Matching-while-Decoding: Enhancing Template-Free Retrosynthesis via Explicit Structural Alignment

Kaipeng Zeng[1,2] **Jiarui Shen**[3,2] **Yi Jiang**[4] **Shiyue Wang**[5,2]
Lin Yao[1,6] **Tong Zhu**[7,8*] **Junchi Yan**[1,2*]

[1]School of Computer Science and School of Artificial Intelligence, Shanghai Jiao Tong University
[2]Shanghai Innovation Institute [3]Department of Chemistry, Fudan University
[4]Global Institute of Future Technology, Shanghai Jiao Tong University
[5]School of Mathematical Sciences, East China Normal University
[6]Zhongguancun Academy
[7]Shanghai Frontiers Science Center of Molecule Intelligent Syntheses,
School of Chemistry and Molecular Engineering, East China Normal University
[8]SII-ECNU Joint Autonomous Chemistry Laboratory, Shanghai Innovation Institute
zengkaipeng@sjtu.edu.cn tzhu@lps.ecnu.edu.cn yanjunchi@sjtu.edu.cn

Abstract

Single-step retrosynthesis prediction serves as the core operational unit for automated synthesis planning and remains a fundamental computational challenge in pharmaceutical design. While a critical prior in organic chemistry is that chemical reactions typically preserve the majority of the molecular scaffold, current template-free deep learning models struggle to exploit this characteristic effectively. Existing methods attempt to leverage this by minimizing sequence-level edit distance through carefully chosen graph traversal orders. However, this structural correspondence is only indirectly induced and remains highly sensitive to imperfect traversal heuristics, which limits its effectiveness for complex transformations. To address this limitation, we introduce the Matching-while-Decoding paradigm, which replaces implicit sequence alignment with direct structural supervision. We operationalize this paradigm within a graph-to-sequence architecture that explicitly maps generated reactant tokens back to the input product graph during decoding. Furthermore, our hierarchical autoregressive formulation utilizes a custom causal attention mask to enable highly efficient single-pass parallel training. Extensive experiments across the USPTO-50K, USPTO-MIT, and USPTO-FULL benchmarks reveal a critical accuracy and robustness trade-off in existing methodologies, where models either over-rely on augmentation or fail to leverage it. Our approach uniquely resolves this bottleneck by delivering highly competitive canonical accuracy while scaling exceptionally well with test-time SMILES augmentation, establishing new state-of-the-art records on USPTO-MIT and augmented USPTO-FULL. By providing this balanced stability across diverse inference protocols, our approach emerges as a highly promising foundation for dependable single-step retrosynthesis prediction. The implementation is available at https://github.com/zengkaipeng/MatchRetro.

1 Introduction

Chemical synthesis plays a foundational role in the chemical industry, particularly in pharmaceutical development. Synthesizing a valuable target compound requires navigating a combinatorially large space of possible synthetic routes while accounting for complex reactivity patterns, selectivity,

*Tong Zhu and Junchi Yan co-supervise this paper and are corresponding authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

and incomplete mechanistic knowledge. This challenge has become increasingly prominent with recent advances in AI-driven drug discovery: while generative models can rapidly propose candidate molecules, their practical utility is often bottlenecked by synthetic feasibility. Consequently, computer-assisted synthesis planning (CASP), a long-standing goal dating back to the seminal work of Corey [6], has received renewed attention. In this paper, we focus on single-step retrosynthesis prediction, a fundamental computational building block of CASP that aims to identify plausible reactant sets capable of producing a target molecule through a single reaction step.

Given the central role of single-step retrosynthesis prediction in CASP, early research sought to leverage established chemical knowledge to reduce the difficulty of the problem. This led to the development of template-based methods [2, 5, 8, 39, 60], which combine expert-curated reaction rules with neural models. These methods cast retrosynthesis as a template prediction problem: given a target product molecule, a neural network predicts the most applicable reaction template from a predefined library, after which rule-based transformation tools [28, 4] apply the selected template to generate reactant candidates. The explicit use of reaction templates provides chemically plausible predictions and strong interpretability. Nevertheless, template-based methods are fundamentally limited by the coverage of their template libraries: reactions outside the predefined template space are difficult to predict, restricting their scalability and generalization to novel transformations.

To overcome the coverage limitations of predefined reaction templates, subsequent studies have shifted toward template-free retrosynthesis prediction, which aims to infer reactants without retrieving a complete reaction template from a fixed library. One representative line of template-free methods introduces a chemically guided decomposition strategy [54, 63, 3, 42, 55]. These methods decompose single-step retrosynthesis into two structured subproblems: reaction center identification, which disconnects the product into synthons, and synthon completion, which attaches suitable leaving groups to recover the reactants. By mimicking the bond-breaking and bond-forming processes underlying chemical reactions, this design better aligns with the chemical nature of retrosynthesis and retains a certain degree of interpretability through explicit reaction centers and synthons. However, its stage-wise formulation may suffer from error accumulation and can become less reliable for complex reactions involving multiple bond changes or multiple reaction centers.

Another line of template-free methods directly generates reactants from the target molecule in an end-to-end manner. Some models generate reactants through autoregressive molecular graph edits [36, 70], while others use non-autoregressive formulations such as Markov bridges, discrete flow matching, and categorical diffusion [16, 62, 53, 7]. Most approaches still formulate retrosynthesis as SMILES-based sequence generation [31, 46, 52, 47]. Pretraining and reaction-description representations have also been explored to address the data bottleneck [21, 9, 61]. Despite its flexibility, this formulation must regenerate the entire reactant sequence, overlooking a key structural prior of single-step retrosynthesis: most atoms and substructures are preserved between products and reactants, with only localized graph transformations. Existing methods, therefore, introduce SMILES-level alignment strategies to reduce product–reactant discrepancies [40, 52, 71, 68]. However, such alignment remains implicit and traversal-dependent. Since SMILES linearizes molecular graphs through depth-first traversals, finding a reactant traversal that respects product-reactant atom correspondence and minimizes the edit distance to a given product traversal requires searching over many valid orderings. This can leave substantial sequence-level discrepancies in complex reactions, hindering the reliable reuse of preserved molecular fragments.

In this work, we introduce an explicit structural-alignment formulation for template-free retrosynthesis prediction. Instead of relying solely on SMILES-level alignment heuristics, our formulation explicitly organizes the decoding process around reusable product-derived fragments and newly generated reaction-specific components. To make this formulation practical, we design a hierarchical autoregressive decoding architecture along with a tailored sequence construction and causal masking strategy, enabling efficient parallel training while preserving the desired generation order. Empirically, our model achieves robust and competitive performance across inference regimes: it maintains strong accuracy under canonical SMILES inference, scales consistently with test-time SMILES augmentation, and obtains strong results on chemically grounded metrics such as round-trip accuracy. These results suggest that explicit structural alignment provides a complementary and effective inductive bias for pushing the boundaries of template-free retrosynthesis prediction.

Our contributions are three-fold. **First,** we introduce an explicit atom-provenance prediction mechanism for template-free autoregressive SMILES-based retrosynthesis prediction. By augmenting

SMILES generation with one-to-one correspondence prediction between generated atoms and product atoms, our approach improves structural grounding while retaining the ability of SMILES-based generation to represent stereochemistry. **Second,** we propose a hierarchical autoregressive decoding architecture with a tailored sequence construction and causal masking scheme, which makes the proposed aligned generation process amenable to efficient parallel training. **Third,** extensive experiments demonstrate that our method delivers robust performance across inference settings, combining strong canonical inference, effective test-time SMILES augmentation scaling, and competitive chemically grounded prediction quality.

2 Preliminaries and Problem Formulation

2.1 Molecular Graph Representation

We represent a product molecule as an attributed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where nodes $v_i \in \mathcal{V}$ correspond to heavy atoms and edges $e_{ij} \in \mathcal{E}$ represent chemical bonds. Each node is initialized with standard atomic descriptors, such as atom type, formal charge, degree, hybridization, and aromaticity, and each edge is associated with bond features. The corresponding reactants are represented as a set of molecular graphs $\mathcal{R} = \{\mathcal{G}_{R_1}, \ldots, \mathcal{G}_{R_l}\}$.

2.2 Sequence Parsing and Token Typology

We formulate single-step retrosynthesis as a graph-to-sequence translation problem, where decoding the target reactants yields a token sequence $\mathbf{t} = [t_1, t_2, \ldots, t_N]$ with $t_i \in \Sigma$. To parse reactant SMILES strings, we adopt the domain-specific tokenizer used in the Molecular Transformer [38], which separates SMILES into chemically meaningful tokens. The resulting vocabulary Σ is divided into two disjoint subsets:

- **Atom Tokens:** Σ_{atom}, whose tokens correspond to physical heavy atoms in the molecular graph, such as 'C', 'N', 'O', and 'Cl'.
- **Symbol Tokens:** Σ_{symbol}, which encode SMILES syntax, topology, or stereochemistry without representing atoms, such as brackets, bond symbols, and ring closure digits.

2.3 Problem Formulation

Given an input product graph $\mathcal{G}$, standard graph-to-sequence retrosynthesis learns a generative model that predicts the reactant SMILES token sequence $\mathbf{t}$. In contrast, our formulation jointly models the token sequence and its structural correspondence to the input graph. Specifically, for each generated token t_i, we introduce a structural provenance label p_i that indicates whether and where the token is inherited from $\mathcal{G}$. The learning objective is to model:

$$P(\mathbf{t}, \mathbf{p} \mid \mathcal{G}) = \prod_{i=1}^{N} P(t_i, p_i \mid t_{<i}, p_{<i}, \mathcal{G}). \tag{1}$$

Here, p_i maps t_i to a node $v \in \mathcal{V}$ when $t_i \in \Sigma_{\text{atom}}$, and the token is inherited from the product graph; otherwise, p_i is assigned a predefined null state that covers symbol tokens and newly introduced atoms, such as leaving-group components.

3 Methodology

Single-step retrosynthesis exhibits a strong structural prior: most product-derived atoms and substructures are preserved in the reactants, while only localized regions are transformed. SMILES-level alignment methods exploit this prior by reducing product-reactant sequence discrepancies through carefully chosen traversal orders. However, such correspondence is only indirectly induced by sequence alignment and can be sensitive to imperfect traversal heuristics, especially for complex structural transformations. To address this limitation, we introduce MatchRetro, a **Matching-while-Decoding** framework that augments autoregressive reactant generation with explicit structural matching: each generated token is decoded together with a structural provenance label that maps it to the input product graph when applicable. An overview of MatchRetrois shown in Fig. 1.

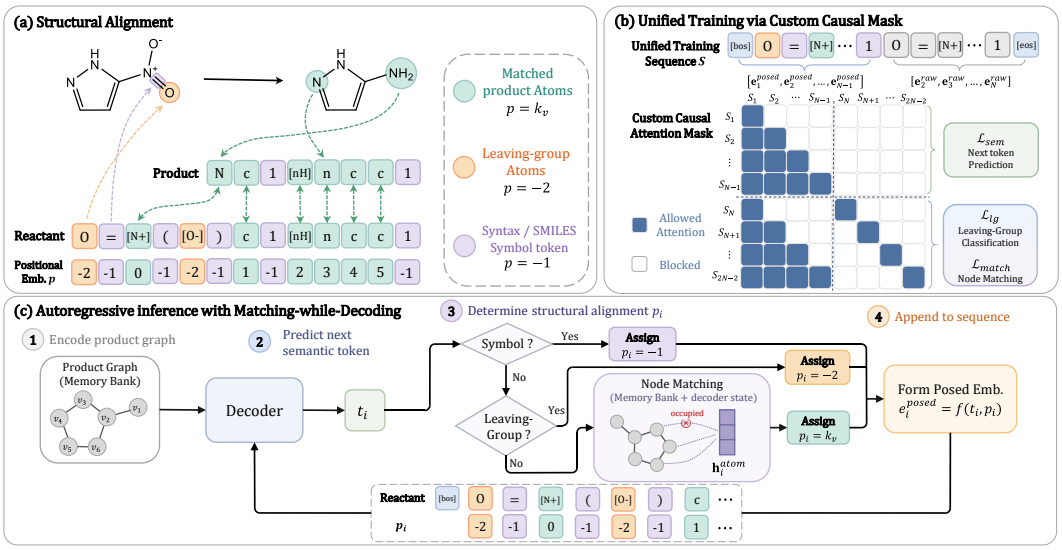


Figure 1: Overview of MatchRetro **a)** Example of explicit structural alignment. **b)** Visual illustration of the causal mask for parallel training. **c)** Pipeline of hierarchical autoregressive generation.

3.1 The Matching-while-Decoding Paradigm

Building on the formulation in Sec. 2, we decompose the joint prediction of each token unit into semantic token generation and structural matching. Let t_i denote the token generated at step i, and let p_i denote its structural provenance, which either maps t_i to a product atom in $\mathcal{G}$ or assigns it to a predefined null state. We factorize the conditional distribution as:

$$P(t_i, p_i \mid t_{<i}, p_{<i}, \mathcal{G}) = P(t_i \mid t_{<i}, p_{<i}, \mathcal{G})\, P(p_i \mid t_{\leq i}, p_{<i}, \mathcal{G}). \tag{2}$$

This factorization forms a conditional prediction cascade: the decoder first predicts the identity of the next token and then predicts its structural provenance conditioned on the newly generated token. Thus, the model is trained to generate chemically valid reactant sequences while explicitly identifying which components are inherited from the product structure. We next describe the structure-aware memory bank for $\mathcal{G}$ and the two conditional predictors.

3.2 Structure-Aware Product Encoding

To parameterize the conditioning on $\mathcal{G}$, we encode the input product graph into a structure-aware memory bank $\mathbf{H}_{\text{prod}}$. The memory bank must provide both chemically informative atom representations and deterministic structural indices for matching. Given $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, we first encode the atom and bond features with a bond-aware graph encoder instantiated as EGAT$^+$ [68]:

$$\mathbf{H}' = \text{EGAT}^+(\mathbf{X}, \mathbf{E}), \tag{3}$$

where $\mathbf{X}$ and $\mathbf{E}$ denote the initial atom and bond features, and $\mathbf{H}'$ denotes the context-aware atom representations.

We then assign each product atom $v \in \mathcal{V}$ a deterministic Depth-First Search positional index $k_v \in \{0, \ldots, |\mathcal{V}| - 1\}$ according to a designated DFS traversal. The final product memory bank is obtained by adding the corresponding DFS positional embedding:

$$\mathbf{H}_{\text{prod}} = \mathbf{H}' + \text{Emb}_{\text{DFS}}(\mathbf{k}), \tag{4}$$

where $\mathbf{k}$ collects the DFS indices of all product atoms.

The DFS positional encoding plays a dual role. It defines the target label space for structural matching and, when paired with the alignment heuristic used to construct training targets [68, 71], deterministically specifies the aligned reactant sequence associated with each augmented product view. This allows our graph-based encoder-decoder architecture to benefit from randomized SMILES augmentation [46] while maintaining explicit product-reactant correspondence.

3.3 Dual-Positional Decoder Representations

We employ a Transformer decoder [51] to translate the structural memory bank $\mathbf{H}_{\text{prod}}$ into the target reactant sequence. Our matching-while-decoding paradigm requires the decoder to maintain strict sequential order and explicit structural awareness simultaneously. We establish this through a dual-embedding strategy that separates spatial origin from the generation sequence.

For structural awareness, we introduce a learnable structural position embedding matrix $\mathbf{E}_{pos}$ indexed by p_i. To explicitly encode the spatial origin of each generated semantic token t_i, its structural position p_i is assigned as follows:

$$p_i = \begin{cases} -1, & \text{if } t_i \in \Sigma_{symbol} \\ -2, & \text{if } t_i \in \Sigma_{atom} \text{ and acts as a leaving group} \\ k_v, & \text{if } t_i \in \Sigma_{atom} \text{ and matches product node } v \end{cases} \tag{5}$$

Here, $p_i = -1$ indicates that the token lacks physical node correspondence, $p_i = -2$ denotes an atom that is absent from the product graph, and $p_i = k_v \geq 0$ maps the token directly to the deterministic DFS-PE of its corresponding product node. For a semantic token t_i at step i, we define the raw token embedding as $\mathbf{e}_i^{raw} = \text{Emb}_{sem}(t_i)$. The posed token embedding then incorporates the structural context via the element-wise sum $\mathbf{e}_i^{posed} = \mathbf{e}_i^{raw} + \text{Emb}_{pos}(p_i)$.

Because the posed state already uses additive embeddings for structural grounding, introducing another absolute sequential position embedding could entangle the generation order with the spatial origin. We therefore employ Rotary Position Embedding [44] in the self-attention layers so that sequential order is encoded through attention dynamics rather than through an additional additive embedding. At each generation step i, the decoder processes the historical sequence of posed token embeddings together with the structural memory bank $\mathbf{H}_{\text{prod}}$, producing a context-aware hidden state $\mathbf{h}_i \in \mathbb{R}^d$ that serves as the shared representation for the subsequent conditional prediction cascade over token identity and structural position.

3.4 Hierarchical Autoregressive Generation

To solve the joint decoding and alignment problem, we implement a conditional prediction cascade. At generation step i, the model executes the hierarchical factorization by dynamically passing the sequence of autoregressive hidden states through up to three specialized prediction modules in a strictly conditional sequence.

Semantic Prediction. First, we feed the historical prefix of posed token embeddings $\mathbf{P}_{<i} = [\mathbf{e}_1^{posed}, \ldots, \mathbf{e}_{i-1}^{posed}]$ into the Transformer decoder. We extract the hidden state corresponding to the final token in this prefix at step $i-1$, denoted as $\mathbf{h}_{i-1}^{ctx} \in \mathbb{R}^d$. We pass this context vector through a two-layer MLP, denoted as MLP_{sem}, to predict the semantic identity of the next token t_i over the vocabulary Σ:

$$P(t_i \mid t_{<i}, p_{<i}) = \text{Softmax}(\text{MLP}_{sem}(\mathbf{h}_{i-1}^{ctx})) \tag{6}$$

Leaving Group Classification. Second, if the generated token represents a physical atom where $t_i \in \Sigma_{atom}$, we must determine its structural origin. We append the newly generated raw embedding $\mathbf{e}_i^{raw}$ to the historical prefix, yielding the extended sequence $[\mathbf{P}_{<i}, \mathbf{e}_i^{raw}]$. We feed this extended sequence back into the decoder and extract the final hidden state corresponding to the newly appended $\mathbf{e}_i^{raw}$, denoted as $\mathbf{h}_i^{atom}$. This state is evaluated by a binary classifier, denoted as MLP_{lg}, to determine the probability that the atom acts as a leaving group under the specific condition $p_i = -2$:

$$P(p_i = -2 \mid t_{\leq i}, p_{<i}) = \sigma(\text{MLP}_{lg}(\mathbf{h}_i^{atom})) \tag{7}$$

where σ denotes the sigmoid activation function. If the semantic prediction yields a symbol token where $t_i \in \Sigma_{symbol}$, this classification step is bypassed, and p_i is deterministically assigned as -1.

Structural Matching. Finally, if the atom token is predicted not to be a leaving group, it must be aligned with a specific node in the product graph. We compute a matching score $s_{i,v}$ between the atom's hidden state $\mathbf{h}_i^{atom}$ and each product node representation $\mathbf{h}_v \in \mathbf{H}_{\text{prod}}$. We achieve this

by concatenating the features and passing them through a dedicated matching network, denoted as MLP_{match}, to yield a scalar logit:

$$s_{i,v} = \text{MLP}_{match}([\mathbf{h}_i^{atom} \parallel \mathbf{h}_v]) \tag{8}$$

where $\parallel$ denotes concatenation. We compute the probability of matching to a specific product node, corresponding to the structural state $p_i = k_v$, via a softmax function over all $|\mathcal{V}|$ nodes:

$$P(p_i = k_v \mid t_{\leq i}, p_{<i}, p_i \neq -2) = \frac{\exp(s_{i,v})}{\sum_{u \in \mathcal{V}} \exp(s_{i,u})} \tag{9}$$

During inference, this cascade is augmented with deterministic structural constraints to ensure chemical validity. For details, refer to Appendix E.2.

3.5 Unified Training via Custom Causal Masking

The hierarchical generation cascade necessitates two distinct visibility paradigms. Predicting the next semantic token t_i relies solely on the historical prefix up to step $i-1$. Conversely, predicting the structural alignment p_i and leaving group status requires this same prefix alongside the newly generated semantic token at step i. Standard shift-right causal masking cannot accommodate this dual-visibility requirement in a single pass.

To enable efficient parallel training, we propose a custom concatenated attention mask. We bound the target sequence of length N with special tokens $t_1 =$ <bos> and $t_N =$ <eos>. Since the deterministic seed t_1 initiates generation, semantic prediction starts at step 2. The unified training sequence $\mathbf{S}$ is constructed by concatenating the posed embedding sequence with the raw token embedding sequence:

$$\mathbf{S} = [\mathbf{e}_1^{posed}, \ldots, \mathbf{e}_{N-1}^{posed}, \ \mathbf{e}_2^{raw}, \ldots, \mathbf{e}_N^{raw}]$$

Within the Transformer decoder, the self-attention matrix enforces the hierarchical autoregressive order without label leakage through two visibility rules. First, a posed token embedding $\mathbf{e}_i^{posed}$ attends exclusively to preceding posed tokens $\{\mathbf{e}_j^{posed} \mid j \leq i\}$. Second, a raw token embedding $\mathbf{e}_k^{raw}$ attends only to strictly preceding posed tokens $\{\mathbf{e}_j^{posed} \mid j < k\}$ and to itself $\mathbf{e}_k^{raw}$. This specific formulation for raw tokens directly prevents the leakage of structural ground truth while enforcing strict training-inference consistency. A visual illustration is presented in Fig. 1(b).

3.6 Training Strategy and Optimization

To establish a robust structural prior and optimize the hierarchical generation cascade, we employ a two-stage training strategy.

Pre-training Phase. Before modeling reaction transformations, we pre-train the model on a standard graph-to-SMILES translation task using non-reaction molecular data. Because this phase models isolated molecules, the leaving group state is inherently absent. The matching objective $\mathcal{L}_{match}$ explicitly aligns each generated atom token to its corresponding physical node within the input molecular graph. We minimize the simplified objective:

$$\mathcal{L}_{pre} = \mathcal{L}_{sem} + \gamma \mathcal{L}_{match} \tag{10}$$

where γ is a scaling hyperparameter. This initialization equips the decoder with fundamental chemical syntax and a baseline structural grounding.

Unified Reaction Fine-Tuning. During the primary retrosynthesis phase, we fine-tune the model end-to-end using the custom causal mask defined in Sec. 3.5. To balance the primary autoregressive generation with auxiliary structural supervision, the composite objective is formulated as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{sem} + \alpha \mathcal{L}_{lg} + \beta \mathcal{L}_{match} \tag{11}$$

where α and β are empirical scaling hyperparameters.Each loss component is conditionally masked to target specific modules within the prediction cascade. $\mathcal{L}_{sem}$ is the standard cross-entropy loss applied universally across all generation steps. $\mathcal{L}_{lg}$ is a binary cross-entropy loss applied exclusively to physical atom tokens $t_i \in \Sigma_{atom}$. $\mathcal{L}_{match}$ is a multi-class cross-entropy loss computed over the $|\mathcal{V}|$ product nodes and is applied strictly to tokens with valid structural correspondences $p_i \geq 0$.

4 Experimental Results

We evaluate MatchRetro on three standard single-step retrosynthesis benchmarks: USPTO-50K, USPTO-MIT, and USPTO-FULL. We first report top-k exact-match accuracy under both canonical and augmented inference settings, following standard evaluation practice. We then complement exact-match evaluation with chemical validity analyses, including round-trip validation and group-wise statistics across different reaction subsets. Together, these results assess both the model's ability to recover ground-truth reactants and the chemical plausibility of its generated retrosynthetic candidates.

4.1 Single-Step Retrosynthesis Accuracy

We compare MatchRetro with representative template-based and template-free baselines. Since test-time SMILES augmentation (TTA) is widely used but inconsistently reported, we distinguish canonical inference from inference with TTA. The top-k exact-match results are summarized in Tab. 1 and Tab. 2. TTA can change the relative rankings of strong template-free methods, including UAlign, RSmiles, EditRetro, and NAG2G. We first examine canonical inference to assess intrinsic predictive strength, and then analyze how TTA affects relative performance. Detailed five-seed statistics and component ablations for our USPTO-50K models are provided in Appendix F.5.

Table 1: Top-k accuracy (%) on USPTO-50K (reaction class unknown) and USPTO-FULL. *: TTA. **Bold**: best; underlined: second best, per group (TB, TF w/o TTA, TF w/ TTA) on each dataset. TB: template-based; TF: template-free.

Type	Model	Top-k accuracy (%)							
		USPTO-50K				USPTO-FULL			
		1	3	5	10	1	3	5	10
TB	RetroSim [5]	37.3	54.7	63.3	74.1	32.8	–	–	56.1
	NeuralSym [39]	44.4	65.3	72.4	78.9	35.8	–	–	60.8
	GLN [8]	52.5	69.0	75.6	83.7	**39.3**	–	–	**63.7**
	LocalRetro [2]	53.4	77.5	85.9	92.4	39.1	**53.3**	**58.4**	**63.7**
	RetroKNN [60]	**57.2**	**78.9**	**86.4**	**92.7**	–	–	–	–
TF	GraphRetro [43]	53.7	68.3	72.2	75.5	–	–	–	–
	RetroPrime [54]	51.4	70.8	74.0	76.1	44.1	59.1	62.8	68.5
	Liu's Seq2seq [31]	37.4	52.4	57.0	61.7	–	–	–	–
	MEGAN [36]	48.1	70.7	78.4	86.1	33.6	–	–	63.9
	Aug. Transformer [46]	48.3	–	73.4	77.4	–	–	–	–
	GTA [40]	51.1	67.6	74.8	81.6	46.6	–	–	70.4
	Graph2SMILES [47]	52.9	66.5	70.0	72.9	45.7	–	–	63.4
	Retroformer [52]	53.2	71.1	76.6	82.1	–	–	–	–
	Graph2Edits [70]	**55.1**	77.3	83.4	89.4	44.0	60.9	66.8	72.5
	NAG2G [66]	**55.1**	76.9	83.4	89.9	49.7	64.6	69.3	74.0
	UAlign [68]	53.5	77.7	84.6	90.6	**50.6**	**67.2**	**72.3**	**76.2**
	RSmiles [71]	54.1	76.4	82.0	85.4	48.1	64.7	70.0	72.8
	EditRetro [13]	50.7	64.0	68.4	73.0	47.0	56.2	58.7	61.9
	Ours	53.6	**77.8**	**85.1**	**91.2**	50.3	66.6	71.4	76.1
TF*	Aug. Transformer* [46]	53.4	–	80.0	84.2	46.2	–	–	73.3
	NAG2G* [66]	56.2	78.3	85.5	91.7	–	–	–	–
	UAlign* [68]	52.4	77.2	84.1	91.1	47.8	65.9	71.4	76.1
	RSmiles* [71]	56.3	79.7	85.8	91.1	48.9	66.5	71.9	76.8
	EditRetro* [13]	**60.8**	**80.6**	86.1	90.1	**53.3**	65.9	69.2	71.9
	Ours*	55.8	79.7	**86.6**	**92.6**	51.5	**68.3**	**73.2**	**77.8**

Canonical inference. Under canonical inference, MatchRetro exhibits strong and stable performance across all three datasets, indicating that it does not rely on test-time enumeration to produce high-quality candidates. On USPTO-50K, Graph2Edits and NAG2G share the highest top-1 accuracy among the non-augmented template-free methods at 55.1%, while MatchRetro reaches 53.6%, placing

it within the leading group. For deeper candidate lists, MatchRetro remains highly competitive in the top-3 and top-5 and achieves the best top-10 accuracy, suggesting reliable recall when multiple predictions are considered. This trend is more pronounced on USPTO-MIT, where MatchRetro slightly trails UAlign at top-1 but leads across deeper k metrics. On the larger USPTO-FULL benchmark, UAlign maintains a moderate advantage across list depths, while MatchRetro consistently ranks second. Overall, MatchRetro delivers first-tier top-k accuracy under canonical inference across all three datasets, demonstrating that its strong baseline performance does not depend on TTA.

Table 2: Top-k accuracy (%) on USPTO-MIT. *: TTA. **Bold**: best; underlined: second best, per group (TB, TF w/o TTA, TF w/ TTA). TB: template-based; TF: template-free.

Type	Model	Top-k accuracy (%)			
		1	3	5	10
TB	NeuralSym [39]	47.8	67.6	74.1	80.2
	LocalRetro [2]	**54.1**	**73.7**	**79.4**	**84.4**
TF	Liu's Seq2seq [31]	46.9	61.6	66.3	70.8
	AutoSynRoute [30]	54.1	71.8	76.9	81.8
	RetroTRAE [48]	58.3	–	–	–
	UAlign [68]	**60.2**	76.9	81.6	85.4
	Ours	60.0	**77.4**	**82.2**	**86.3**
TF*	UAlign* [68]	57.0	75.8	80.6	84.7
	Ours*	**61.1**	**78.9**	**83.5**	**87.6**

Impact of TTA. TTA exposes markedly different behaviors across architectures. On USPTO-50K, EditRetro obtains an approximately ten-point gain in top-1 accuracy, indicating a strong reliance on stochastic enumeration. RSmiles benefits more modestly, whereas UAlign suffers a noticeable performance drop, suggesting limited compatibility with augmented input variations. In contrast, MatchRetro occupies a favorable middle ground: it retains competitive canonical accuracy while still gaining meaningful improvements from TTA. Specifically, on USPTO-50K, MatchRetro achieves best-in-class augmented top-5 and top-10 accuracies of 86.6% and 92.6%, respectively. On USPTO-MIT, MatchRetro achieves the best results across all reported k thresholds. On USPTO-FULL, although EditRetro leads in top-1 accuracy, MatchRetro attains the highest top-3, top-5, and top-10 accuracies of 68.3%, 73.2%, and 77.8%, respectively. Overall, TTA improves MatchRetro's accuracy at every reported top-k threshold, with its strongest comparative results at deeper candidate-list depths.

Robustness and practical advantages. The combined canonical and augmented evaluation highlights an accuracy-robustness trade-off among existing methods: some models achieve substantial gains from TTA but rely heavily on it, while others remain stable under canonical inference but benefit less from augmented compute. MatchRetro achieves a favorable balance between these regimes. It remains competitive without TTA, improves consistently when TTA is applied, and achieves strong deeper-k performance across most dataset and protocol combinations. This behavior is relevant for synthesis planning, where downstream search algorithms typically expand multiple candidate disconnections rather than relying solely on a single top-1 prediction. The recall of MatchRetro at top-3, top-5, and top-10 therefore indicates its potential to provide reliable candidate pools for downstream planning, while its competitive canonical performance supports practical use when TTA is computationally constrained.

4.2 Chemical Feasibility: Round-Trip Accuracy and Group-Wise Analysis

Table 3: Top-k round-trip accuracy on USPTO-50K. *: TTA. Bold: best; underlined: second best.

Model	Top-k round-trip acc. (%)			
	1	3	5	10
GraphRetro [43]	80.5	73.3	68.3	59.3
Retroformer [52]	78.9	72.0	67.1	57.2
UAlign [68]	81.0	73.9	68.9	59.1
RSmiles*[71]	81.5	73.6	67.7	58.2
EditRetro*[13]	81.3	67.4	59.0	46.0
Ours*	**81.8**	**75.4**	**70.8**	**63.2**

Exact-match accuracy measures agreement with the recorded ground-truth reactants but may penalize chemically valid alternative reactant sets. We therefore complement exact-match evaluation with round-trip accuracy, where a predicted reactant set is considered valid if a pre-trained forward reaction model maps it back to the target product. Following prior protocols [68], we use the pre-trained Molecular Transformer [38] as the forward oracle. For a rigorous comparison, we benchmark against competitive recent methods under their empirically optimal inference settings. As shown in Tab. 3, MatchRetro achieves the best round-trip performance across

all top-k thresholds on USPTO-50K. It consistently outperforms fully generative template-free baselines and also surpasses decomposition-based methods such as GraphRetro, whose reaction-center prediction and leaving-group selection impose a more constrained generation space that can naturally favor chemical validity.

Aggregate round-trip scores summarize overall chemical plausibility but may obscure group-dependent behavior. To further analyze fine-grained robustness, we partition the test set along two structurally challenging axes: stereochemistry and ring topology. The corresponding group-wise exact-match and round-trip distributions are shown in Fig. 2, with extended tabular results provided in Appendix F.7.

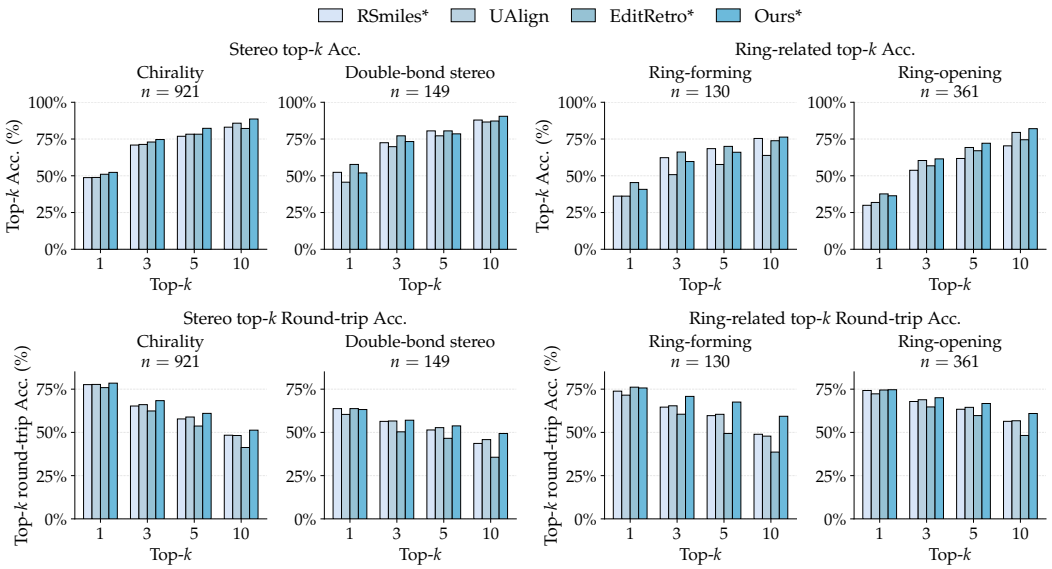


Figure 2: The group-wise top-k accuracy and top-k round-trip accuracy on USPTO-50K. *: TTA.

Stereochemical Robustness. For reactions whose products contain atomic chirality, the five-seed mean for MatchRetro is consistently higher in both exact-match and round-trip accuracy across all evaluation depths. This suggests that explicit structural matching may help the decoder condition on the local product-side environment around stereogenic atoms and maintain more accurate atom-level correspondence during generation. Double-bond stereochemistry, however, presents a different representational challenge. EditRetro achieves higher exact-match accuracy at the early retrieval depths, while MatchRetro's mean is highest at Top-10. This is consistent with a limitation of graph-based encoders: unless explicitly augmented with directional stereochemical features, message-passing representations may struggle to distinguish `E/Z` isomerism as directly as SMILES-based representations. A similar phenomenon is observed for UAlign and is analyzed in Appendix G. For round-trip accuracy, MatchRetro's mean is slightly below the strongest baselines at Top-1, but is highest from Top-3 onward. Thus, even when the recorded isomer is not recovered, its ranked candidates often include alternatives that reproduce the target under the forward oracle.

Ring Topology Transformations. Ring transformations further illustrate the complementary roles of exact-match and round-trip evaluation. For ring-forming reactions, MatchRetro's five-seed mean trails the strongest baselines in exact-match accuracy through Top-5, suggesting that it does not always recover the specific recorded disconnection, but is highest at Top-10. In round-trip evaluation, its mean is slightly below the best baseline at Top-1, then leads from Top-3 onward. This indicates that many alternatives beyond the recorded disconnection can nevertheless reproduce the target product under the forward oracle. For ring-opening reactions, MatchRetro leads in both metrics from Top-3 onward; at Top-1, its mean round-trip accuracy is highest while exact-match accuracy is below the strongest baseline. This pattern indicates strength in both recorded-disconnection recovery and oracle-validated plausibility at deeper retrieval depths.

Overall, the round-trip and group-wise evaluations show that MatchRetro can generate chemically plausible candidate sets beyond exact-match recovery. Exact-match accuracy measures agreement with the recorded reactant set, whereas round-trip validation tests whether predicted precursors can reproduce the target product under a forward oracle. The five-seed means retain strong performance on atom-chirality reactions and show round-trip advantages at most retrieval depths across the stereochemical and ring-topology subsets. Together, these complementary results support explicit structural matching as an effective inductive bias for improving atom-level grounding and the chemical plausibility of generated retrosynthetic candidates, while the size and consistency of the advantage vary by group and retrieval depth.

5 Conclusion

In this work, we introduced MatchRetro, an architecture that explicitly incorporates structural alignment into template-free retrosynthesis through a hierarchical autoregressive framework and a specialized parallel training paradigm. Extensive evaluations demonstrate that this explicit alignment achieves a robust balance between predictive accuracy and generative reliability. MatchRetro maintains highly competitive baseline performance under canonical inference and scales effectively under TTA to deliver superior candidate recall at deeper retrieval depths. By combining efficient inference throughput with the consistent generation of chemically plausible precursors, MatchRetro serves as a dependable foundational engine for automated multi-step synthesis planning.

Limitations. While our supervised explicit structural alignment provides stronger grounding than popular unsupervised strategies, it reduces tolerance for erroneous atom-mapping annotations and demands highly accurate training data. Furthermore, the matching-while-decoding paradigm requires generating a structural position alongside each semantic token. This directly doubles both the effective sequence length during training and the output dimensions per decoding step during inference, presenting a necessary computational trade-off for achieving explicit structural awareness.

Acknowledgments and Disclosure of Funding

This work was supported by the Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (Grant No. JYB2025XDXM411) and the Shanghai Municipal Universities Industry-Academia-Research Joint Innovation Program. This work was also supported by the National Natural Science Foundation of China (Grant No. 92570205), the Chuangqi Program of the Shanghai Innovation Institute, and the Shanghai Municipal Education Commission. We acknowledge the support from the Open Research Fund of Suzhou Laboratory (No. SZLAB-1508-2024-TS017) and the ECNU Multifunctional Platform for Innovation (No. 001) for providing supercomputer time.

References

[1] N. S. Bodor. Soft anticholinergic esters, Jan. 14 2014. US Patent 8,628,759.

[2] S. Chen and Y. Jung. Deep retrosynthetic reaction prediction using local reactivity and global attention. *JACS Au*, 1(10):1612–1620, 2021.

[3] Z. Chen, O. R. Ayinde, J. R. Fuchs, H. Sun, and X. Ning. G 2 retro as a two-step graph generative models for retrosynthesis prediction. *Communications Chemistry*, 6, 12 2023.

[4] C. W. Coley, W. H. Green, and K. F. Jensen. Rdchiral: An rdkit wrapper for handling stereochemistry in retrosynthetic template extraction and application. *Journal of chemical information and modeling*, 59(6):2529–2537, 2019.

[5] C. W. Coley, L. Rogers, W. H. Green, and K. F. Jensen. Computer-assisted retrosynthesis based on molecular similarity. *ACS central science*, 3(12):1237–1245, 2017.

[6] E. J. Corey. The logic of chemical synthesis: multistep synthesis of complex carbogenic molecules (nobel lecture). *Angewandte Chemie International Edition in English*, 30(5):455–465, 1991.

[7] S. Current, Z. Chen, D. Adu-Ampratwum, X. Ning, and S. Parthasarathy. Differ: categorical diffusion ensembles for single-step chemical retrosynthesis. *Journal of Cheminformatics*, 17(1):112, 2025.

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

[9] Y. Deng, X. Zhao, H. Sun, Y. Chen, X. Wang, X. Xue, L. Li, J. Song, C.-Y. Hsieh, T. Hou, X. Pan, T. S. Alomar, X. Ji, and X. Wang. RSGPT: a generative transformer model for retrosynthesis planning pre-trained on ten billion datapoints. *Nature Communications*, 16(1):7012, 2025.

[10] M. Fey and J. E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[11] S. Fujita. Description of organic reactions based on imaginary transition structures. 1. introduction of new concepts. *Journal of Chemical Information and Computer Sciences*, 26(4):205–212, 1986.

[12] S. Fujita. Description of organic reactions based on imaginary transition structures. 6. classification and enumeration of two-string reactions with one common node. *Journal of Chemical Information and Computer Sciences*, 27(3):99–104, 1987.

[13] Y. Han, X. Xu, C. Y. Hsieh, K. Ding, H. Xu, R. Xu, T. Hou, Q. Zhang, and H. Chen. Retrosynthesis prediction with an iterative string editing model. *Nature Communications*, 15(1):1–16, 2024.

[14] E. Heid, K. P. Greenman, Y. Chung, S.-C. Li, D. E. Graff, F. H. Vermeire, H. Wu, W. H. Green, and C. J. McGill. Chemprop: A machine learning package for chemical property prediction. *Journal of Chemical Information and Modeling*, 64(1):9–17, 2024.

[15] W. Hu, M. Fey, M. Zitnik, Y. Dong, H. Ren, B. Liu, M. Catasta, and J. Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems*, 33:22118–22133, 2020.

[16] I. Igashov, A. Schneuing, M. Segler, M. M. Bronstein, and B. Correia. Retrobridge: Modeling retrosynthesis with markov bridges. In *The Twelfth International Conference on Learning Representations*, 2024.

[17] R. Irwin, S. Dimitriadis, J. He, and E. J. Bjerrum. Chemformer: a pre-trained transformer for computational chemistry. *Machine Learning: Science and Technology*, 3(1):015022, 2022.

[18] S. Ishida, T. Miyazaki, Y. Sugaya, and S. Omachi. Graph neural networks with multiple feature extraction paths for chemical property estimation. *Molecules*, 26(11):3125, 2021.

[19] Y. Ji, L. Zhang, J. Wu, B. Wu, L. Li, L.-K. Huang, T. Xu, Y. Rong, J. Ren, D. Xue, H. Lai, W. Liu, J. Huang, S. Zhou, P. Luo, P. Zhao, and Y. Bian. DrugOOD: Out-of-distribution dataset curator and benchmark for AI-aided drug discovery – a focus on affinity prediction problems with noise annotations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8023–8031. AAAI Press, 2023.

[20] Y. Jian, Y. Zhang, Y. Wei, H. Fan, and Y. Yang. Reaction graph: Towards reaction-level modeling for chemical reactions with 3d structures. In *Forty-second International Conference on Machine Learning*, 2025.

[21] Y. Jiang, W. Ying, F. Wu, Z. Huang, K. Kuang, and Z. Wang. Learning chemical rules of retrosynthesis with pre-training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 5113–5121, 2023.

[22] W. Jin, C. Coley, R. Barzilay, and T. Jaakkola. Predicting organic reaction outcomes with weisfeiler-lehman network. *Advances in neural information processing systems*, 30, 2017.

[23] Y.-T. Kao, S.-F. Wang, M.-H. Wu, S.-H. Her, Y.-H. Yang, C.-H. Lee, H.-F. Lee, A.-R. Lee, L.-C. Chang, and L.-H. Pao. A substructure-based screening approach to uncover n-nitrosamines in drug substances. *Journal of Food & Drug Analysis*, 30(1), 2022.

[24] E. Kim, D. Lee, Y. Kwon, M. S. Park, and Y.-S. Choi. Valid, plausible, and diverse retrosynthesis using tied two-way transformers with latent variables. *Journal of Chemical Information and Modeling*, 61(1):123–133, 2021.

[25] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[26] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. M. Rush. Opennmt: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, system demonstrations*, pages 67–72, 2017.

[27] N. Kumar, K. Kaur, S. Aeron, S. Dharmarajan, A. D. Silamkoti, A. Mehta, S. Gupta, A. Chugh, J. B. Gupta, M. Salman, V. P. Palle, and I. A. Cliffe. Synthesis and optimization of novel and selective muscarinic m3 receptor antagonists. *Bioorganic & Medicinal Chemistry Letters*, 17(18):5256–5260, 2007.

[28] G. Landrum. Rdkit: A software suite for cheminformatics, computational chemistry, and predictive modeling. *Greg Landrum*, 8:31, 2013.

[29] C. M. Lanthier, B. Gorin, J. Oudenes, C. E. Dixon, A. Q. Lu, J. D. Copp, and J. M. Janusz. Process for preparing [(3-hydroxypyridine-2-carbonyl) amino] alkanoic acids, esters and amides, Sept. 29 2015. US Patent 9,145,366.

[30] K. Lin, Y. Xu, J. Pei, and L. Lai. Automatic retrosynthetic route planning using template-free models. *Chemical science*, 11(12):3355–3364, 2020.

[31] B. Liu, B. Ramsundar, P. Kawthekar, J. Shi, J. Gomes, Q. Luu Nguyen, S. Ho, J. Sloane, P. Wender, and V. Pande. Retrosynthetic reaction prediction using neural sequence-to-sequence models. *ACS central science*, 3(10):1103–1113, 2017.

[32] H. L. Morgan. The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service. *Journal of chemical documentation*, 5(2):107–113, 1965.

[33] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[34] S. Pathak, E. Pandit, and A. Patil. Sofpironium bromide: A comprehensive review of its efficacy and safety in patients with primary axillary hyperhidrosis. *National Journal of Pharmacology and Therapeutics*, 2(3):129–133, 2024.

[35] S. Rashmeen, I. Sohail, M. Mohsin, D. Anwar, S. Farhat, S. Shahnoor, A. Maryyum, and A. M. Khan. Sofpironium bromide: a novel solution in the battle against primary axillary hyperhidrosis, 2025.

[36] M. Sacha, M. Błaz, P. Byrski, P. Dabrowski-Tumanski, M. Chrominski, R. Loska, P. Włodarczyk-Pruszynski, and S. 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.

[37] R. Scheffler. The partial search order problem. *The Electronic Journal of Combinatorics*, pages P4–5, 2025.

[38] P. Schwaller, T. Laino, T. Gaudin, P. Bolgar, C. A. Hunter, C. Bekas, and A. A. Lee. Molecular Transformer: A model for uncertainty-calibrated chemical reaction prediction. *ACS Central Science*, 5(9):1572–1583, 2019.

[39] M. H. Segler and M. P. Waller. Neural-symbolic machine learning for retrosynthesis and reaction prediction. *Chemistry–A European Journal*, 23(25):5966–5971, 2017.

[40] S.-W. Seo, Y. Y. Song, J. Y. Yang, S. Bae, H. Lee, J. Shin, S. J. Hwang, and E. Yang. Gta: Graph truncated attention for retrosynthesis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(1):531–539, May 2021.

[41] F. Shaikh and S. Sharma. Unveiling vadadustat: comprehensive review of its chemistry, pharmacology, bioanalysis, and patent landscape as a novel hif-ph inhibitor. *Current topics in medicinal chemistry*, 2025.

[42] C. Shi, M. Xu, H. Guo, M. Zhang, and J. Tang. A graph to graphs framework for retrosynthesis prediction. In *International conference on machine learning*, pages 8818–8827. PMLR, 2020.

[43] V. R. Somnath, C. Bunne, C. Coley, A. Krause, and R. Barzilay. Learning graph models for retrosynthesis prediction. *Advances in Neural Information Processing Systems*, 34:9405–9415, 2021.

[44] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu. RoFormer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[45] Y. Sun, H. Wu, K. Zeng, and X. Wang. Leveraging llms for lncrna classification and lncrna-protein interaction prediction: A novel approach for sequence-based feature extraction. In *2025 International Conference on Machine Intelligence and Nature-Inspired Computing (MIND)*, pages 465–470. IEEE, 2025.

[46] I. V. Tetko, P. Karpov, R. Van Deursen, and G. Godin. State-of-the-art augmented nlp transformer models for direct and single-step retrosynthesis. *Nature communications*, 11(1):5575, 2020.

[47] Z. Tu and C. 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.

[48] U. V. Ucak, I. Ashyrmamatov, J. Ko, and J. Lee. Retrosynthetic reaction pathway prediction through neural machine translation of atomic environments. *Nature communications*, 13(1):1186, 2022.

[49] P. van Gerwen, K. R. Briling, C. Bunne, V. R. Somnath, R. Laplaza, A. Krause, and C. Corminboeuf. 3dreact: Geometric deep learning for chemical reactions. *Journal of chemical information and modeling*, 64(15):5771–5785, 2024.

[50] A. Varnek, D. Fourches, F. Hoonakker, and V. P. Solov'ev. Substructural fragments: an universal language to encode reactions, molecular and supramolecular structures. *Journal of computer-aided molecular design*, 19(9):693–703, 2005.

[51] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[52] Y. Wan, C.-Y. Hsieh, B. Liao, and S. Zhang. Retroformer: Pushing the limits of end-to-end retrosynthesis transformer. In *International Conference on Machine Learning*, pages 22475–22490. PMLR, 2022.

[53] C. Wang, Z. Zhou, L. Bai, and T. Yu. Order matters in retrosynthesis: Structure-aware generation via reaction-center-guided discrete flow matching. In *Forty-third International Conference on Machine Learning*, 2026.

[54] X. Wang, Y. Li, J. Qiu, G. Chen, H. Liu, B. Liao, C. Y. Hsieh, and X. Yao. Retroprime: A diverse, plausible and transformer-based method for single-step retrosynthesis predictions. *Chemical Engineering Journal*, 420, 9 2021.

[55] Y. Wang, C. Pang, Y. Wang, J. Jin, J. Zhang, X. Zeng, R. Su, Q. Zou, and L. Wei. Retrosynthesis prediction with an interpretable deep-learning framework based on molecular assembly tasks. *Nature Communications*, 14(1):6155, 2023.

[56] Y. Wang, F. Yang, B. Wang, L. Xie, and W. Chen. New fda drug approvals for 2024: Synthesis and clinical application. *European Journal of Medicinal Chemistry*, 285:117241, 2025.

[57] D. Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.

[58] H. Wu, W. Liu, Y. Bian, J. Wu, N. Yang, and J. Yan. Ebmdock: Neural probabilistic protein-protein docking via a differentiable energy model. In *International Conference on Learning Representations*, volume 2024, pages 20317–20336, 2024.

[59] H. Wu, X. Ye, and J. Yan. Qvae-mole: The quantum vae with spherical latent variable learning for 3-d molecule generation. *Advances in Neural Information Processing Systems*, 37:22745–22771, 2024.

[60] S. Xie, R. Yan, J. Guo, Y. Xia, L. Wu, and T. Qin. Retrosynthesis prediction with local template retrieval. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(4):5330–5338, Jun. 2023.

[61] J. Xiong, W. Zhang, Y. Wang, J. Huang, Y. Shi, M. Xu, M. Li, Z. Fu, X. Kong, Y. Wang, Z. Xiong, and M. Zheng. Bridging chemistry and artificial intelligence by a reaction description language. *Nature Machine Intelligence*, 7(5):782–793, May 2025.

[62] R. Yadav, Q. Yan, G. Wolf, J. Bose, and R. Liao. Retro synflow: Discrete flow-matching for accurate and diverse single-step retrosynthesis. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen, editors, *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pages 55229–55258. Curran Associates, Inc., 2025.

[63] C. Yan, Q. Ding, P. Zhao, S. Zheng, J. Yang, Y. Yu, and J. Huang. Retroxpert: Decompose retrosynthesis prediction like a chemist. *Advances in Neural Information Processing Systems*, 33:11248–11258, 2020.

[64] G. Yan, H. Wu, and J. Yan. Quantum 3d graph learning with applications to molecule embedding. In *International Conference on Machine Learning*, pages 39126–39137. PMLR, 2023.

[65] N. Yang, H. Wu, K. Zeng, Y. Li, S. Bao, and J. Yan. Molecule generation for drug design: a graph learning perspective. *Fundamental Research*, 6(1):40–52, 2026.

[66] L. Yao, W. Guo, Z. Wang, S. Xiang, W. Liu, and G. Ke. Node-aligned graph-to-graph: Elevating template-free deep learning approaches in single-step retrosynthesis. *JACS Au*, 4(3):992–1003, 2024.

[67] K. Zeng, X. Liu, Y. Zhang, X. Yang, Y. Jin, and Y. Xu. A general-purpose framework for chemical reaction representation with atomic correspondence and flexible condition adaptation. *Journal of Cheminformatics*, May 2026.

[68] K. Zeng, B. Yang, X. Zhao, Y. Zhang, F. Nie, X. Yang, Y. Jin, and Y. Xu. Ualign: pushing the limit of template-free retrosynthesis prediction with unsupervised smiles alignment. *Journal of Cheminformatics*, 16(1):80, 2024.

[69] S. Zheng, J. Rao, Z. Zhang, J. Xu, and Y. Yang. Predicting retrosynthetic reactions using self-corrected transformer neural networks. *Journal of chemical information and modeling*, 60(1):47–55, 2019.

[70] W. Zhong, Z. Yang, and C. Y.-C. Chen. Retrosynthesis prediction using an end-to-end graph generative architecture for molecular graph editing. *Nature Communications*, 14(1):3009, 2023.

[71] Z. Zhong, J. Song, Z. Feng, T. Liu, L. Jia, S. Yao, M. Wu, T. Hou, and M. Song. Root-aligned smiles: a tight representation for chemical reaction prediction. *Chemical Science*, 13(31):9023–9034, 2022.

A Related Works

A.1 Molecular Representations and Structural Alignment for Reaction Modeling

Molecular representation learning is commonly organized around string-based and structure-based paradigms [65]. The Simplified Molecular Input Line Entry System (SMILES) [57] represents molecular graphs as linear textual sequences, enabling the direct application of modern language modeling techniques to chemical data [17, 45]. This string-based formulation naturally supports large-scale pre-training and sequence-to-sequence modeling and has been widely adopted in molecular property prediction, reaction prediction, and retrosynthesis. However, SMILES encodes molecular topology only implicitly through a traversal-dependent sequence, so structurally similar molecules may still appear distant at the sequence level. This limitation has motivated the development of structure-aware representations that operate more directly on molecular connectivity. Early structure-based approaches relied on handcrafted molecular fingerprints, such as Morgan fingerprints [32], which characterize molecules through predefined local fragments. Although fingerprints provide compact and chemically meaningful descriptors, they are typically fixed, non-differentiable representations that are not optimized end-to-end for downstream objectives. More recently, graph neural networks have become a dominant framework for molecular representation learning and molecule generation [19, 58, 59]. By treating atoms as nodes and bonds as edges, these models learn topology-aware molecular representations through iterative message passing while respecting the permutation invariance of molecular graphs [22, 23, 18, 64].

Reaction modeling further requires representing the correspondence and structural changes between reactants and products rather than modeling each molecule independently. Classical representations such as Imaginary Transition Structures [11] and Condensed Graphs of Reaction [50, 12] encode reactions by unifying reactant and product graphs and annotating bond changes. Similar ideas have been adopted in neural reaction models, including reaction-aware implementations in Chemprop [14], which supports reaction inputs in its message-passing framework. Recent works further incorporate reaction-level structure in different ways: Reaction Graph [20] and 3DReact [49] extend reaction modeling to three-dimensional geometries, while AlignReact [67] integrates atom-level reactant-product correspondence into reaction representation learning. These studies collectively suggest that reaction modeling benefits from representations that make structural correspondence explicit. Our work follows this principle in the context of template-free retrosynthesis but differs from prior reaction representation learning methods by incorporating explicit structural alignment into the generative decoding process rather than only into the encoder representation.

A.2 Single-Step Retrosynthesis Prediction

According to their dependence on reaction templates, existing single-step retrosynthesis prediction methods can be broadly categorized into template-based and template-free methods. Template-based methods rely on a predefined library of reaction templates, while template-free methods aim to predict reactants without explicitly retrieving a complete template from such a library.

Template-based methods formulate single-step retrosynthesis as a template retrieval or classification problem. Given a target product molecule, these methods select applicable reaction templates from a predefined template library and apply them to generate candidate reactants. Early approaches such as RetroSim [5] rank candidate templates by measuring molecular similarity to precedent reactions. NeuralSym [39] replaces hand-crafted similarity scoring with a neural classifier that predicts suitable symbolic transformation rules from molecular fingerprints. GLN [8] models retrosynthesis with a conditional graph logic network that combines reaction-template constraints with graph neural scoring. LocalRetro [2] improves template-based prediction by deriving local reaction templates and using graph neural networks to predict template applicability at atom- and bond-level reaction centers. RetroKNN [60] further enhances local template retrieval with non-parametric K-nearest-neighbor search, alleviating the long-tailed distribution and data imbalance of reaction templates. By explicitly encoding chemically meaningful transformation rules, template-based methods offer strong interpretability and can generate chemically plausible reactants. However, their predictions are inherently constrained by the coverage of the extracted template library, making it difficult to generalize to reactions outside the predefined template space and limiting their scalability.

Template-free methods encompass a diverse range of modeling strategies that can be broadly divided into two-stage synthon-based approaches and end-to-end generative architectures. Two-

stage methods follow chemist-inspired retrosynthetic reasoning by decomposing the prediction task into synthon identification and subsequent reactant completion. Synthon generation is typically achieved through reaction center prediction and molecular editing operations. After synthons are extracted, final reactants are constructed through different completion strategies, including leaving-group or substructure attachment [43, 3], variational graph translation [42], autoregressive SMILES generation [63], or hierarchical molecular assembly [55]. RetroPrime [54] follows a similar two-stage decomposition but implements both product-to-synthon and synthon-to-reactant transformations using Transformer-based sequence translation. While these formulations align closely with chemical intuition and provide interpretable intermediate states, their stage-wise nature may introduce error accumulation. In particular, methods relying on explicit reaction center prediction can become less reliable for complex transformations involving multiple reaction sites or substantial structural rearrangements.

A parallel line of template-free research abandons explicit subtask decomposition and directly generates reactants from the target product. A prominent family of end-to-end methods formulates retrosynthesis as sequence generation, ranging from SMILES-to-SMILES translation [31, 46, 24] to graph-to-SMILES translation [47]. To better exploit molecular structure within sequence generation, GTA [40] and Retroformer [52] introduce graph-aware or local attention mechanisms that guide reactant-product reasoning. RSmiles [71] and UAlign [68] instead improve sequence-level alignment by reducing the discrepancy between product and reactant representations, encouraging the model to reuse preserved molecular fragments. Beyond standard sequence translation, MEGAN [36] and Graph2Edits [70] model retrosynthesis through autoregressive molecular graph edits that transform the product into intermediate structures and final reactants, while EditRetro [13] uses iterative string editing. Their action spaces make local transformations explicit, but these edit sequences do not impose the same separate synthon-generation and reactant-completion stages as the decomposed methods above. Other end-to-end frameworks use categorical diffusion over output SMILES [7], Markov-bridge-based generation over molecular graphs [16], or discrete flow matching [62] with reaction-center-guided ordering [53]. These approaches provide flexible alternatives to standard autoregressive sequence generation and highlight the tension between chemically structured intermediate reasoning and largely implicit structural alignment, motivating our explicit alignment formulation for template-free retrosynthesis.

Pretraining has also been used to address the data bottleneck in retrosynthesis. Jiang et al. [21] pretrain on reaction data to learn chemical rules, RSGPT [9] pretrains on roughly ten billion synthetic reaction records, and ReactSeq [61] combines a reaction-description language with molecular pretraining on a large external corpus. These studies show how data scale and chemically informed representations can extend retrosynthesis models beyond the information available in a single benchmark split.

B Motivation Justification: On the Hardness of SMILES Traversal Alignment

We first formalize why minimizing product-reactant discrepancy through SMILES traversal selection is intrinsically difficult. Consider the following optimization problem: given a reactant graph $G_R = (V_R, E_R)$ and a fixed product DFS sequence σ_P, find a DFS preorder $\sigma_R \in \mathrm{DFS}(G_R)$ that minimizes the discrepancy:

$$\mathrm{ED}(\sigma_P, \sigma_R).$$

Here, the atoms in the product are uniquely matched to the atoms in the reactant, and each reactant atom can be matched at most once. After relabeling each product atom by its corresponding reactant atom, the product vertex set can be viewed as a subset of the reactant vertex set, namely $V_P \subseteq V_R$. The remaining vertices

$$L = V_R \setminus V_P$$

correspond to reactant atoms that lack counterparts on the product-side.

To establish the hardness of this optimization problem, we consider its decision version. Given G_R, σ_P, and an integer threshold K, the decision problem asks whether there exists a DFS preorder σ_R of G_R such that

$$\mathrm{ED}(\sigma_P, \sigma_R) \leq K.$$

This problem is in NP since a candidate order σ_R can be verified in polynomial time, and the edit distance between two sequences can be computed using dynamic programming.

We prove NP-hardness by reducing from an ordered-terminal form of the DFS Partial Search Order Problem (DFS-PSOP) [37]. Given a graph $G = (V, E)$, a terminal set $T = \{t_1, \ldots, t_q\}$, and a prescribed terminal order $\tau = (t_1, \ldots, t_q)$, the problem asks whether there exists a DFS preorder π of G whose restriction to the terminals is exactly τ:

$$\mathrm{proj}_T(\pi) = \tau.$$

Equivalently, this asks whether the terminal vertices can appear in the DFS preorder according to the chain constraint

$$t_1 \prec t_2 \prec \cdots \prec t_q.$$

Given an instance (G, T, τ), we construct an instance of the reactant DFS edit-distance decision problem by setting

$$G_R = G, \qquad V_P = T, \qquad \sigma_P = \tau.$$

The vertices in

$$L = V_R \setminus V_P$$

are interpreted as unmatched reactant vertices. We set the threshold to

$$K = |L| = |V_R| - |V_P|.$$

This threshold is not arbitrary. For any DFS preorder σ_R of G_R, the sequence σ_R contains all vertices in V_R, whereas σ_P contains only the vertices in V_P. Their lengths, therefore, differ by exactly $|L|$. Since the edit distance is at least the absolute difference in sequence lengths, we have

$$\mathrm{ED}(\sigma_P, \sigma_R) \geq |L|.$$

Thus, $K = |L|$ is the smallest edit distance that any feasible reactant DFS order can achieve.

It remains to characterize when this lower bound is attainable. Equality holds if and only if σ_P can be obtained from σ_R by deleting exactly the unmatched reactant vertices in L, without requiring substitutions or additional edits. Since the matched atoms are uniquely labeled, this condition is equivalent to requiring that the projection of σ_R onto V_P equals the product sequence:

$$\mathrm{ED}(\sigma_P, \sigma_R) = |L| \quad \Longleftrightarrow \quad \mathrm{proj}_{V_P}(\sigma_R) = \sigma_P.$$

Therefore, for the constructed instance,

$$\exists \sigma_R \in \mathrm{DFS}(G_R) : \mathrm{ED}(\sigma_P, \sigma_R) \leq |L|$$

if and only if

$$\exists \pi \in \mathrm{DFS}(G) : \mathrm{proj}_T(\pi) = \tau.$$

The construction is polynomial; hence, the reactant DFS edit-distance decision problem is NP-hard. Since it is also in NP, it is NP-complete. Consequently, the original optimization problem

$$\min_{\sigma_R \in \mathrm{DFS}(G_R)} \mathrm{ED}(\sigma_P, \sigma_R)$$

is NP-hard. Indeed, a polynomial-time algorithm for computing the optimal reactant DFS order would immediately solve the decision problem by checking whether the optimal edit distance is at most K.

This hardness result helps explain why traversal-level alignment remains a limiting factor for SMILES-based retrosynthesis models. Even when the product-to-reactant atom correspondence is known, selecting a reactant DFS traversal that best agrees with a fixed product traversal can require solving a combinatorially hard ordering problem. Existing SMILES-level alignment methods, therefore, rely on practical heuristics rather than exact optimization.

To examine how much discrepancy remains under such heuristics, we analyze the edit distance between aligned product and reactant DFS sequences generated by RSmiles [71]. As shown in Fig. 3, substantial sequence-level gaps persist even after excluding unmatched reactant atoms. In particular, more than 30% of reactions still exhibit an edit distance larger than 8. The discrepancy becomes even more pronounced when unmatched reactant atoms are included, as these leaving-group atoms are not guaranteed to appear at the end of a SMILES traversal. These observations suggest that implicit SMILES-level alignment cannot fully exploit the structural correspondence between products and reactants.

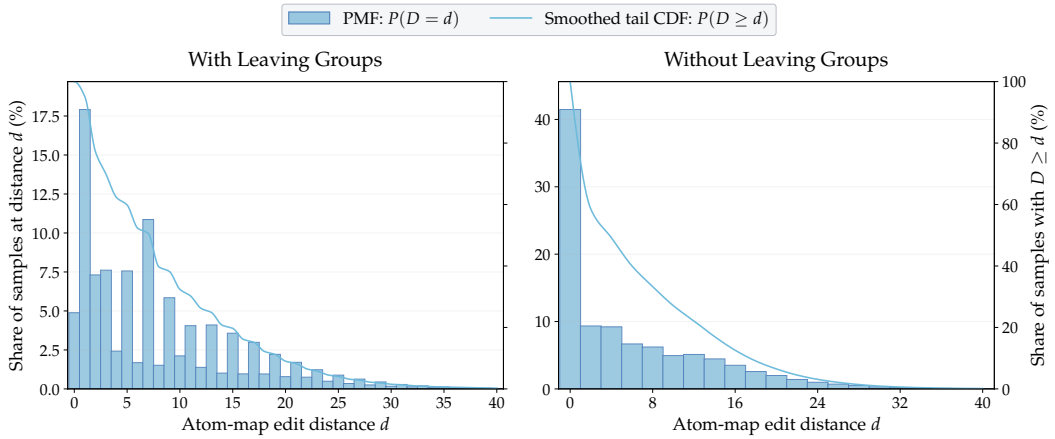


Figure 3: Overall atom-map edit-distance distributions on USPTO-50K (aug20). The left panel measures distance on full mapped reactants, including leaving groups; the right panel removes reactant-only atom maps and thus has support only on even values. Bars show the exact PMF $P(D = d)$, and the curve shows the smoothed tail distribution $P(D \geq d)$. Removing leaving groups shifts the distribution toward smaller distances.

This motivates our explicit structural-alignment formulation. Rather than searching for a single traversal that makes the reactant sequence resemble the product sequence, we directly organize the generation around the preserved product-derived atoms and the newly introduced reaction-specific atoms. In this way, the model is encouraged to reuse structurally aligned fragments while separately modeling the components that are absent from the product. This design pushes product–reactant alignment beyond traversal-level heuristics and provides a stronger inductive bias for template-free retrosynthesis prediction.

C Datasets and Preprocessing

We evaluate three public single-step retrosynthesis benchmarks: USPTO-50K, USPTO-MIT, and USPTO-FULL. Since the raw releases contain atom-mapping and reaction-format inconsistencies, we reprocess each dataset before training and evaluation.

USPTO-50K and USPTO-FULL. We start from the GLN release [8] and apply the same validation pipeline to both datasets. Specifically, we remove reactions with invalid reactant or product SMILES, reactions whose products contain fewer than five heavy atoms, reactions whose canonicalized reactants and products are identical, reactions with duplicate atom map numbers in either reactants or products, and reactions whose product atoms cannot be consistently matched to reactant atoms by atom mapping, including missing correspondences and element or isotope mismatches. We also check for missing atom mappings in the product.

USPTO-MIT. We start from the USPTO-MIT release [22]. Unlike some earlier preprocessing pipelines, we retain multi-component products. We first assign valid atom maps to reactant atoms with missing mappings or to atoms that do not participate in the product, and then we remove reactant molecules whose atom maps have no overlap with the product, treating them as reagents. After this normalization step, we apply the same validation rules as described above.

The statistics of the processed datasets used in this work are summarized in Table 4.

Table 4: Statistics of the processed datasets used in this work.

Dataset	#Train	#Valid	#Test	#Total
USPTO-50K	40,004	5,001	5,007	50,012
USPTO-MIT	408,927	29,994	39,990	478,911
USPTO-FULL	762,875	95,386	95,282	953,543

D Evaluation Metrics

We evaluate retrosynthesis performance using both exact-match top-k accuracy and round-trip top-k accuracy. Let $\mathcal{D} = \{(x_i, y_i^\star)\}_{i=1}^{M}$ denote the test set, where x_i is the target product and $y_i^\star$ is the ground-truth reactant set. For each product x_i, the model produces a ranked list of reactant predictions $\{\hat{y}_{i,1}, \ldots, \hat{y}_{i,K}\}$. All reactant sets are canonicalized before comparison.

Let $\mathrm{CS}(\cdot)$ denote the canonical SMILES string of a reactant set. The top-k accuracy is defined as

$$\mathrm{Acc@}k = \frac{1}{M} \sum_{i=1}^{M} \mathbb{I}\left(\exists j \le k,\ \mathrm{CS}(\hat{y}_{i,j}) = \mathrm{CS}(y_i^\star)\right). \tag{12}$$

Top-k Round-trip Accuracy. Exact-match accuracy may underestimate retrosynthesis performance because a target product can often be synthesized from multiple valid reactant sets, while the dataset provides only one recorded ground-truth solution. Therefore, we additionally report top-k round-trip accuracy as a proxy for reaction validity. Following Retroformer [52], we use a pre-trained Molecular Transformer [38] as the forward reaction prediction oracle.

For each predicted reactant set $\hat{y}_{i,j}$, the forward oracle predicts a product $\mathcal{F}(\hat{y}_{i,j})$. A predicted route is considered round-trip valid if the canonical SMILES string of the forward-predicted product exactly matches that of the target product x_i. Let $\mathrm{CS}(\cdot)$ denote the canonical SMILES string. The top-k round-trip accuracy is defined as

$$\mathrm{RTAcc@}k = \frac{1}{Mk} \sum_{i=1}^{M} \sum_{j=1}^{k} \mathbb{I}\left(\mathrm{CS}\big(\mathcal{F}(\hat{y}_{i,j})\big) = \mathrm{CS}(x_i)\right), \tag{13}$$

where M is the number of test products, k is the number of evaluated predictions for each product, and $\mathbb{I}(\cdot)$ is the indicator function. Unlike exact-match top-k accuracy, this metric averages the validity of all top-k predicted routes rather than checking whether at least one valid route appears among them.

E Implementation Details

E.1 Reranking with TTA

Following prior TTA-based reranking strategies in retrosynthesis [46, 71, 13], we apply TTA during inference on the validation and test sets. For each product molecule, we construct N input SMILES, including its canonical form and $N-1$ randomly generated non-canonical forms. Each input is decoded independently to produce a ranked list of reactant predictions. We then canonicalize the predicted reactant SMILES, collapse duplicate candidates within each augmented branch, and aggregate the resulting ranked lists across branches for final reranking. For USPTO-50K, we set $N = 20$, while for USPTO-MIT and USPTO-FULL, we use $N = 5$.

Let $\tilde{y}_{n,k}$ denote the k-th unique canonical candidate in the n-th augmented branch after within-branch deduplication, where $n \in \{1, \ldots, N\}$ and $k \in \{1, \ldots, K_n\}$. The final reranking score for a candidate c is defined as

$$\mathrm{Score}(c) = \sum_{n=1}^{N} \sum_{k=1}^{K_n} \frac{\mathbb{I}(c = \tilde{y}_{n,k})}{1 + \alpha(k-1)}, \tag{14}$$

where $\mathbb{I}(\cdot)$ is the indicator function, K_n is the number of valid unique candidates retained in branch n, and α controls the rank decay for lower-ranked predictions. In all experiments, we set $\alpha = 0.1$. The final predictions are obtained by sorting all unique canonical candidates according to $\mathrm{Score}(c)$ and selecting the top-K results.

E.2 Inference-Time Structural Constraints

To guaranty chemical validity and strictly enforce the fundamental one-to-one mapping of atoms between reactants and products, we augment the conditional prediction cascade with deterministic masking constraints during the inference phase.

We formalize the tracking of available structural targets by maintaining a dynamic node occupancy state throughout the autoregressive generation process. Let $\mathcal{U}_i \subseteq \mathcal{V}$ denote the set of unoccupied product nodes at generation step i. At the beginning of decoding, all product nodes are available and initialized as $\mathcal{U}_1 = \mathcal{V}$. Whenever a generated token is successfully mapped to a product node v, that specific node is permanently removed from the unoccupied set for all subsequent decoding steps.

For any newly generated atom token t_i, we define a chemically valid candidate set $\mathcal{A}_i$. A product node v is included in $\mathcal{A}_i$ if and only if it satisfies two strict criteria. First, the node must not be occupied, which requires $v \in \mathcal{U}_i$. Second, to enforce the chemical law of elemental conservation, the atomic symbol of node v must exactly match the semantic identity of token t_i.

This deterministic candidate set directly governs the routing and matching probabilities through two specific interventions:

Deterministic Leaving Group Override. Prior to passing the hidden state to the leaving group classifier, we evaluate the size of the candidate set. If no valid product nodes exist ($\mathcal{A}_i = \emptyset$), mapping the atom to the structural memory is physically impossible. In this scenario, we bypass the MLP_{lg} module entirely and deterministically set the leaving group probability to 1, forcing the structural state to $p_i = -2$.

Masked Structural Matching. If valid candidate nodes do exist ($\mathcal{A}_i \neq \emptyset$) and the token is not classified as a leaving group, the structural matching space is strictly confined to the candidate set. We compute the matching logits exclusively for the valid nodes. Consequently, the softmax distribution defined in Section 3.4 is reformulated as a constrained masked softmax:

$$P(p_i = k_v \mid t_{\leq i}, p_{<i}, p_i \neq -2) = \frac{\exp(s_{i,v})}{\sum_{u \in \mathcal{A}_i} \exp(s_{i,u})} \tag{15}$$

for any node $v \in \mathcal{A}_i$. For all invalid or previously occupied nodes where $v \notin \mathcal{A}_i$, the matching probability is strictly zeroed out. This intervention effectively prunes structurally impossible branches without altering the learned parameters of the matching network.

E.3 Model Implementation Details

We implement our model using `PyTorch 2.3.0` [33] and `torch_geometric 2.6.1` [10]. For input-output preprocessing, we adopt the reactant-product alignment algorithm introduced in UAlign [68] and reimplement it with a modified RDKit backend, which is extended for our alignment pipeline. During training, input product SMILES are fully randomized on the fly. During validation and testing, we use the canonical input representation by default and apply TTA only in the reranking setting described above. We train the model with Adam [25] on a cluster equipped with 128 NVIDIA RTX 4090 GPUs. Gradient accumulation is used to achieve the effective batch sizes reported in Table 5. For all datasets, we linearly warm up the learning rate in the first few epochs and then decay it using an exponential scheduler with the exponent η. Both training stages are run for at most 300 epochs, and we apply early stopping if the validation performance does not improve for 10 consecutive epochs. For the pre-training loss in Eq. 10, we set $\gamma = 1$. For the Stage-II loss in Eq. 11, we set $\alpha = \beta = 1$ on USPTO-50K and USPTO-MIT, and $\alpha = \beta = 0.1$ on USPTO-FULL.

Table 5: Training hyperparameters for the reported models. BS denotes the effective batch size after data parallelism and gradient accumulation.

Dataset	Dim	Heads	Enc.	Dec.	Dropout	LR	Epochs	η	BS
USPTO-50K	512	8	6	6	0.1	2×10^{-4}	300	0.992	512
USPTO-MIT	768	12	8	8	0.1	5×10^{-5}	300	0.992	1024
USPTO-FULL	768	12	8	10	0.1	5×10^{-5}	300	0.992	2048

E.4 Initial Node Features and Edge Features

We use the atom and bond descriptors provided by the Open Graph Benchmark [15]. Specifically, each atom is represented by nine discrete descriptors, including atomic number, chirality, formal charge, and other properties computed from RDKit [28]. Each bond is represented by three discrete

descriptors: bond type, bond stereochemistry, and whether the bond is conjugated. Since all descriptors are integer-valued, we use learnable embedding tables for each descriptor and obtain the initial node and edge representations by summing the corresponding embeddings.

In addition to the standard bond descriptors, we introduce a directed edge feature named `local_pe`. For a message passed from node i to node j, if j is a chirality center, `local_pe` encodes the relative position of i among the neighbors of j, with the neighbor order determined by the input SMILES. This feature provides chirality-sensitive local ordering information to the message passing layers. For edges whose target node is not a chirality center, a dedicated default index is used.

F Supplementary Experimental Results

F.1 Baseline Reproducibility and Evaluation Alignment

To ensure a rigorous and equitable comparison, models included in our protocol-aligned rankings must be evaluated under comparable data distributions, molecular-identity definitions, and inference constraints. Performance metrics for several older foundational baselines are sourced directly from their original publications. This methodological choice is justified by the substantial performance gap between these legacy methods and modern architectures, ensuring that minor evaluation discrepancies do not alter the overarching hierarchical conclusions. Furthermore, many of these early models lack the reliable open-source implementations required for standardized reproduction. Conversely, for recent and highly competitive baselines where precise performance comparison is critical, we identified significant inconsistencies in the evaluation protocols of UAlign [68], EditRetro [13], and RSmiles [71]. Relying on their originally reported figures would introduce confounding variables such as unequal training data volumes, simplified test sets, or mismatched compute budgets. Consequently, we systematically retrained or re-evaluated these specific models to align them with our unified evaluation framework. The specific adjustments are detailed below:

UAlign: The original data preprocessing pipeline for UAlign inadvertently discarded product molecules containing multiple disconnected components within the USPTO-MIT dataset. Chemical inspection reveals that these cases predominantly represent acid-base adducts (salts), which are valid and chemically meaningful species that serve as standard components of retrosynthetic benchmarks. To prevent artificially inflating the model's performance by testing on a simplified subset, we restored these multi-component products to the USPTO-MIT training, validation, and test splits. The UAlign performance reported in our tables reflects the model retrained and re-evaluated exclusively on this corrected and comprehensive data regime.

EditRetro: While the original publication reports strong performance on the USPTO-FULL dataset, the official open-source implementation achieves these results by utilizing a TTA factor of 10. To ensure a fair comparison regarding inference compute budgets and latency across all TTA-reliant baselines, we standardized this sampling factor to 5 to precisely match the settings used by models such as RSmiles [71]. Consequently, we re-evaluated EditRetro on the USPTO-FULL benchmark using the officially provided checkpoints under this standardized sampling constraint.

RSmiles: The originally reported performance of RSmiles relies on a massive pretraining phase utilizing the USPTO-FULL dataset. Evaluating the model on the USPTO-FULL benchmark using its original checkpoint constitutes a fair comparison; therefore, we report these officially provided metrics. However, applying this pre-trained model to the smaller USPTO-50k benchmark would introduce an unfair auxiliary data advantage. To evaluate core architectural efficacy on USPTO-50k, we restricted the pretraining corpus strictly to this dataset and retrained the RSmiles model from scratch. Finally, we do not report RSmiles performance on the USPTO-MIT benchmark. When we restored the chemically valid multi-component products to this dataset, as detailed in the UAlign evaluation above, the RSmiles algorithm encountered fatal execution errors and could not process these standardized inputs.

F.2 Full Top-k Accuracy on USPTO-50K

In this section, we compare our model with the protocol-aligned baselines in Tab. 6.

Table 6: Top-k accuracy (%) on USPTO-50K with reaction class unknown and reaction class known. *: enhanced inference beyond single canonical inference. †: DiffER uses multi-model diffusion ensemble; other * methods use TTA. Bold: best; underlined: second best, in each group (TB, TF, TF*). TB: template-based; TF: template-free.

Model Type	Model	Top-k accuracy (%)							
		Class unknown				Class known			
		1	3	5	10	1	3	5	10
TB	RetroSim [5]	37.3	54.7	63.3	74.1	52.9	73.8	81.2	88.1
	NeuralSym [39]	44.4	65.3	72.4	78.9	55.3	76.0	81.4	85.1
	GLN [8]	52.5	69.0	75.6	83.7	64.2	79.1	85.2	90.0
	LocalRetro [2]	53.4	77.5	85.9	92.4	63.9	86.8	92.4	96.3
	RetroKNN [60]	**57.2**	**78.9**	**86.4**	**92.7**	**66.7**	**88.2**	**93.6**	**96.6**
TF	RetroXpert [63]	50.4	61.1	62.3	63.4	62.1	75.8	78.5	80.9
	G2G [42]	48.9	67.6	72.5	75.5	61.0	81.3	86.0	88.7
	GraphRetro [43]	53.7	68.3	72.2	75.5	63.9	81.5	85.2	88.1
	RetroPrime [54]	51.4	70.8	74.0	76.1	64.8	81.6	85.0	86.9
	Transformer [51]	42.4	58.6	63.8	67.7	57.1	71.5	75.0	77.7
	SCROP [69]	43.7	60.0	65.2	68.7	59.0	74.8	78.1	81.1
	Liu's Seq2seq [31]	37.4	52.4	57.0	61.7	–	–	–	–
	Tied Transformer [24]	47.1	67.1	73.1	76.3	–	–	–	–
	Aug. Transformer [46]	48.3	–	73.4	77.4	–	–	–	–
	MEGAN [36]	48.1	70.7	78.4	86.1	60.7	82.0	87.5	91.6
	GTA [40]	51.1	67.6	74.8	81.6	–	–	–	–
	G2Retro [3]	54.1	74.1	81.2	86.7	63.1	84.2	88.5	91.7
	Graph2SMILES [47]	52.9	66.5	70.0	72.9	–	–	–	–
	Retroformer [52]	53.2	71.1	76.6	82.1	64.0	82.5	86.7	90.2
	Graph2Edits [70]	**55.1**	77.3	83.4	89.4	67.1	**87.5**	91.5	93.8
	NAG2G [66]	**55.1**	76.9	83.4	89.9	67.2	86.4	90.5	93.8
	UAlign [68]	53.5	77.7	84.6	90.6	66.2	86.9	**91.9**	**95.1**
	RSmiles [71]	54.1	76.4	82.0	85.4	–	–	–	–
	EditRetro [13]	50.7	64.0	68.4	73.0	–	–	–	–
	Ours	53.6	**77.8**	**85.1**	**91.2**	**67.8**	**87.5**	91.7	**95.1**
TF*	DiffER*† [7]	57.6	79.0	84.1	87.4	–	–	–	–
	Aug. Transformer*[46]	53.4	–	80.0	84.2	–	–	–	–
	NAG2G* [66]	56.2	78.3	85.5	91.7	–	–	–	–
	UAlign* [68]	52.4	77.2	84.1	91.1	63.6	87.0	91.8	95.0
	RSmiles* [71]	56.3	79.7	85.8	91.1	–	–	–	–
	EditRetro* [13]	**60.8**	**80.6**	86.1	90.1	–	–	–	–
	Ours*	55.8	79.7	**86.6**	**92.6**	**68.6**	**88.5**	**92.8**	**96.0**

F.3 Additional USPTO-50K Results under Different Protocols

For completeness, Tab. 7 collects author-reported USPTO-50K results for recent methods whose training data, inference procedure, or molecular correctness definition differs from our protocol-aligned comparison. We group these results by the source of this difference: external pretraining and augmentation (EP), or a molecular reconstruction and correctness criterion that is not equivalent to complete stereochemistry-aware precursor matching (EM). The entries are reproduced as reported by their authors and provide context rather than values pooled into the protocol-aligned rankings in Tabs. 1 and 6.

Table 7: Author-reported Top-k accuracy (%) on USPTO-50K under training or evaluation protocols that are not pooled into our protocol-aligned ranking. EP: substantial external pretraining; EM: a different molecular reconstruction or correctness criterion. *: the publication's augmented setting, rather than canonical single-input inference. "–" denotes an unreported setting.

Protocol	Model	Top-k accuracy (%)							
		Class unknown				Class known			
		1	3	5	10	1	3	5	10
EP	RSGPT [9]	63.4	84.2	89.2	93.0	72.8	87.7	91.7	94.3
	RSGPT* [9]	77.0	90.9	94.3	96.7	–	–	–	–
	ReactSeq* [61]	58.9	80.5	86.4	91.4	68.5	89.2	93.1	95.9
EM	RetroExplainer [55]	57.7	79.2	84.8	91.4	66.8	88.0	92.5	95.8
	RetroSynFlow [62]	60.0	77.9	82.7	85.3	–	–	–	–
	RetroBridge [16]	50.8	74.1	80.6	85.6	–	–	–	–

External-pretraining protocols. RSGPT [9] is pre-trained on more than ten billion synthetic reaction records generated by applying extracted reaction templates to a fragment library, followed by reinforcement learning from AI feedback and benchmark-specific fine-tuning. ReactSeq [61] likewise benefits from external molecular pretraining on approximately 15 million molecules; its reported result also utilizes extensive SMILES augmentation, including 20-fold test inference. The augmented RSGPT row similarly represents the publication's 20-fold augmentation setting.

Action-based evaluation in RetroExplainer. RetroExplainer [55] formulates retrosynthesis as a constrained molecular assembly process involving leaving-group selection, reaction-center prediction, leaving-group connection, and hydrogen or charge adjustment. In the released evaluator, Top-1 accuracy checks the predicted reaction-center, leaving-group, connection, and hydrogen-change targets, whereas Top-3, Top-5, and Top-10 correctness is derived from the combined rank of the ground-truth reaction-center and leaving-group actions (`mix_pred`). It therefore does not canonicalize every completed candidate and apply the same stereochemistry-aware precursor-SMILES comparison used by our evaluator. Consequently, its reported metric is tied to the coverage and granularity of its predefined action and leaving-group spaces and does not separately test stereochemical identity.

Reduced molecular identity in RetroSynFlow and RetroBridge. RetroSynFlow [62] and RetroBridge [16] generate molecular graphs whose node states encode atom identity and whose edge states distinguish no bond, single, double, triple, and aromatic bonds. Their reported exact-match pipelines canonicalize molecules reconstructed from these reduced graphs. A double-bond edge therefore records bond order but not its E/Z configuration; the reconstruction also lacks tetrahedral chirality. RetroBridge's reconstruction further omits formal charge and explicit hydrogen state. Thus, two precursors that differ only in these omitted attributes can collapse into the same evaluated graph. RetroSynFlow additionally removes test reactions requiring more than the ten appended dummy atoms, retaining 99.4% of the nominal USPTO-50K test set.

F.4 Top-k Accuracy on USPTO-MIT without Multi-Mol Product

To provide a more comprehensive comparison, we evaluate our models on the USPTO-MIT dataset processed by UAlign [68], as summarized in Tab. 8. Compared with the dataset used in the main text,

this version excludes the multi-product reactions discussed in Appendix F.1. Notably, RSmiles [71] was retrained from scratch on this filtered dataset, with both pretraining and fine-tuning restricted to USPTO-MIT. whereas in the original work, pretraining was conducted on USPTO-FULL followed by fine-tuning on USPTO-MIT.

Table 8: Top-k accuracy (%) on USPTO-MIT. *: TTA. **Bold**: best; underlined: second best, per group (TB, TF w/o TTA, TF w/ TTA). TB: template-based; TF: template-free.

Type	Model	Top-k accuracy (%)			
		1	3	5	10
TB	NeuralSym [39]	47.8	67.6	74.1	80.2
	LocalRetro [2]	**54.1**	**73.7**	**79.4**	**84.4**
TF	Liu's Seq2seq [31]	46.9	61.6	66.3	70.8
	AutoSynRoute [30]	54.1	71.8	76.9	81.8
	RetroTRAE [48]	58.3	–	–	–
	RSmiles [71]	55.4	73.3	78.1	81.1
	UAlign [68]	**59.9**	**77.7**	**82.6**	86.3
	Ours	59.8	77.6	82.3	**86.5**
TF*	RSmiles* [71]	56.4	75.7	81.1	85.9
	UAlign* [68]	55.9	76.4	81.6	85.5
	Ours*	**60.7**	**78.8**	**83.6**	**87.6**

On this filtered dataset, our model exhibits trends consistent with the main text. Under canonical inference, MatchRetro achieves nearly the same performance as UAlign while clearly outperforming other baselines. When we apply TTA, MatchRetro gains a substantial margin over all baselines, whereas UAlign's performance decreases. Compared with the main-text results that include multi-component products, UAlign's performance improves after removing these reactions, while MatchRetro remains stable. This stability indicates that our model is robust to multi-component products such as salts or adducts, which frequently appear in practical multi-step synthesis planning. Overall, these findings demonstrate the model's balanced behavior: it maintains strong canonical performance and effectively exploits TTA to further improve accuracy, underscoring its potential for real-world applications.

F.5 Ablation Study

To quantify the contributions of the principal components in our pipeline, we conduct a controlled ablation study on USPTO-50K over five random seeds. Table 9 summarizes the model configurations, which vary implicit target alignment, explicit atom-provenance prediction, training-time SMILES augmentation, the inference-time occupancy constraint, and Stage-I molecular pretraining. Implicit alignment refers to correspondence-aware target construction, whereas explicit alignment refers to predicting atom provenance during decoding. The occupancy constraint applies only when explicit provenance matching is enabled and is therefore not applicable to configurations without explicit alignment. Table 10 reports exact-match top-k accuracy under canonical inference and TTA, and Table 11 reports round-trip accuracy. All results are summarized as the mean and population standard deviation across the five seeds. For each configuration included in the round-trip evaluation, we select canonical inference or TTA according to its overall exact-match top-k profile in Table 10.

Table 9: Configurations used in the USPTO-50K ablation study. A checkmark or cross indicates whether a component is enabled or disabled; "–" denotes a component that is not applicable. Occ. mask denotes the inference-time one-to-one product-node occupancy mask.

Config.	Implicit alignment	Explicit alignment	Training aug.	Occ. mask	Stage I
A	✓	✓	✓	✓	✓
B	✓	✓	✓	×	✓
C	✓	✓	✓	✓	×
D	✓	×	✓	–	✓
E	×	✓	✓	✓	✓
F	×	✓	×	✓	✓
G	×	×	✓	–	✓
H	✓	✓	×	✓	✓
I	×	×	×	–	✓

Table 10: Five-seed top-k exact-match accuracy (%) for the USPTO-50K ablations. Values are mean $\pm$ population standard deviation. **Bold**: best; underlined: second best in each column.

Config.	Without TTA				With TTA			
	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5	Top-10
A	53.61 ± 0.35	$\mathbf{77.84 \pm 0.18}$	$\mathbf{85.06 \pm 0.20}$	$\mathbf{91.20 \pm 0.26}$	55.80 ± 0.22	$\mathbf{79.66 \pm 0.27}$	$\mathbf{86.59 \pm 0.12}$	$\mathbf{92.60 \pm 0.23}$
B	53.64 ± 0.35	77.78 ± 0.24	84.83 ± 0.46	90.95 ± 0.48	$\mathbf{56.03 \pm 0.17}$	79.49 ± 0.19	86.49 ± 0.48	92.50 ± 0.38
C	$\mathbf{53.76 \pm 0.36}$	77.49 ± 0.10	84.27 ± 0.16	90.22 ± 0.27	55.40 ± 0.31	79.14 ± 0.06	86.07 ± 0.15	91.83 ± 0.28
D	53.23 ± 0.21	77.01 ± 0.19	84.04 ± 0.23	90.18 ± 0.39	55.06 ± 0.25	79.05 ± 0.28	86.09 ± 0.21	92.17 ± 0.24
E	49.06 ± 0.28	68.31 ± 0.72	73.07 ± 0.65	76.01 ± 0.71	49.33 ± 0.21	69.97 ± 0.56	75.71 ± 0.73	79.90 ± 0.80
F	48.13 ± 0.32	70.15 ± 0.19	77.41 ± 0.27	83.65 ± 0.26	44.50 ± 0.46	68.23 ± 0.38	76.17 ± 0.32	82.69 ± 0.29
G	45.64 ± 0.34	65.72 ± 0.47	72.31 ± 0.58	79.08 ± 0.57	45.60 ± 0.30	66.04 ± 0.52	72.77 ± 0.62	79.45 ± 0.66
H	50.36 ± 0.17	72.44 ± 0.50	79.62 ± 0.62	86.65 ± 0.61	42.71 ± 0.58	66.87 ± 0.65	75.31 ± 0.35	82.66 ± 0.48
I	44.76 ± 0.31	63.76 ± 0.23	70.14 ± 0.35	76.52 ± 0.23	–	–	–	–

Influence of the inference occupancy mask. The inference-time occupancy mask enforces one-to-one atom assignment by preventing a product node from being selected repeatedly during decoding. Comparing A and B isolates this constraint while keeping all other model components and decoding settings unchanged. The differences are at most 0.25 percentage points in exact-match accuracy and 0.24 points in round-trip accuracy, indicating that most of the predictive benefit comes from the learned matching network rather than this hard constraint. Beam search partly mitigates the removal of the occupancy mask by retaining alternative node assignments and rejecting completed hypotheses that leave product nodes unmatched, which helps explain the small aggregate gap. Nevertheless, this mitigation depends on sufficient search width, and the occupancy mask remains important as a deterministic one-to-one guaranty. For equivalent atoms, such as the two oxygen atoms in O=C=O, deterministic tie-breaking can repeatedly favor the same product node. A narrow beam may discard the alternative assignment, with greedy decoding as the limiting case, whereas the occupancy mask immediately excludes the previously selected node.

Influence of Stage-I pretraining. Stage-I pretraining uses the same-molecule graph-to-SMILES cross-modal translation to initialize structure-aware molecular representations for subsequent reaction fine-tuning. Relative to A, training Stage II from scratch in C leaves Top-1 accuracy similar but reduces deeper candidate retrieval: Top-10 decreases from 91.20 to 90.22 without TTA and from 92.60 to 91.83 with TTA. This pattern indicates that Stage-I pretraining primarily improves structural grounding and candidate coverage, while Stage-II is responsible for learning reaction transformations.

Table 11: Five-seed top-k round-trip accuracy (%) for the evaluated ablation configurations. Values are mean $\pm$ population standard deviation. **Bold**: best; underlined: second best in each column.

Config.	Inference	Top-1	Top-3	Top-5	Top-10
A	With TTA	81.84 ± 0.15	$\mathbf{75.40 \pm 0.31}$	$\mathbf{70.83 \pm 0.54}$	$\mathbf{63.21 \pm 0.80}$
B	With TTA	81.79 ± 0.23	75.16 ± 0.53	70.63 ± 0.86	62.97 ± 1.13
D	With TTA	$\mathbf{81.87 \pm 0.08}$	74.89 ± 0.27	70.10 ± 0.37	62.09 ± 0.56
F	Without TTA	77.95 ± 0.22	68.02 ± 0.24	61.92 ± 0.38	51.07 ± 0.75
G	With TTA	71.15 ± 0.45	58.77 ± 1.14	51.34 ± 1.53	40.81 ± 1.61
I	Without TTA	69.72 ± 0.36	55.44 ± 0.42	47.69 ± 0.57	37.16 ± 0.54

Influence of training augmentation. Comparing A and H shows that SMILES augmentation is beneficial when correspondence is maintained across randomized molecular views. Across the four reported ranks, the canonical gains lie between 3.25 and 5.44 percentage points, while the TTA gains span from 9.94 to 13.09 points. The different behavior of E and F reveals that the relevant factor is not augmentation itself but its compatibility with the target construction. Without implicit alignment, E pairs randomized product traversals with a canonical reactant target. Although the provenance labels remain chemically valid, the relationship between product-node indices and target-token order varies across views. Fixing both orders in F yields gains that increase from 1.84 points at Top-3 to 7.64 points at Top-10 relative to E, despite a 0.93-point decrease at Top-1; applying TTA to F, which is trained only on canonical inputs, decreases all four metrics. Implicit alignment, on the other hand, maintains mutual consistency between product-node indices and target-token order for each

randomized molecular view, allowing the full model to combine stable provenance supervision with the coverage and robustness benefits of augmentation.

Influence of structural alignment. The matched controls isolate the two alignment mechanisms. F versus I holds implicit alignment and augmentation off, showing that explicit provenance prediction raises canonical exact-match accuracy by at least 3.37 and up to 7.27 percentage points; its round-trip gains span from 8.23 to 14.23 points. Conversely, D versus G holds explicit prediction off and augmentation on. Implicit target alignment then raises exact-match accuracy by at least 7.59 points without TTA and 9.46 points with TTA, peaking at 11.73 and 13.32 points, respectively. Its round-trip advantage grows from 10.72 points at Top-1 to 21.28 points at Top-10. Together, these controls demonstrate substantial independent contributions from target-level alignment and model-level provenance supervision.

Explicit provenance prediction also complements implicit alignment. Adding it to D yields the full model A; across the four ranks, exact-match gains fall between 0.38 and 1.02 points without TTA and range from 0.43 to 0.74 points with TTA. Round-trip improvement rises from 0.51 points at Top-3 to 1.12 points at Top-10, while Top-1 is preserved. RSMILES and UAlign use offline alignment to construct favorable product-reactant targets; MatchRetro retains this target-level correspondence while adding atom-level provenance as an explicit model-level decoding signal.

F.6 Computational Efficiency and Inference Latency

Practical multi-step retrosynthesis requires rapid candidate generation to keep the search tractable, particularly under TTA, where stochastic sampling multiplies the inference load. We therefore measured the wall-clock time for the entire 5,000-reaction USPTO-50K [8] test set on a single NVIDIA RTX 4090 GPU. We compared our model with the leading UAlign, RSmiles, and EditRetro baselines under canonical inference and a twenty-sample TTA regime; Fig. 4 shows the complete timing distributions.

Under canonical inference, all evaluated methods return predictions with near-instantaneous latency. Per-sample computational costs range from 0.021 seconds for RSmiles to 0.122 seconds for EditRetro. Our model processes samples in 0.110 seconds, remaining firmly within the operational bounds of modern architectures. Under TTA, inference times scale predictably but remain strictly within interactive planning limits. Our model requires 2.44 seconds per augmented sample, maintaining parity with EditRetro at 2.30 seconds and remaining competitive with UAlign at 1.97 seconds. While RSmiles exhibits a substantial speed advantage at 0.36 seconds per augmented sample, this efficiency stems from its pure character-sequence translation formulation within the highly optimized OpenNMT framework [26]. However, as established previously, this architectural simplicity severely degrades canonical predictive accuracy.

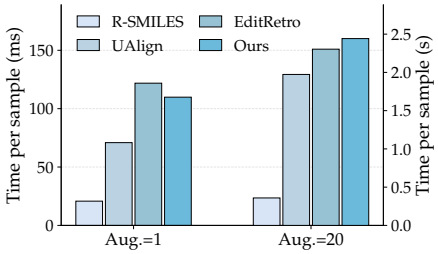


Figure 4: Inference latency per sample on USPTO-50K with and without TTA.

These measurements demonstrate that our proposed cascade is fully competitive with the strongest template-free baselines across both inference paradigms. The approximate two-second per-sample cost incurred during augmentation represents a highly favorable computational trade-off for the substantial top-k accuracy gains documented in Sec. 4.1. By achieving efficient throughput on standard hardware, the model demonstrably satisfies real-world latency requirements for computer-aided synthesis planning without sacrificing generative chemical validity.

F.7 Extended Group-Wise Results and Dataset Statistics

While Sec. 4.2 investigates highly constrained structural edge cases, these complex configurations represent a minor fraction of the overall benchmark, as quantified in Tab. 12. To comprehensively evaluate general architectural efficacy, this appendix analyzes the majority subsets consisting of non-stereochemical and non-ring-transforming reactions. Constituting approximately eighty to ninety percent of the test dataset, these standard transformations serve as the primary drivers of aggregate

benchmark performance. The exact group-wise top-k accuracy and round-trip accuracy values for all subsets are summarized in Tab. 13 through Tab. 16.

Table 12: Statistics of the USPTO-50K test set used in the group-wise analysis. Shares are computed against all 5,007 test reactions. The subtype counts within each family are not mutually exclusive.

Stereo-related subsets			**Ring-related subsets**		
Group	**#Rxns**	**Share (%)**	**Group**	**#Rxns**	**Share (%)**
Any stereo-related	1,043	20.8	Any ring-related	482	9.6
Atom chirality	921	18.4	Ring-opening	361	7.2
Double-bond stereo	149	3.0	Ring-forming	130	2.6
Non-stereo	3,964	79.2	Non-ring-transforming	4,525	90.4

For all group-wise tables below, the MatchRetro* entries report the mean across seeds 2026–2030; baseline entries retain their originally reported single-run values. The plotted group-wise curves likewise show the five-seed means for MatchRetro.

Non-Stereochemical Reactions. On the non-stereochemical subset, the five-seed mean exact-match accuracy is 56.8% at Top-1, below the strongest baseline, approximately tied with the best baseline at Top-3, and highest at Top-5 and Top-10 (87.9% and 93.6%, respectively). For round-trip accuracy, the mean ties with the strongest baseline at Top-1 and is highest from Top-3 onward, reaching 66.3% at Top-10. These results indicate strong performance on this majority subset, while the exact-match advantage depends on retrieval depth.

Table 13: Stereo-group top-k accuracy (%) on USPTO-50K. *: TTA. **Bold**: best; underlined: second best, within each metric column.

Model	**Chirality**				**Double-bond stereo**				**Non-stereo**			
	Top-1	**Top-3**	**Top-5**	**Top-10**	**Top-1**	**Top-3**	**Top-5**	**Top-10**	**Top-1**	**Top-3**	**Top-5**	**Top-10**
RSmiles* [71]	48.8	70.9	76.9	83.1	52.3	72.5	**80.5**	87.9	57.6	81.1	87.0	91.9
UAlign [68]	48.9	71.3	78.3	85.8	45.6	69.8	77.2	86.6	54.9	79.5	86.4	91.5
EditRetro* [13]	51.0	73.0	78.3	82.2	**57.7**	**77.2**	**80.5**	87.2	**62.2**	81.2	87.1	91.0
Ours*	**52.3**	**74.7**	**82.3**	**88.6**	51.9	73.3	78.5	**90.5**	56.8	81.1	**87.9**	**93.6**

Table 14: Stereo-group top-k round-trip accuracy (%) on USPTO-50K. *: TTA. **Bold**: best; underlined: second best, within each metric column.

Model	**Chirality**				**Double-bond stereo**				**Non-stereo**			
	Top-1	**Top-3**	**Top-5**	**Top-10**	**Top-1**	**Top-3**	**Top-5**	**Top-10**	**Top-1**	**Top-3**	**Top-5**	**Top-10**
RSmiles* [71]	77.6	65.3	57.8	48.4	**63.8**	56.4	51.4	43.6	82.9	76.1	70.5	60.9
UAlign [68]	77.7	66.1	58.9	48.2	60.4	56.6	52.8	45.8	82.4	76.2	71.6	61.9
EditRetro* [13]	75.9	62.4	53.7	41.3	**63.8**	50.3	46.6	35.6	**83.1**	69.1	60.6	47.4
Ours*	**78.5**	**68.4**	**61.0**	**51.3**	63.2	**57.0**	**53.8**	**49.4**	**83.1**	**77.6**	**73.6**	**66.3**

Non-ring-transforming reactions. On the non-ring-transforming subset, the five-seed mean exact-match accuracy is below the strongest baseline at Top-1, marginally below the best baseline at Top-3, and highest at Top-5 and Top-10 (88.3% and 93.9%, respectively). The mean round-trip accuracy is highest at all four retrieval depths, reaching 63.5% at Top-10.

Collectively, these standard transformations account for most of the USPTO-50K test set and therefore provide an important test of whether the benefits of explicit structural alignment extend beyond rare, structurally challenging cases. Across both majority subsets, the five-seed means show the clearest exact-match advantages at Top-5 and Top-10, while round-trip accuracy is strongest across retrieval depths, particularly for non-ring-transforming reactions. Thus, the gains are not confined to stereo- or ring-related edge cases: explicit structural matching also supports reliable candidate retrieval and oracle-validated chemical plausibility on common transformations. The advantage is nevertheless metric- and rank-dependent rather than uniform, with smaller or absent exact-match gains at early ranks.

Table 15: Ring-group top-k accuracy (%) on USPTO-50K. *: TTA. **Bold**: best; underlined: second best, within each metric column.

Model	Ring-forming				Ring-opening				Non-ring-transforming			
	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5	Top-10
RSmiles* [71]	36.2	62.3	68.5	75.4	29.9	53.7	61.8	70.4	58.4	81.4	87.3	92.2
UAlign [68]	36.2	50.8	57.7	63.8	31.9	60.4	69.3	79.5	55.8	79.9	86.6	91.9
EditRetro* [13]	**45.4**	**66.2**	**70.0**	73.8	**37.7**	56.8	67.0	74.5	**62.3**	81.8	87.2	90.9
Ours*	40.8	59.7	66.0	**76.3**	36.4	**61.5**	**72.1**	**82.0**	57.8	81.7	**88.3**	**93.9**

Table 16: Ring-group top-k round-trip accuracy (%) on USPTO-50K. *: TTA. **Bold**: best; underlined: second best, within each metric column.

Model	Ring-forming				Ring-opening				Non-ring-transforming			
	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5	Top-10
RSmiles* [71]	73.8	64.6	59.7	48.9	74.2	67.9	63.4	56.4	82.3	74.3	68.3	58.6
UAlign [68]	71.5	65.4	60.5	47.8	72.3	68.9	64.5	56.7	82.0	74.5	69.4	59.6
EditRetro* [13]	**76.2**	60.5	49.4	38.6	74.5	64.7	59.7	48.2	82.0	67.8	59.2	46.0
Ours*	75.7	**70.8**	**67.5**	**59.3**	**74.7**	**70.0**	**66.7**	**60.9**	**82.6**	**75.9**	**71.2**	**63.5**

F.8 Case Study: Mutli-step Retrosynthesis Planning

To assess whether our trained single-step retrosynthesis model can support multi-step retrosynthetic pathway planning, we selected two structurally distinct targets from recently FDA-approved drugs for synthetic route design [56]. In this expert-guided setting, chemical experts served as planners: at each retrosynthetic stage, they evaluated the candidate retrosynthetic steps proposed by MatchRetro and recursively queried the model for plausible precursors. The model used in this case study was trained on the USPTO-FULL dataset. The resulting retrosynthetic pathways are summarized in Fig. 5.

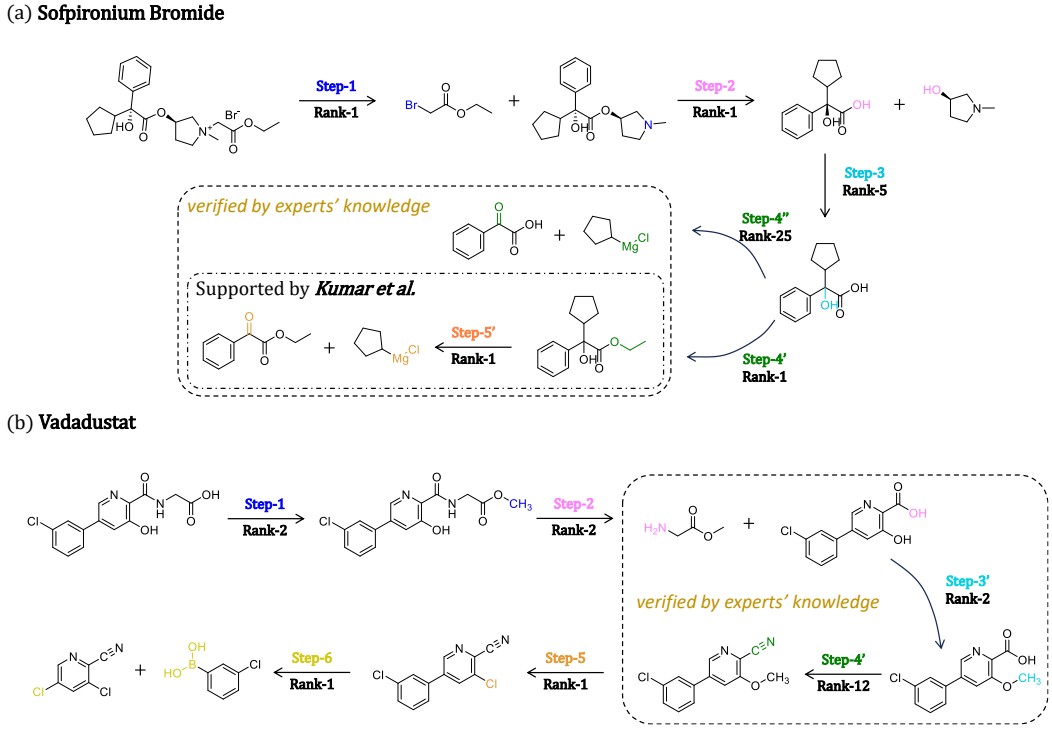


Figure 5: Multi-step retrosynthesis predictions by our method. The pathways illustrated in the figure are based on literature data, with the alternative steps verified by chemical experts. ″ or ′ denotes the steps that not exactly match the literature. **a)** Sofpironium Bromide **b)** Vadadustat.

In the first case study, we evaluated Sofpironium Bromide [1], a recently approved topical anticholinergic agent for primary axillary hyperhidrosis [34, 35]. As illustrated in Fig. 5(a), MatchRetro suggested two concise and synthetically viable retrosynthetic pathways, which share the same first three steps before diverging at the upstream precursor level. Within this shared route segment, the model identified the key quaternization of (R)-1-methylpyrrolidin-3-yl (R)-2-cyclopentyl-2-hydroxy-2-phenylacetate with ethyl 2-bromoacetate as the top-ranked prediction in the first retrosynthetic step. It also prioritized the esterification between (R)-2-cyclopentyl-2-hydroxy-2-phenylacetic acid and (R)-1-methylpyrrolidin-3-ol as the top-ranked prediction in the second step. In the third step, the model further recovered a chiral resolution that selects the required stereochemical configuration of the α-cyclopentylmandelic acid fragment, ranking fifth among the candidate predictions. Together, these shared steps closely recapitulate the established route reported in the patent literature.

The most instructive aspect of this case emerges after the shared late-stage sequence, where the model proposes two upstream strategies for constructing the α-cyclopentylmandelic acid scaffold. The pathway that most closely follows the patent route is shown as step $4''$ in Fig. 5(a), although it is only the 25th-ranked prediction. In this route, the model traces 2-cyclopentyl-2-hydroxy-2-phenylacetic acid back to 2-oxo-2-phenylacetic acid through a Grignard addition. The patent route uses cyclopentylmagnesium bromide for this transformation, whereas MatchRetro predicts the corresponding chloride reagent.

Notably, MatchRetro also suggests an alternative branch composed of two top-ranked predictions. Retrosynthetically, this branch first maps 2-cyclopentyl-2-hydroxy-2-phenylacetic acid to ethyl 2-cyclopentyl-2-hydroxy-2-phenylacetate through the retrosynthetic reversal of ester hydrolysis, denoted as step $4'$ in Fig. 5(a). It then traces this ester intermediate to ethyl 2-oxo-2-phenylacetate through a Grignard addition, denoted as step $5'$ in Fig. 5(a). This ester-based strategy is consistent with the synthetic logic reported by Kumar et al. [27], where the α-hydroxy ester is constructed from the corresponding keto ester using cyclopentylmagnesium bromide. In contrast, MatchRetro predicts cyclopentylmagnesium chloride as a reagent-level variation that preserves the same cyclopentyl organomagnesium equivalent.

Taken together, these two branches indicate that the model captures the underlying organometallic logic rather than memorizing a specific halide identity from the literature. In both cases, the chemically decisive element is the delivery of a cyclopentyl carbanion equivalent to the electrophilic keto-carbonyl precursor, thereby forming the key `C-C` bond and generating the tertiary alcohol center. The replacement of the bromide Grignard reagent with its chloride analog is, therefore, a chemically reasonable deviation; it preserves the same nucleophilic cyclopentylmagnesium species required for the transformation while reflecting a plausible reagent-level variation. This behavior suggests that MatchRetro can recover patent-aligned retrosynthetic logic while also identifying literature-supported alternatives with practical synthetic equivalence.

In the second case study, we selected Vadadustat [29], an FDA-approved hypoxia-inducible factor prolyl hydroxylase (HIF-PHI) inhibitor [41], to further evaluate whether the model can handle route-level deviations from patent precedent.

As illustrated in Fig. 5(b), the model successfully delineated a 6-step retrosynthetic pathway, effectively deconstructing the target molecule into readily available commercial starting materials: (3-chlorophenyl)boronic acid and 3,5-dichloropicolinonitrile. With the exception of one key intermediate transformation, the predicted steps, including the Suzuki cross-coupling, methoxylation via nucleophilic aromatic substitution, and the late-stage amidation and ester hydrolysis, are highly consistent with the established patent literature, with most steps ranking within the top 2 choices.

The most compelling feature of this case is that the model proposes a chemically rational alternative precisely at the step where it deviates from the patent route. In the reported patent route, the hydrolysis of the picolinonitrile group and the cleavage of the methoxy ether are executed simultaneously in a single step under exceedingly harsh conditions, such as refluxing in 48% hydrobromic acid. Rather than reproducing this severe one-pot transformation, MatchRetro decouples it into a milder two-step sequence: hydrolysis of the nitrile to the corresponding picolinic acid intermediate, followed by an independent demethylation step to yield 5-(3-chlorophenyl)-3-hydroxypicolinic acid, corresponding to Step 3 and Step 4 in Fig. 5(b).

By separating a drastic dual transformation into two sequential steps, the proposed route avoids the direct use of hazardous reagents and harsh thermal conditions in the patent procedure, which may

help reduce the risk of undesired side reactions. This divergence from the patent literature highlights a complementary aspect of MatchRetro's predictions compared with the Sofpironium Bromide case: rather than introducing a reagent-level variation while preserving the same organometallic transformation, the model outputs a route in which the order and granularity of functional group transformations are adjusted in a chemically plausible manner. Together, the two case studies indicate that MatchRetro has the potential to assist expert-guided multi-step planning by recovering patent-consistent steps and suggesting plausible alternatives at the levels of reagents, intermediates, and route design.

G Representational Limitations in Double-Bond Stereochemistry

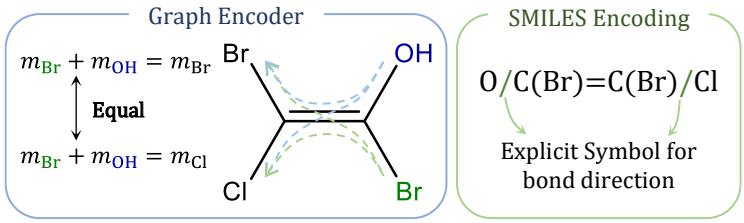


Figure 6: An illustration of implicit and explicit encoding of double-bond stereochemistry.

As shown in Fig. 6, graph-based encoders encode double-bond stereochemistry only implicitly. In an MPNN-style encoder, substituents on one side of a `C=C` bond, such as `Br` and `Cl`, receive the same aggregated messages from the substituents on the opposite side. Although stereochemical information may be stored in the double-bond edge features, this information does not directly specify the local orientation of each substituent; instead, the model must recover it through implicit reasoning over compressed message representations. Since the relevant information must propagate across several message-passing steps, the stereochemical signal can become diluted, making this positional reasoning more difficult. By contrast, SMILES-based methods explicitly mark bond directions with `/` and `\`, giving the model direct access to the relative orientation around the double bond.

H Broader Impacts

This work may benefit computer-aided synthesis planning by improving the reliability of single-step retrosynthesis prediction, which could accelerate molecular design in areas such as medicinal chemistry and materials discovery. More accurate candidate generation may also improve the efficiency of downstream multi-step planning systems.

At the same time, retrosynthesis models are dual-use technologies and could lower the barrier to planning synthetic routes for harmful compounds. In addition, incorrect predictions may be over-trusted if used without expert verification. Our method is intended as a decision-support tool rather than a replacement for human judgment, and practical use should retain appropriate chemical oversight.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the paper's main contributions, including the explicit structural-alignment formulation, the matching-while-decoding architecture, and the claimed empirical advantages. These claims are supported by the methodological sections and the reported experimental results on USPTO-50K, USPTO-MIT, and USPTO-FULL.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The Limitations are Discussed in Sec. 5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: This is a application-oriented paper, there is no theroems here.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper specifies datasets, preprocessing, evaluation metrics, reranking, structural constraints, and model hyperparameters in the appendix. See Appendix C, Appendix D and Appendix E.3.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The MatchRetro implementation is publicly available at https://github.com/zengkaipeng/MatchRetro. The raw benchmark datasets are publicly available from their original sources, and the data preprocessing, evaluation protocol, and implementation details are documented in Appendix C, Appendix D, and Appendix E.3.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the processed data splits, preprocessing rules, optimizer, learning-rate schedule, training epochs, effective batch sizes, test-time augmentation, and dataset-specific hyperparameters. See Appendix C and Appendix E.3.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We follow the common evaluation protocol in this literature and report one run on the fixed benchmark splits, without error bars or repeated-seed significance analysis. Prior work such as [68] suggests that seed sensitivity is relatively small in this setting.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We report the GPU type, distributed training setup, effective batch sizes, and an inference-time benchmark in Appendix E.3 and Appendix F.6.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Yes, the research conducted in this paper conform with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

Justification: The paper discusses both positive and negative societal impacts in Sec. H, including benefits for computer-aided synthesis planning and potential dual-use risks.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: This paper poses no such risks.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The original datasets and software assets are credited in the paper through citations, and the datasets used in this work follow MIT licenses.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper introduces the MatchRetro implementation, which is documented and released at `https://github.com/zengkaipeng/MatchRetro`. The repository provides the implementation and usage information needed to work with the released code.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: We only use LLM for polishing the wordings and making visualization scripts.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.