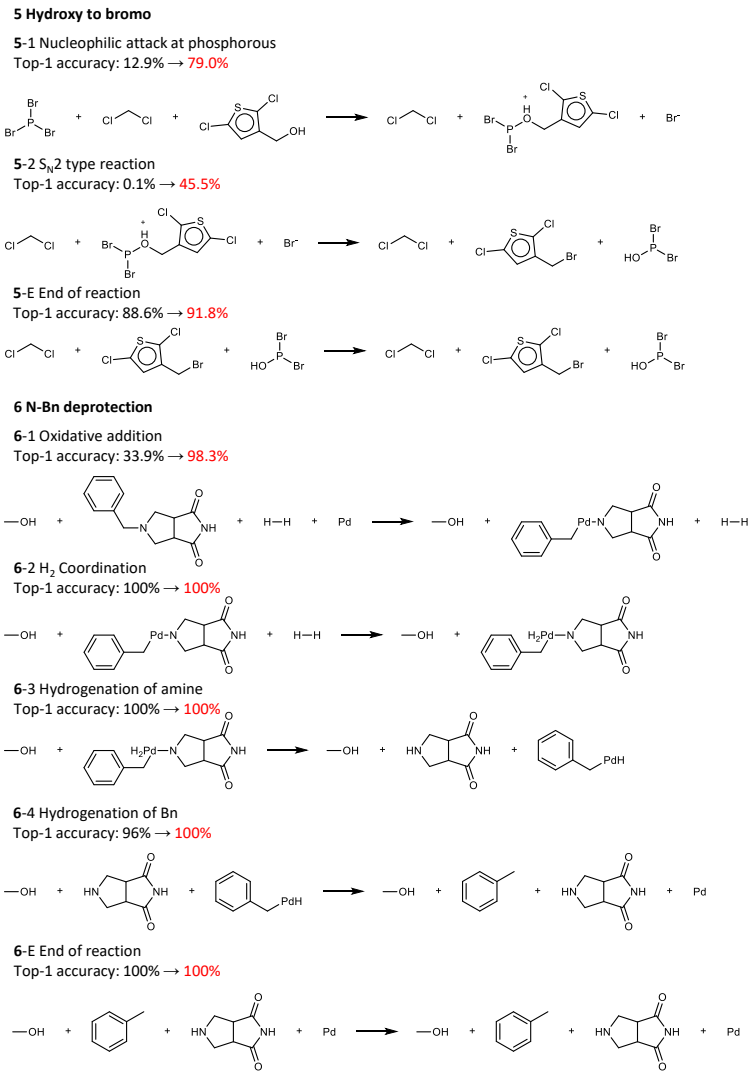

Fig. S24 Changes in top-1 step accuracy for elementary steps of **5** Hydroxy to bromo and **6** N-Bn deprotection before and after fine-tuning on 32 examples.

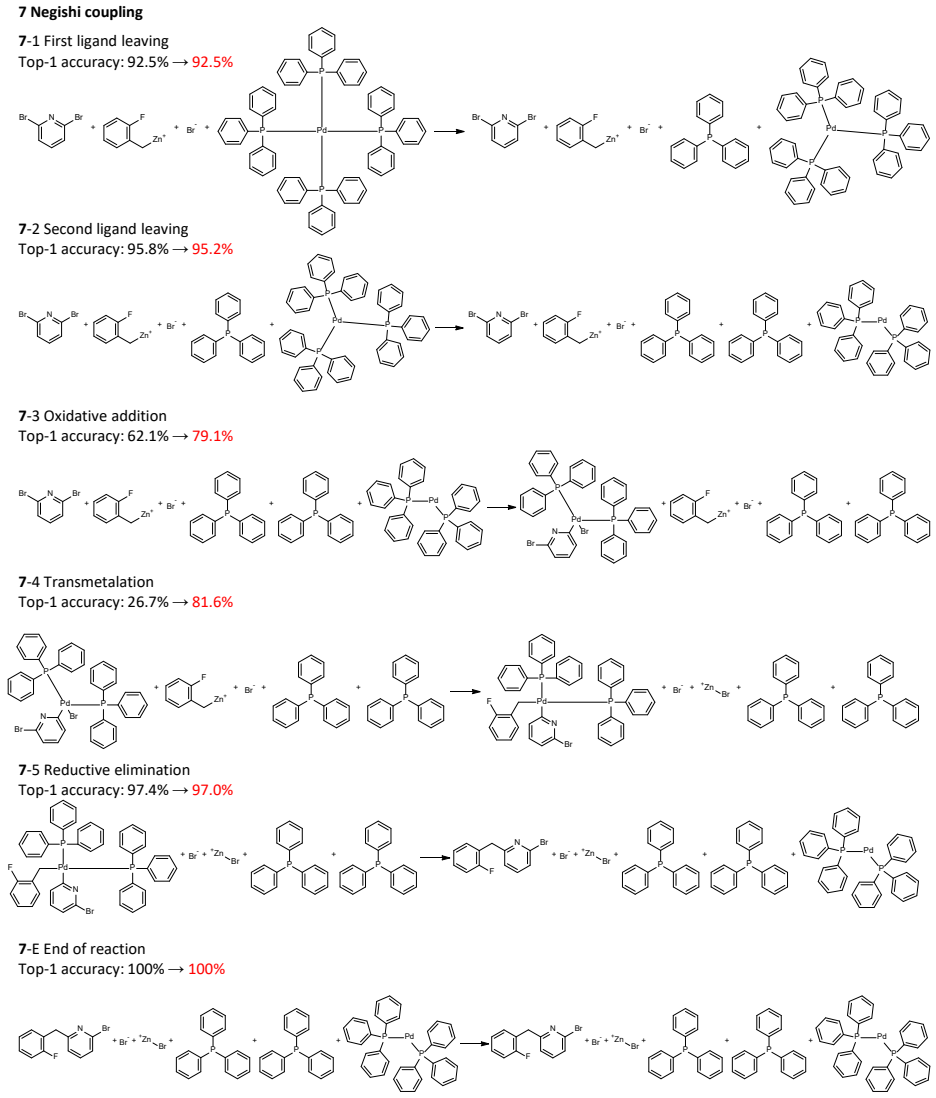

Fig. S25 Changes in top-1 step accuracy for elementary steps of **7** Negishi coupling before and after fine-tuning on 32 examples. *Note*: the halide–zinc bond is not formally shown, and is visualized as a cationic zinc intermediate.

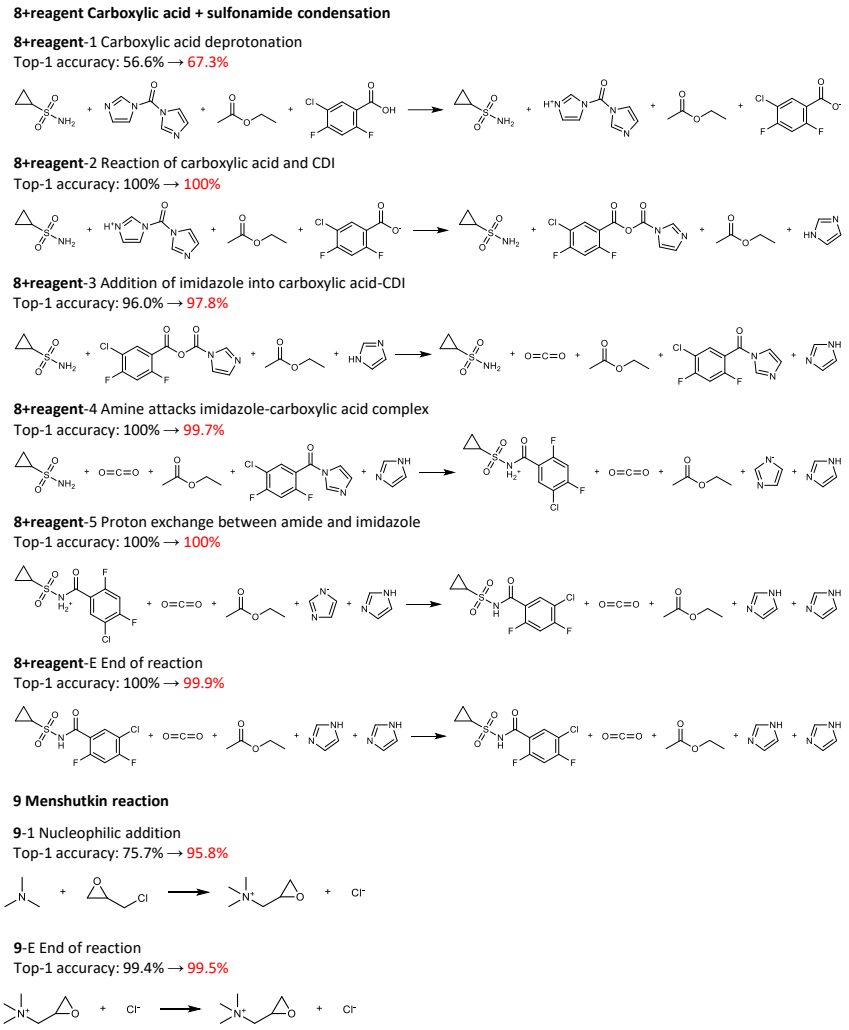

Fig. S26 Changes in top-1 step accuracy for elementary steps of **8** Carboxylic acid + sulfonamide condensation with reagent and **9** Menshutkin reaction before and after fine-tuning on 32 examples. *Note*: For the carboxylic acid + sulfonamide condensation, attack on CDI in Step 2 is stepwise. Similarly, Step 4—wherein the second imidazole is released—is also stepwise. See [39] and [40] for further reading.

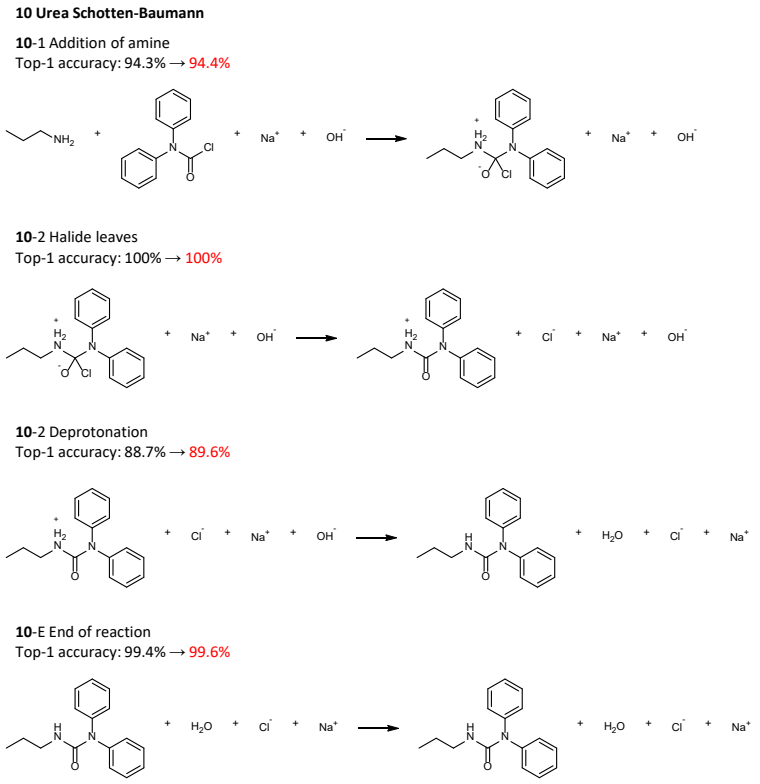

Fig. S27 Changes in top-1 step accuracy for elementary steps of **10** Urea Schotten-Baumann before and after fine-tuning on 32 examples.

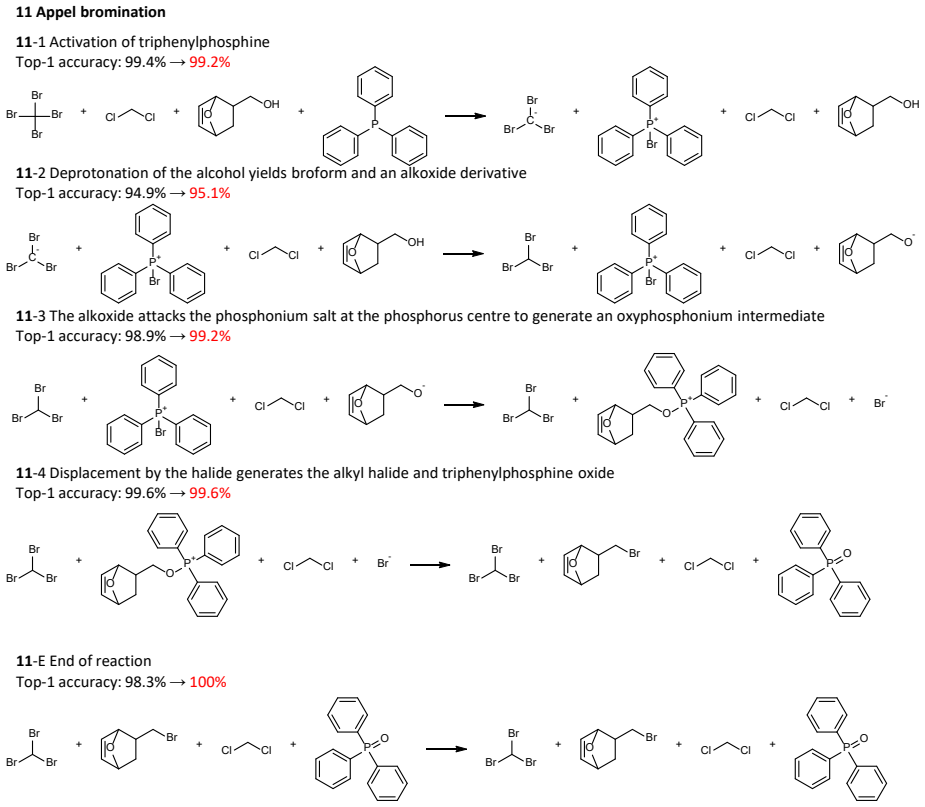

Fig. S28 Changes in top-1 step accuracy for elementary steps of **11** Appel bromination before and after fine-tuning on 32 examples.

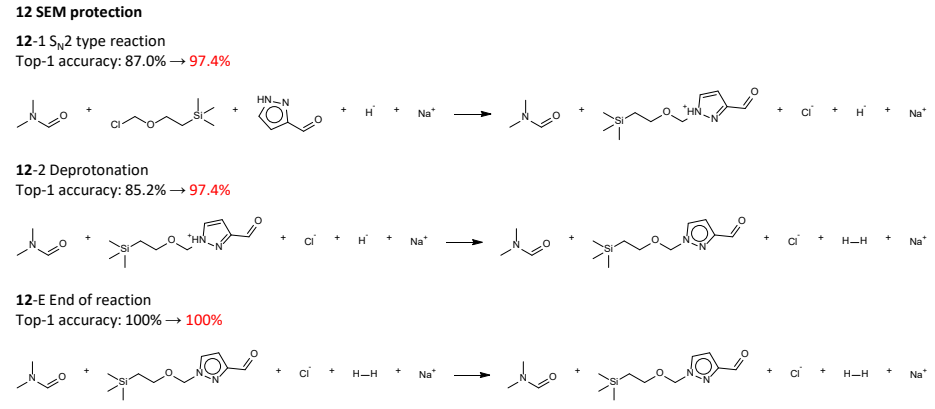

Fig. S29 Changes in top-1 step accuracy for elementary steps of **12** SEM protection before and after fine-tuning on 32 examples. *Note*: NaH is more than sufficiently basic to deprotonate pyrazole and likely proceeds via this alternative pathway.

S11 Retraining FlowER with revised mechanistic templates

S11.1 Revision of mechanistic templates

The mechanistic dataset used to train FlowER in the main text was constructed using the set of reaction templates found at https://github.com/jfjoung/Mechanistic_dataset/blob/main/templates/Reaction_templates_20250216.py. During peer review, several potential improvements in the initial set of templates were brought to our attention, prompting a careful re-evaluation. We identified multiple problems that could compromise mechanistic accuracy or physical realism.

The transamidation example—discussed in the fine-tuning experiments and illustrated in Fig. S22—is one such example. In this case, the reaction proceeds through nucleophilic attack of an amine on an amide carbonyl, followed by intramolecular proton transfer and elimination of the original amide group. Although this mechanism is chemically implausible, similar transformations are more commonly observed in enzymatic systems, where non-covalent interactions and catalysis play a central role. FlowER does not currently account for such catalytic effects or non-covalent stabilization, which remains an open area for future development. The transamidation example in the fine-tuning experiments was left unchanged in the main text.

We sought to prepare a revised version of the mechanistic templates used to impute the mechanistic data that FlowER relies on for training. We expect the mechanistic dataset generated in this study to be used by subsequent work, both by us and by others, and so have tried to ensure that it most faithfully represents the well-accepted textbook mechanisms for the popular reaction types included herein.

First, we identified cases in which a single molecule, often an intermediate, simultaneously contained both a protonated and a deprotonated moiety. The neutralization of such species was modeled as a single elementary step via *intra*molecular proton transfer, without explicit involvement of solvent, acid, or base, implicitly violating realistic mechanistic expectations.

Second, in some reactions involving carbonyl additions, the mechanistically distinct steps of nucleophilic addition and leaving group departure were compressed into a single elementary step. As a result, the formation of the tetrahedral intermediate was bypassed. This compression caused the overall transformation to be modeled as a direct substitution, in which addition and elimination appear to occur simultaneously, which does not follow accepted mechanistic understanding.

Third, several other minor issues were identified across various templates, including inconsistencies in the logical ordering of elementary steps, which in some cases led to ambiguous or chemically implausible mechanistic sequences.

To address these problems, we revised the template set and updated the associated mechanistic dataset as follows:

1. Proton transfer steps that have the end-to-end appearance of an intramolecular transfer are now explicitly represented as intermolecular and proceed over multiple elementary steps.
2. Reactions involving nucleophilic addition to carbonyl groups followed by elimination of a leaving group were split into two distinct elementary steps to better

reflect the formation and collapse of the tetrahedral intermediate. When proton transfers were required to stabilize the tetrahedral intermediate, additional neutralization steps were also introduced.

3. All other issues were systematically corrected under the guidance of a PhD-level organic chemist, ensuring that each template and associated mechanism adheres to chemically sound conventions. A line-by-line revision can be seen in the git history of the templates file in our code repository.

These revisions promote higher mechanistic fidelity and physical plausibility, which are critical for downstream applications such as interpretability, generalization, and integration with quantum chemical calculations.

The most up-to-date version of the mechanistic templates is available at: https://github.com/jfjoung/Mechanistic_dataset/blob/main/templates/Reaction_templates.py. This template library is actively maintained, with ongoing efforts to improve existing entries and expand coverage across broader reaction types. We anticipate that the collection of templates will continue to evolve over time and welcome community feedback, contributions, and pull requests to the repository.

S11.2 Retrained model performance

To retrain FlowER using the revised template set, we regenerated the mechanistic dataset and partitioned it into training, validation, and test splits. The final dataset consisted of 257,159, 2,890, and 28,967 overall reactions, corresponding to 1,933,781, 21,724, and 218,997 elementary steps for training, validation, and testing, respectively. In this new dataset, 31,674 training steps and 2,429 test steps originated from RMechDB [6] and PMechDB [7], which were remapped using RXNMapper [8].

All data used in this study, along with pretrained single-step models, are publicly available on Figshare: https://doi.org/10.6084/m9.figshare.28359407.v3. The dataset is provided in a text format, including reaction SMILES.

Using this updated dataset, we retrained FlowER-large from scratch. The performances of retrained mdoel are summarized in Fig. S30. It should be noted that the top-k step accuracy has a different upper limit in the retrained model, due to increased branching in the revised mechanistic templates. Specifically, several proton transfer steps that were previously modeled as intramolecular have been replaced with intermolecular proton neutralization steps. When multiple acids or bases are available, any of them can serve as the proton donor or acceptor, leading to multiple mechanistically valid branches. This introduces additional diversity in the reaction pathways, which affects the upper limit of the step-wise accuracy.

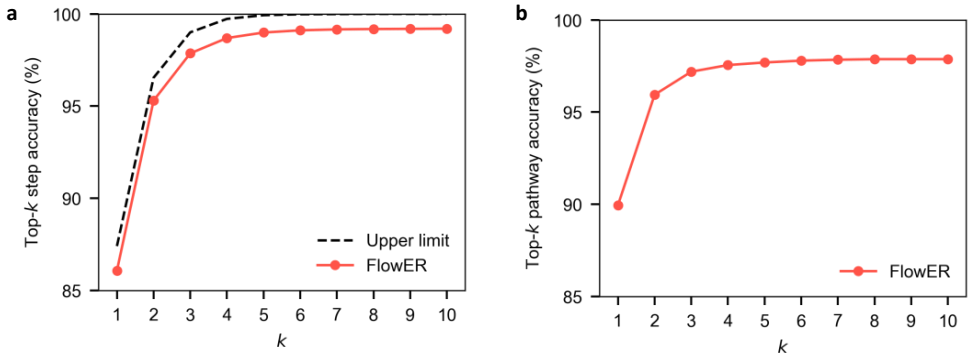

Fig. S30 Performance of FlowER on the revised mechanistic dataset after template correction. **a**. Top-k step accuracy. **b** Top-k pathway accuracy. The quantitative performance is comparable to quantitative performance on the mechanistic dataset reported in the main text. The one-step accuracy is very close to the upper bound, indicating that the data is well-fit by the model.

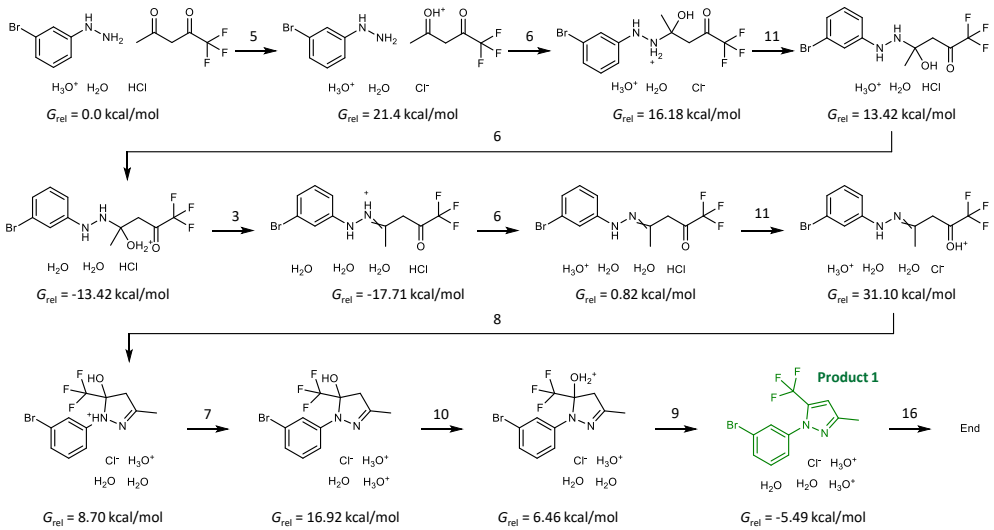

Fig. S31 Prediction of the retrained FlowER from the reactants in Fig. 3b to Product 1 (green). The numbers above each arrow represent the number of times that reaction was proposed during 16 independent sampling steps. Compared to the original proposal of intramolecular proton transfers consistent with the mechanistic dataset used in the main article, the pathway that is now proposed by the retrained FlowER model uses discrete intermolecular proton transfers at each necessary step that are more physically realistic. Further, the order of proton transfers is aligned with organic chemistry principles (i.e., no di-cations are formed). While minor incompatibilities are noted–such as the speciation of neutral water in the presence of HCl–FlowER uses the more strongly acidic species at each proton transfer in line with relative p$K_a$values. We additionally note the final rearomatization step could alternatively form through iminium formation with loss of water rather than the direct elimination sequence proposed.

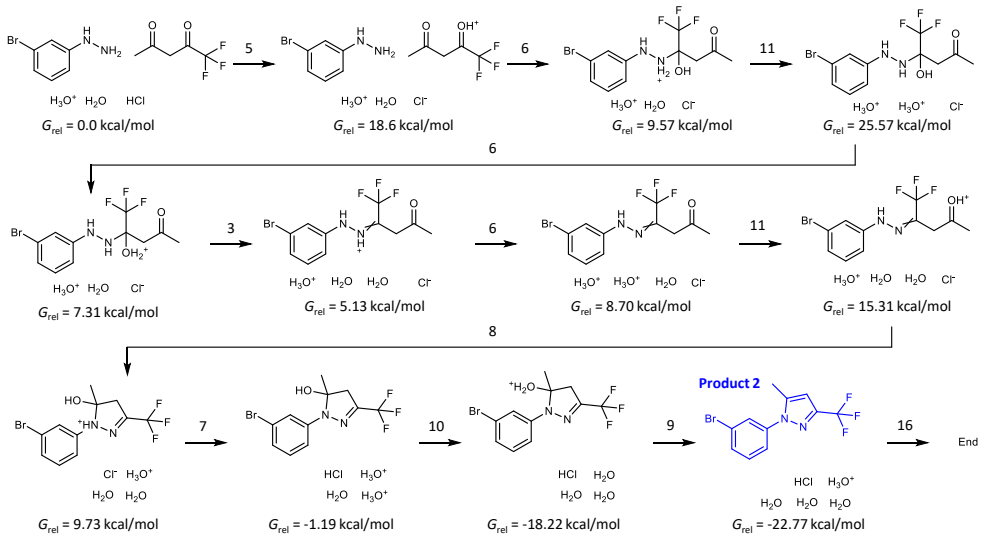

Fig. S32 Prediction of the retrained FlowER from the reactants in Fig. 3b to Product 2 (blue). The numbers above each arrow represent the number of times that reaction was proposed during 16 independent sampling steps. Compared to the original proposal of intramolecular proton transfers consistent with the mechanistic dataset used in the main article, the pathway that is now proposed by the retrained FlowER model uses discrete intermolecular proton transfers at each necessary step that are more physically realistic. Further, the order of proton transfers is aligned with organic chemistry principles (i.e., no di-cations are formed). While minor incompatibilities are noted–such as the speciation of neutral water in the presence of HCl–FlowER uses the more strongly acidic species at each proton transfer in line with relative p$K_a$values. We additionally note the final rearomatization step could alternatively form through iminium formation with loss of water rather than the direct elimination sequence proposed.

References

[1] Dai, H., Li, C., Coley, C., Dai, B., Song, L.: Retrosynthesis prediction with conditional graph logic network. Advances in Neural Information Processing Systems **32** (2019)

[2] Joung, J.F., Fong, M.H., Roh, J., Tu, Z., Bradshaw, J., Coley, C.W.: Reproducing reaction mechanisms with machine-learning models trained on a large-scale mechanistic dataset. Angewandte Chemie International Edition **63**(43), 202411296 (2024)

[3] NameRXN. Nextmove Software. https://www.nextmovesoftware.com/namerxn.html Accessed 2024-02-01

[4] Dugundji, J., Ugi, I.: An algebraic model of constitutional chemistry as a basis for chemical computer programs. In: Computers in Chemistry, pp. 19–64. Springer, Berlin, Heidelberg (1973)

[5] Ugi, I., Bauer, J., Bley, K., Dengler, A., Dietz, A., Fontain, E., Gruber, B., Herges, R., Knauer, M., Reitsam, K., *et al.*: Computer-assisted solution of chemical problems—the historical development and the present state of the art of a new discipline of chemistry. Angewandte Chemie International Edition in English **32**(2), 201–227 (1993)

[6] Tavakoli, M., Chiu, Y.T.T., Baldi, P., Carlton, A.M., Van Vranken, D.: Rmechdb: A public database of elementary radical reaction steps. Journal of chemical information and modeling **63**(4), 1114–1123 (2023)

[7] Tavakoli, M., Miller, R.J., Angel, M.C., Pfeiffer, M.A., Gutman, E.S., Mood, A.D., Van Vranken, D., Baldi, P.: Pmechdb: A public database of elementary polar reaction steps. Journal of Chemical Information and Modeling **64**(6), 1975–1983 (2024)

[8] Schwaller, P., Hoover, B., Reymond, J.-L., Strobelt, H., Laino, T.: Extraction of organic chemistry grammar from unsupervised learning of chemical reactions. Science Advances **7**(15), 4166 (2021)

[9] DeBoer, C.: iteround. GitHub (2018)

[10] NextMove Software: Pistachio. https://www.nextmovesoftware.com/pistachio.html. Accessed: 2020-11-19

[11] Andreux, P., Dugi, K.A., Rinsch, C.L., Greten, F., Denk, D.: Therapeutic Uses of Urolithin Derivatives. US20240139149A1, 2024. https://patents.google.com/patent/US20240139149A1/en

[12] Yang, D., Wong, M.K.: Targeted Delivery of 1,2,4,5-Tetraoxane Compounds

and Their Uses. US20240101584A1, 2024. https://patents.google.com/patent/US20240101584A1/en

[13] Ammann, S.E., Cai, X., Canales, E.Y., Chang, W.K., Chin, G.F., Kinfe, H.H., Lazerwith, S.E., McKinley, J.L., Mish, M.R., Naduthambi, D., Perry, J.K., Rodriguez, K.X., Schroeder, S.D., Swank, C.J., Veldhuizen, J.J.V.: SARS-CoV-2 Main Protease Inhibitors. US20240116946A1, 2024. https://patents.google.com/patent/US20240116946A1/en

[14] Ammann, S.E., Cai, X., Canales, E.Y., Chang, W.K., Chin, G.F., Kinfe, H.H., Lazerwith, S.E., McKinley, J.L., Mish, M.R., Naduthambi, D., Perry, J.K., Rodriguez, K.X., Schroeder, S.D., Swank, C.J., Veldhuizen, J.J.V.: SARS-CoV-2 Main Protease Inhibitors. US20240239807A1, 2024. https://patents.google.com/patent/US20240239807A1/en

[15] Choi, J.Y., Kim, M.G., Lee, S.H., Kang, W.C., Yun, J.C., Han, J.S., Jang, M.J.: Electrolyte and Secondary Battery Including the Same. US20240128510A1, 2024. https://patents.google.com/patent/US20240128510A1/en

[16] Kotian, P.L., Babu, Y.S., Wu, M., Chintareddy, V.R., Kumar, V.S., Zhang, W.: Human Plasma Kallikrein Inhibitors. US20240150295A1, 2024. Accessed: 2024-06-01. https://patents.google.com/patent/US20240150295A1/en

[17] Kotian, P.L., Babu, Y.S., Wu, M., Chintareddy, V.R., Kumar, V.S., Zhang, W.: Human Plasma Kallikrein Inhibitors. US20240150296A1, 2024. Accessed: 2024-06-01. https://patents.google.com/patent/US20240150296A1/en

[18] Xu, Y., Wu, W., Tan, H., Zhang, D., Sun, J., Zhang, Y., Chen, S.: 5,6-Dihydrothieno[3,4-h]quinazoline Compound. US20240140962A1, 2024. https://patents.google.com/patent/US20240140962A1/en

[19] Pasteris, R.J., McMahon, T.C., Wang, H., Alvin Donald Crews, J., Hie, L., Reed, E.W., Chittaboina, S., Reddy, R.P., Uppalapati, S.R., Chen, Y., Vega-Jimenez, B.: Fungicidal Halomethyl Ketones and Hydrates and Their Mixtures. US20240122180A1, 2024. https://patents.google.com/patent/US20240122180A1/en

[20] Shi, Y., Ma, W.: Benzene Ring Compound and Use Thereof. US20240124445A1, 2024. https://patents.google.com/patent/US20240124445A1/en

[21] Fukushima, M., Suzuki, T., Sagehashi, M., Hasegawa, K.: Polymer, Resist Composition, And Patterning Process. US20240118617A1, 2024. https://patents.google.com/patent/US20240118617A1/en

[22] Afanasyev, O.I., Kuchuk, E., Usanov, D.L., Chusov, D.: Reductive amination in the synthesis of pharmaceuticals. Chemical reviews **119**(23), 11857–11911 (2019)

[23] Alam, R., Das, A., Huang, G., Eriksson, L., Himo, F., Szabó, K.J.: Stereoselective allylboration of imines and indoles under mild conditions. an in situ e/z isomerization of imines by allylboroxines. Chemical Science **5**(7), 2732–2738 (2014)

[24] Mejuch, T., Gilboa, N., Gayon, E., Wang, H., Houk, K., Marek, I.: Axial preferences in allylation reactions via the zimmerman–traxler transition state. Accounts of chemical research **46**(7), 1659–1669 (2013)

[25] Mattson, B.M., Graham, W.A.G.: Mechanism of halide abstraction from .eta.5-cyclopentadienyl(dicarbonyl)iodoiron by silver tetrafluoroborate. Inorganic Chemistry **20**(10), 3186–3189 (1981) https://doi.org/10.1021/ic50224a012 . Publisher: American Chemical Society

[26] Pérez-Ortega, I., Albéniz, A.C.: Multifaceted role of silver salts as ligand scavengers and different behavior of nickel and palladium complexes: beyond halide abstraction. Dalton Transactions **52**(5), 1425–1432 (2023) https://doi.org/10.1039/D2DT03948F . Publisher: The Royal Society of Chemistry

[27] Maity, K., Panda, D.K., Lochner, E., Saha, S.: Fluoride-Induced Reduction of Ag(I) Cation Leading to Formation of Silver Mirrors and Luminescent Ag-Nanoparticles. Journal of the American Chemical Society **137**(8), 2812–2815 (2015) https://doi.org/10.1021/ja512020w . Publisher: American Chemical Society

[28] Benton, F., Dillon, T.E.: The cleavage of ethers with boron bromide. i. some common ethers. Journal of the American Chemical Society **64**(5), 1128–1129 (1942)

[29] McOmie, J., Watts, M., West, D.: Demethylation of aryl methyl ethers by boron tribromide. Tetrahedron **24**(5), 2289–2292 (1968)

[30] Kosak, T.M., Conrad, H.A., Korich, A.L., Lord, R.L.: Ether cleavage re-investigated: Elucidating the mechanism of bbr3-facilitated demethylation of aryl methyl ethers. European Journal of Organic Chemistry **2015**(34), 7460–7467 (2015)

[31] Sousa, C., Silva, P.J.: Bbr3-assisted cleavage of most ethers does not follow the commonly assumed mechanism. European Journal of Organic Chemistry **2013**(23), 5195–5199 (2013)

[32] Knorr, L.: Einwirkung von acetessigester auf phenylhydrazin. Berichte der deutschen chemischen Gesellschaft **16**(2), 2597–2599 (1883)

[33] Flood, D.T., Hintzen, J.C., Bird, M.J., Cistrone, P.A., Chen, J.S., Dawson, P.E.: Leveraging the knorr pyrazole synthesis for the facile generation of thioester surrogates for use in native chemical ligation. Angewandte Chemie **130**(36),

11808–11813 (2018)

[34] Patel, N.D., Wei, X., Byrne, D., Narayanan, B.A., Pennino, S., Sarvestani, M., Saha, A., Haddad, N., Kapadia, S., Lorenz, J.C., *et al.*: Sulfone-mediated snar reaction as a powerful tool for the synthesis of 4-quinolinyl ethers and more—application to the synthesis of hcv ns3/4a protease inhibitor bi 201420. The Journal of Organic Chemistry **85**(13), 8339–8351 (2020)

[35] Galat, A., Elion, G.: The Interaction of Amides with Amines: A General Method of Acylation1. Journal of the American Chemical Society **65**(8), 1566–1567 (1943) https://doi.org/10.1021/ja01248a033 . Publisher: American Chemical Society

[36] Bon, E., Bigg, D.C.H., Bertrand, G.: Aluminum Chloride-Promoted Transamidation Reactions. The Journal of Organic Chemistry **59**(15), 4035–4036 (1994) https://doi.org/10.1021/jo00094a004 . Publisher: American Chemical Society

[37] Lanigan, R.M., Sheppard, T.D.: Recent Developments in Amide Synthesis: Direct Amidation of Carboxylic Acids and Transamidation Reactions. European Journal of Organic Chemistry **2013**(33), 7453–7465 (2013) https://doi.org/10.1002/ejoc.201300573 . Accessed 2025-05-05

[38] Pyun, S.-Y., Lee, Y.-H., Kim, T.-R.: Mechanism of the hydrolysis of n-aryliminotriphenylphosphoranes. Kinetics and catalysis **46**, 21–28 (2005)

[39] Montalbetti, C.A.G.N., Falque, V.: Amide bond formation and peptide coupling. Tetrahedron **61**(46), 10827–10852 (2005) https://doi.org/10.1016/j.tet.2005.08.031 . Accessed 2025-04-29

[40] Liu, J., Sato, Y., Yang, F., Kukor, A.J., Hein, J.E.: An Adaptive Auto-Synthesizer using Online PAT Feedback to Flexibly Perform a Multistep Reaction. Chemistry–Methods **2**(8), 202200009 (2022) https://doi.org/10.1002/cmtd.202200009 . Accessed 2025-05-05