Retrieval-Augmented Foundation Models for Matched Molecular Pair Transformations to Recapitulate Medicinal Chemistry Intuition

Bo Pan[1], **Yang Qiao**[1], **Peter Zhiping Zhang**[2], **Hao-Wei Pang**[2], **Alex Zhu**[1], **Xiang Yu**[2], **Liying Zhang**[2], **Liang Zhao**[1]
[1]Emory University, Atlanta, GA, USA
[2]Merck & Co., Inc., Rahway, NJ, USA
{bo.pan, liang.zhao}@emory.edu

Abstract

Matched molecular pairs (MMPs) capture the local chemical edits that medicinal chemists routinely use to design analogs, but existing ML approaches either operate at the whole-molecule level with limited edit controllability or learn MMP-style edits from restricted settings and small models. We propose a variable-to-variable formulation of analog generation and train a foundation model on large-scale MMP transformations (MMPTs) to generate diverse variables conditioned on an input variable. To enable practical control, we develop prompting mechanisms that let the users specify preferred transformation patterns during generation. We further introduce MMPT-RAG, a retrieval-augmented framework that uses external reference analogs as contextual guidance to steer generation and generalize from project-specific series. Experiments on general chemical corpora and patent-specific datasets demonstrate improved diversity, novelty, and controllability, and show that our method recovers realistic analog structures in practical discovery scenarios.

1 Introduction

In drug discovery, a fundamental strategy to optimize lead molecules is *analog design*, which involves medicinal chemists leveraging their intuition to make localized, knowledge-driven edits to existing molecules, instead of designing entirely novel molecules (Fischer and Ganellin, 2010; Choung et al., 2023; Shurtleff et al., 2024). Machine learning models for molecular optimization have increasingly adopted transformation-based formulations, learning to transform one molecule into another through graph edits or sequence-to-sequence generation (Tysinger et al., 2023; Tibo et al., 2024; He et al., 2022, 2021). However, most such approaches treat transformations as implicit, molecule-level operations, without distinguishing which edits correspond to chemically meaningful local modifications versus arbitrary global rewrites (Ozcelik et al., 2025). In contrast, medicinal chemists typically reason in terms of **matched molecular pairs (MMPs)** (Dossetter et al., 2013; Yang et al., 2023; Fischer and Ganellin, 2010; Choung et al., 2023; Andreev et al., 2020; Meanwell, 2017; Papadatos et al., 2010; Griffen et al., 2011; Qiao et al., 2026), which are pairs of molecules that differ by discrete and minimal modifications, such as R-group substitutions or core replacements, that preserve synthetic feasibility and support interpretable structure–activity comparisons, as shown in Fig. 1 (a). Abstracting away the constant chemical context and focusing solely on the localized edit yields an **MMP transformation (MMPT)**, which captures a context-independent medicinal chemistry modification corresponding to a single, well-defined *variable-to-variable* change, as an example shown in Fig. 1 (b). MMPTs directly reflect how chemists use their medicinal chemistry intuition to optimize lead molecules, and provide a principled way to represent and learn transferable medicinal chemistry modifications across different molecular contexts (Pang et al., 2026).

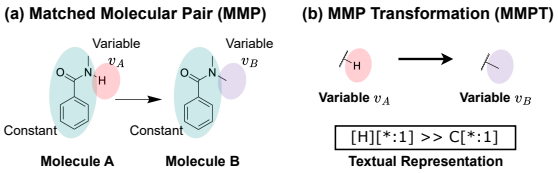


Figure 1: An example of (a) Matched Molecular Pairs (MMP); (b) Matched Molecular Pair Transformation (MMPT) and its textual representation.

While MMPTs have appeared occasionally in machine learning-based molecular optimization, they remain underexploited as a first-class generative representation: prior models are typically limited in scale or trained with specific data (Chen et al., 2021; Jin et al., 2020), or embedded within molecule-level generators where MMP relationship

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 30210–30223
October 24-29, 2026 ©2026 Association for Computational Linguistics

is only weakly enforced (Tysinger et al., 2023; Tibo et al., 2024; He et al., 2022, 2021). At the same time, recent advances in large-scale chemical corpora (Gaulton et al., 2012; Sun et al., 2017; Yu et al., 2024) and foundation models (Achiam et al., 2023; Bai et al., 2023) make it newly feasible to learn generalizable, MMPT-level priors directly from millions of real data points. This convergence creates a timely opportunity to revisit MMPTs: not as an auxiliary constraint, but as the central abstraction for controllable and scalable analog generation.

Despite their conceptual appeal, realizing MMPT-centric learning within modern ML systems poses several technical challenges. First, many existing analog generation models are optimized for molecule-level similarity rather than explicit localized modifications (Tibo et al., 2024; He et al., 2022), making it difficult to guarantee that generated candidates differ from the input by a single, well-defined transformation. As a result, users cannot reliably specify where a modification should occur (e.g., which R-group or core), nor ensure that unintended global changes are avoided. Second, prior MMPT-based learning approaches are often constrained to small models, limited transformation vocabularies, or narrowly curated datasets, which restrict their ability to absorb the long-tailed, heterogeneous transformation patterns observed across large chemical corpora (Chen et al., 2021; Jin et al., 2020). Third, existing controllable methods, whether graph-editing models with fixed operators (Chen et al., 2021; Jin et al., 2020) or prompt-based use of large language models (Wu et al., 2024; Wang et al., 2024; Brown et al., 2020), lack mechanisms to learn transferable, transformation-level priors that generalize across scaffolds while remaining interpretable and synthetically plausible. Finally, practical deployment in medicinal chemistry requires models to adapt to user- or project-specific preferences, such as emphasizing rare but relevant modifications or following established series patterns, without costly retraining (Zhang et al., 2024). Addressing these challenges requires moving beyond molecule-level generators toward scalable, controllable models that operate directly in the MMPT space.

Motivated by the above complementarity and limitations, we aim to build a practical MMPT-centric generation framework that synergizes chemists' intuition with the ability of modern ML to learn from massive data. First, we train an MMPT foundation model on large-scale transformation data to generate variables conditioned on input variables, enabling high-throughput analog design in a user-controlled edit region. Second, we develop prompting mechanisms that expose substructure-level control to users, allowing them to customize the structural patterns of generated variables. Third, we introduce an MMPT-RAG framework that incorporates external reference datasets: retrieved analogs are organized into structural clusters and used to reweight the generation distribution, improving coverage of infrequent yet chemically meaningful transformations while preserving plausibility.

Our main contributions are summarized as follows:

- We formalize analog generation in the MMPT space, treating it as context-independent local edits composable across diverse molecular scaffolds.
- We train a foundation model on large-scale MMPTs and enable controllable generation via prompting mechanisms that specify desired structural patterns.
- We propose MMPT-RAG, which retrieves and clusters reference analogs to steer generation toward user-preferred patterns without retraining.
- We validate our approach on in-distribution, within-patent, and cross-patent benchmarks, consistently improving recall of ground-truth transformations while maintaining high validity and novelty.

2 Related Work

MMP-Based Analog Generation. Most of the existing MMP-based analog design methods explore the setting of molecule-level generation, i.e., generating entire molecules that are similar to the given molecule (Jin et al., 2018; Tysinger et al., 2023; Tibo et al., 2024; He et al., 2022, 2021), and thus this stream of methods does not support users specifying a substructure (variable) to edit. Some existing methods also allow users to specify a structure to change, with one stream of work formulating it as a partial molecule completion problem, including those VAE-based graph generative models which operate via node generation (Jin et al., 2018, 2020; Lim et al., 2020; You et al., 2018;

Pan et al., 2025; Ghaemi et al., 2026) and autoregressive generative models that implements it as a token generation task in the SMILES space (Olivecrona et al., 2017; Loeffler et al., 2024). With the advancements of LLMs, some work also explored leveraging LLMs' zero-shot ability to suggest variable replacements (Wu et al., 2024; Wang et al., 2024; Brown et al., 2020). Among these methods, the LibINVENT module (Loeffler et al., 2024) in REINVENT 4 (Loeffler et al., 2024) stands out as a strong baseline with a wide industrial applicability, benefiting from its large training corpus. However, all the above methods target learning the conditioned molecular completion task. To our best knowledge, no existing method attempts to directly learn a context-independent variable replacement objective.

3 Problem Formulation

We formalize analog design as variable-to-variable generation in MMPT space. An MMPT is defined as a transformation $v_A \rightarrow v_B$ where v_A and v_B are substructural variables that differ by a single localized edit, while the rest of the molecule remains unchanged (Fig. 1). Given an input variable v_A, our goal is to generate a set of candidate substitutions $\{v_B^{(1)}, v_B^{(2)}, \dots\}$ that constitute valid MMPTs.

Unlike molecule-level generation where edits are implicit and entangled across the structure, directly operating in MMPT space recasts the task as learning conditional distributions over local chemical transformations rather than entire molecules. However, this formulation introduces several challenges: (i) the MMPT space is large and highly imbalanced, where rare but meaningful transformations are difficult for small models to learn; (ii) models using formal chemical languages capture precise structure but are hard to control via prompts, whereas natural language models support flexible prompting but lack explicit structural enforcement; and (iii) practical use demands distributional steering toward project-specific patterns without retraining.

4 Methodology

To address the challenges outlined above, we propose a two-component framework centered on MMPTs. First, to address the challenges of the large, long-tailed MMPT search space and the challenge in enabling flexible user control, we introduce MMPT-FM, a foundation model trained directly on large-scale MMPT data. By modeling variable-to-variable transformations in a chemically grounded token space and supporting structured prompting, MMPT-FM learns generalizable transformation-level priors while enabling user-guided generation. To address the challenge of explicit, project-level controllability without retraining, we introduce MMPT-RAG, a retrieval-augmented generation framework that incorporates external reference analogs as guidance, allowing users to bias generation toward preferred transformation patterns and project-specific needs.

4.1 MMPT-FM: A Promptable MMPT Foundation Model

Here, we construct our foundation model, MMPT-FM, that operates directly in the MMPT space, learning variable-to-variable transformations as the primary generative unit. We further equip it with a prompting mechanism that conditions generation on optional user-specified structural templates, thereby enabling more flexible user control.

4.1.1 Training of MMPT-FM.

We train our foundation model on pairs of localized transformation $v_A \rightarrow v_B$, where both v_A and v_B are represented as SMARTS (Daylight Chemical Information Systems, 2019), a textual chemical representation method for substructures. Formally, let Σ denote a finite vocabulary of SMARTS tokens, including atomic symbols (e.g., C, N, O), bond descriptors (e.g., =, #), and other syntactic elements for SMARTS expressions. Each variable is represented as a sequence over Σ as $v_A = (\tau_1^A, \dots, \tau_n^A), v_B = (\tau_1^B, \dots, \tau_m^B)$, where $\tau_i^A, \tau_j^B \in \Sigma$. An MMPT is therefore formulated as a conditional sequence-to-sequence mapping from v_A to v_B, and the model learns the conditional distribution $p(v_B \mid v_A)$ defined over syntactically valid and chemically plausible SMARTS variables.

To construct the training data, we extract MMPTs from ChEMBL (Gaulton et al., 2012) using MMPDB (Dalke et al., 2018), yielding approximately 2.6M unique transformations after filtering for drug-like compounds. We model this using an encoder-decoder Transformer initialized from T5Chem (Christofidellis et al., 2023; Lu and Zhang, 2022), a chemically pretrained T5-style model. Implementation details and full dataset statistics are provided in Appendix A and B.

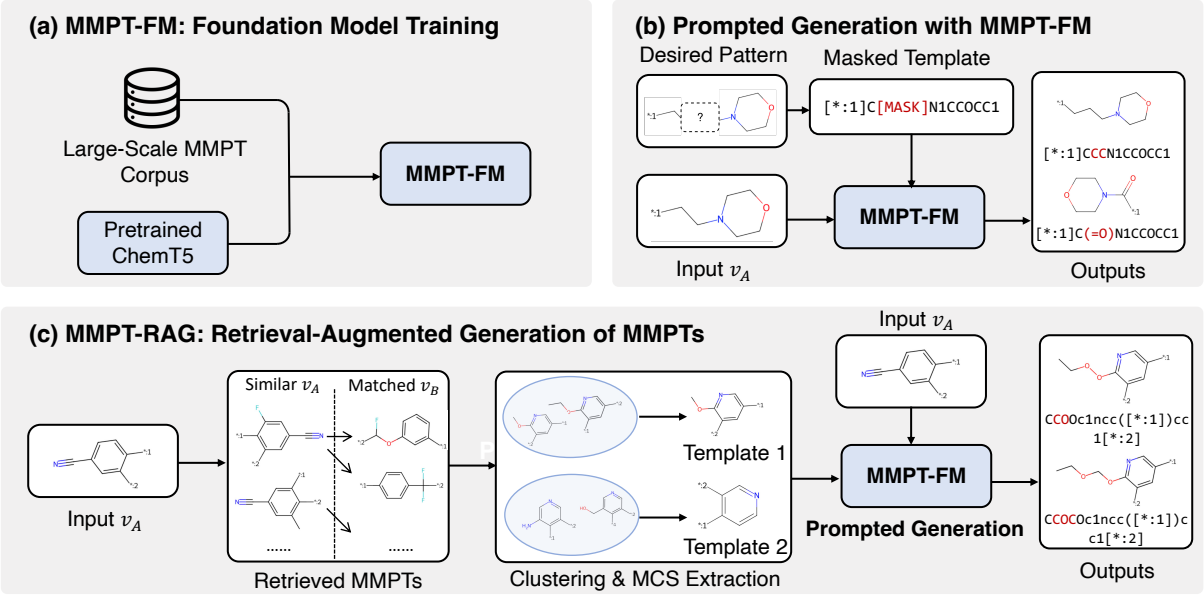


Figure 2: Overview of the proposed MMPT framework. (a) The foundation model (MMPT-FM) is trained on large-scale MMPT data. (b) MMPT-FM supports controllable generation via masked template prompting. (c) MMPT-RAG augments generation with retrieval, clustering, and MCS-based template extraction to guide context-aware transformation generation.

4.1.2 Prompted Generation with MMPT-FM.

In practical medicinal chemistry workflows, analog design is rarely unconstrained; chemists often seek to impose explicit structural conditions, such as preserving a core motif or exploring a specific transformation family, making it essential for an MMPT generation framework to support user-specified structural patterns. Unlike natural language models, where intent can be expressed through appended text instructions, MMPT-FM operates purely in chemical token space, so desired structural patterns cannot be specified as free-form text. To preserve the variable-to-variable formulation while enabling control, we encode user intent as partial structural constraints on the output.

The prompted generation process is illustrated in Fig. 2 (b). Specifically, we formulate prompted generation of MMPT-FM as a constrained sequence completion task, where the user can impose a structural constraint S that defines the preferred chemical substructure to be preserved in the generated v_B, with some undefined positions that the model is asked to complete. Formally, we define a masked template T as $T = (\tau_1, \tau_2, \ldots, \tau_L)$, where each $\tau_i \in \Sigma \cup \{\texttt{[MASK]}\}$. The mapping from a user-specified structural constraint S to a textual template T is operated by preserving the chemical tokens corresponding to the fixed substructure S while replacing the undefined positions with [MASK] tokens.

The generation objective is to find complete variable sequences v_B that are compatible with T. At inference time, the generator performs approximate inference to produce a set of candidate variables $\mathcal{V}_B = \{v_b^{(1)}, \ldots, v_b^{(K)}\}$ that complete the masked positions conditioned on T. These candidates are selected to have high likelihood under the model as

$$\begin{aligned}&\text{PromptGen}(v_A, \tilde{T}, K)\\ =&\underset{\mathcal{S},|\mathcal{S}|=K}{\text{argmax}} \sum_{v_B^{(i)} \in \mathcal{S}} \log p_\theta\left(v_B^{(i)} \mid v_A, T\right),\end{aligned} \tag{1}$$

where θ denotes the parameters of the trained MMPT-FM, and K is the desired number of output variables.

We implement masked infilling via an explicit search over the space of possible span completions, using the model likelihood to score each candidate. To make this search tractable, in practice, we limit the branching factor at each masked position by selecting only the effective number N_{eff} (Jost, 2006) of high-probability tokens, calculated by $N_{\text{eff}} = 2^{H_2(p)}$, where p is the token probability distribution of generating each token, and $H_2(p) = -\log_2 \sum_i p_i^2$ is the Rényi entropy (Rényi, 1961) of order 2 of the token distribution p. Starting from the initial masked template, we conduct a tree search by iteratively filling masked positions with

these candidate tokens, and accumulating sequence-level log-likelihoods from the model. Finally, we rank completed candidates by their likelihood and return the top-k infilled fragments as the model's suggested variable substitutions.

4.2 MMPT-RAG: A RAG Framework for MMPT Generation

MMPT-FM learns a global prior over chemically meaningful MMPTs from large-scale data, whereas practical design often relies on project-specific reference analogs. A principled generator should therefore interpolate between general chemical knowledge and project-specific patterns, rather than overriding one with the other. We realize this idea through MMPT-RAG, which integrates retrieval as guidance (Section 4.2.1) and admits a theoretical interpretation as an explicit distribution shift from model prior to reference dataset distribution (Section 4.2.2).

4.2.1 Workflow.

MMPT-RAG guides the generation towards a reference database $\mathcal{D}$. Let $\mathcal{D} = \{(v_A^{(i)} \to v_B^{(i)})\}_{i=1}^N$ denote a reference database of MMPTs. Given an input variable v_A, the framework retrieves relevant MMPTs, clusters them to identify representative patterns, and converts each cluster into a structural template. These templates are then used to prompt MMPT-FM via masked infilling, as illustrated in Fig. 2 (c). The workflow proceeds in three steps detailed below.

Step 1: Retrieval with input similarity. To leverage the useful examples from the retrieval dataset, we retrieve variables structurally similar to the query. Let $\psi(\cdot)$ denote an embedding function, which is usually implemented with the Morgan fingerprint (Morgan, 1965). For query v_A, we retrieve top-K nearest neighbors $\mathcal{V}_A$ from the database $\mathcal{D}$ as

$$\mathcal{V}_A = \mathrm{Retrieve}(v_A; \mathcal{D}) = \left\{ v_A^{(i)} \;\middle|\; \begin{array}{c} (v_A^{(i)} \to v_B^{(i)}) \in \mathcal{D}, \\ i \in \mathrm{TopK}\Big(\mathrm{sim}\Big(\psi(v_A), \psi(v_A^{(i)})\Big)\Big) \end{array} \right\}. \quad (2)$$

where $\mathrm{sim}(\cdot)$ denotes a function to calculate similarity, usually implemented using cosine similarity, yielding candidate contexts $v'_A \in \mathcal{V}_A$. Given a retrieved set of input variables $\mathcal{V}_A = \{v_A^{(i)}\}$, we denote by $\mathcal{V}_B = \{v_B^{(j)}\}$ the set of all variables such that there exists some i, j that $(v_A^{(i)} \to v_B^{(j)})$ forms an MMPT in $\mathcal{D}$ as

$$\mathcal{V}_B = \{v_B \mid \exists v_A \in \mathcal{V}_A, (v_A, v_B) \in \mathcal{D}\} \quad (3)$$

Step 2: Clustering of Retrieved Examples. To extract representative structural patterns in retrieved $\mathcal{V}_\mathcal{B}$, we perform clustering over the retrieved outputs. We compute embeddings $\phi(v_B)$ for all $v_B \in \mathcal{V}_\mathcal{B}$ and partition them into K clusters

$$\mathcal{C}_1, \ldots, \mathcal{C}_K = \mathrm{Cluster}(\{\, \phi(v_B) : v_B \in \mathcal{V}_B \,\}), \quad (4)$$

using a clustering algorithm, which is implemented as HDBSCAN (McInnes et al., 2017) in this work.

Step 3: Cluster pattern-prompted generation. For each cluster $\mathcal{C}_k$, we first identify its invariant chemical scaffold. The invariant substructure is defined as the Maximum Common Substructure (MCS) (Raymond et al., 2002) among these retrieved molecules as $S_k = \mathrm{MCS}(\mathcal{C}_k)$. In practice, this can be automatically computed using the `rdkit.Chem.MCS.FindMCS` function (Landrum, 2013), which finds the largest subgraph common to all variables in $\mathcal{C}_k$. Then we construct the masked template T_k for each MCS S_k via the approach introduced in Section 4.1.2. The resulting T_k serves as the cluster-invariant template, which is further used as the prompt to guide the generation.

Given cluster-specific templates $\{T_k\}_{k=1}^K$ and an (optional) user-specified preference distribution $\tilde{\boldsymbol{\pi}}(x)$, MMPT-RAG generates outputs from each template proportionally to its assigned weight. Formally, the RAG output is defined as

$$\mathrm{RAG}(v_A) = \bigcup_{k=1}^{K} \mathrm{PromptGen}(v_A, T_k, N_k), \quad (5)$$

where N_k denotes the generation budget allocated to cluster k and $N_k \propto \tilde{\pi}_k(x)$, $\tilde{\boldsymbol{\pi}}(x)$ is the user-specified preferred cluster distribution for each cluster C_k:

$$\tilde{\boldsymbol{\pi}}(x) = \big(\tilde{\pi}_1(x), \ldots, \tilde{\pi}_K(x)\big), \tilde{\pi}_k(x) \geq 0, \quad \sum_{k=1}^{K} \tilde{\pi}_k(x) = 1, \quad (6)$$

where $\tilde{\pi}_k(x)$ can encode arbitrary user preferences.

4.2.2 Theoretical Analysis.

To analyze the RAG mechanism, we formalize the interaction between the foundation model and the reference set as a Bayesian integration. We show that MMPT-RAG performs a global distribution shifts toward the reference dataset while maintaining the knowledge of the foundation model

Theorem 4.1 (Global Steering). *Let $p_\theta(y \mid x)$ be the conditional distribution over variables $y \in \mathcal{V}$ defined by the unconstrained foundation model. Assume that for each cluster k, prompting the model with template T_k (via masked infilling) results in a local distribution $p(y \mid x, T_k)$ that is an adaptive interpolation between the model's prior and the cluster-specific reference $p(y \mid T_k)$:*

$$p(y \mid x, T_k) = (1 - \alpha_k) p_\theta(y \mid x) + \alpha_k p(y \mid T_k), \tag{7}$$

where $\alpha_k \in (0, 1]$ is an adaptive gating factor *reflecting the model's adherence to template T_k under context x.*

Then, the global RAG distribution defined in (5), $p_{RAG}(y \mid x)$, satisfies:

$$p_{RAG}(y \mid x) = (1 - \bar{\alpha}) p_\theta(y \mid x) + \bar{\alpha} p^*_{ref}(y \mid x), \tag{8}$$

where $\bar{\alpha} = \sum_k \tilde{\pi}_k \alpha_k$ and $p^*_{ref}(y \mid x) = \sum_k \frac{\tilde{\pi}_k \alpha_k}{\bar{\alpha}} p(y \mid T_k)$.

Proof of Theorem 4.1. See Appendix F.

5 Experiments

5.1 Experimental Setup

5.1.1 Main Experiment Tasks.

We evaluate our framework on three progressively more realistic analog-generation tasks.

Task 1: In-distribution MMPT Generation. This task evaluates whether the model can recover valid and novel local transformations under an in-distribution setting. We use the 10% held-out split of the ChEMBL MMPT dataset, constructed with the same MMPT extraction pipeline as training but with disjoint MMPTs.

Task 2: Within-Patent Analog Expansion. This task evaluates MMPT generation within a real-world medicinal chemistry project. We use the PMV Pharmaceutical patent dataset (PMV17) (Vu et al., 2017); dataset statistics are reported in Appendix B. This setting tests whether the model can discover promising variables within an active patent series.

Task 3: Cross-Patent Follow-up Generation. This task evaluates whether a model can propose forward-looking MMPTs that appear in later patents, reflecting temporal medicinal chemistry progression. We construct a patent-to-patent setting by extracting MMPTs linking compounds from PMV17 (2017) to those in PMV21 (Vu et al., 2021) (2021), both from PMV Pharmaceuticals patent filings.

5.1.2 Compared Methods.

Since no existing method is explicitly designed for the variable-to-variable MMPT formulation, we compare against: **Retrieval** (nearest-neighbor lookup in the reference set); **GPT-5.4**, a general-purpose LLM; and **REINVENT4** (Lib-INVENT) (Loeffler et al., 2024), a molecule-level analog generation framework conditioned on the constant scaffold. For our own methods, we report **MMPT-FM** (foundation model only) and **MMPT-RAG** (our full framework).

5.1.3 Evaluation Metrics.

We report three metrics: **Valid** (percentage of chemically valid variables with the correct number of attachment points), **Novel** (percentage of outputs not seen during training; reported as **Novel/valid** and **Novel/all**, i.e., among valid outputs or among all outputs), and **Recall** (percentage of ground-truth transformations recovered). For patent-based tasks, we further report **Recall-i** and **Recall-o** for transformations inside and outside the training set, respectively.

5.1.4 Implementation Details.

All implementation details are given in Appendix A.

5.2 Main Results

Results for all three tasks are presented in Table 1.

Looking across all three tasks, the baselines each exhibit characteristic weaknesses. Database retrieval achieves moderate recall (22.7–43.5%) but zero novelty and out-of-training coverage. REINVENT4 produces novel outputs but suffers from low recall ($<13\%$) and validity ($<33\%$) across all tasks. GPT-5.4 attains the highest novelty among baselines yet achieves recall of at most 26.6%, confirming that general-purpose LLMs lack the specialized transformation coverage needed for MMPT prediction.

In contrast, our framework substantially outperforms all baselines on every metric. MMPT-FM improves recall by at least >24 points over the best baseline across all tasks (67.6% vs. 43.5% on Task 1; 41.4% vs. 22.7% on Task 2; 43.8% vs. 28.6% on Task 3), while maintaining validity above 98%. MMPT-RAG pushes recall further to 82.1%, 49.2%, and 46.8% respectively—a relative improvement of $>18\%$ over MMPT-FM—and achieves the strongest out-of-training-set recovery (Recall-o = 15.2% on Task 2, 13.0% on Task 3).

Table 1: MMPT generation performance on three tasks.

(a) Task 1: ChEMBL MMPT Dataset

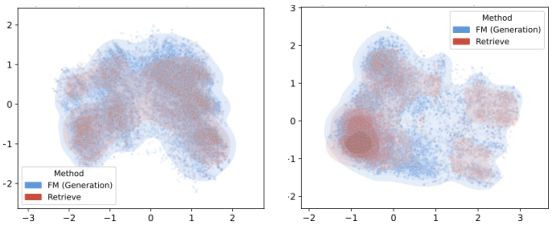
Method	Recall	Novel/val.	Novel/all	Valid
Retrieval	43.5%	0.0%	0.0%	100%
GPT-5.4	5.3%	74.5%	60.3%	81.0%
REINVENT 4	12.7%	23.0%	5.6%	24.4%
MMPT-FM	67.6%	26.0%	25.8%	99.3%
MMPT-RAG	82.1%	30.1%	29.8%	99.1%

(b) Task 2: PMV17 MMPT Dataset

Method	Recall			Novel		Valid
	Total	/i	/o	/valid	/all	
Retrieval	22.7%	29.4%	0.0%	0.0%	0.0%	100%
GPT-5.4	26.6%	31.4%	10.1%	73.5%	61.7%	83.9%
REINVENT 4	5.1%	7.1%	0.0%	48.2%	15.7%	32.5%
MMPT-FM	41.4%	52.1%	13.2%	23.0%	22.7%	98.9%
MMPT-RAG	49.2%	62.1%	15.2%	23.7%	23.4%	98.6%

(c) Task 3: PMV17-PMV21 Cross-Patent Generation

Method	Recall	Recall/i	Recall/o
Retrieval	28.6%	57.5%	0.0%
GPT-5.4	20.0%	35.0%	5.0%
REINVENT 4	7.4%	12.2%	1.9%
MMPT-FM	43.8%	76.5%	11.5%
MMPT-RAG	46.8%	81.4%	13.0%



(a) ChEMBL dataset. (b) PMV17 dataset.

Figure 3: Visualizations of the chemical space explored by our foundation model MMPT-FM (blue) versus Retrieval (red) on (a) ChEMBL and (b) PMV17 datasets.

These results demonstrate that MMPT-centric foundation models combined with retrieval augmentation consistently outperform both conventional and LLM-based approaches across diverse analog-generation settings.

5.3 Decoupled Analysis

Evaluation of MMPT-FM's Chemical Space Coverage. To assess the fundamental generative capability of MMPT-FM, we evaluate its ability to explore the chemical space compared to Retrieval. We utilize a randomly sampled group of 50 unique variables from both ChEMBL and PMV17 to generate candidate sets. The structural distributions are visualized by projecting their Morgan fingerprints using PCA, comparing the spans of FM-generated candidates against retrieved variables.

Table 2: Effect of beam size on average validity of MMPT-FM, averaged on the ChEMBL MMPT held-out test set.

Beam	400	600	800	1000	1200
Avg Validity	0.9992	0.9991	0.9989	0.9988	0.9986

Table 3: Prompted generation on ChEMBL and PMV17 at different numbers of generations (K). We report Validity, Recall of ground truth, and numbers of generated unique and valid variables (#Unique).

	ChEMBL			PMV17		
K	Valid (%)	Recall (%)	#Uniq	Valid (%)	Recall (%)	#Uniq
1	100.0	58.0	1.00	98.0	46.0	0.98
10	86.0	96.0	8.60	79.2	90.0	7.92
20	74.1	100.0	14.82	62.9	96.0	12.58
50	48.0	100.0	24.00	39.24	98.0	19.62
100	31.7	100.0	31.70	24.84	98.0	24.84
200	20.8	100.0	41.60	15.79	98.0	31.58

Visual analysis of the PCA projections (Fig. 3) reveals that MMPT-FM (blue) consistently explores a substantially larger chemical volume compared to Retrieval (red), demonstrating superior extrapolation beyond the training distribution.

We further study how beam size affects the validity of MMPT generation on the ChEMBL MMPT held-out test set. As shown in Table 2, validity remains consistently high across all beam sizes, with only a slight decrease from 0.9992 at beam 400 to 0.9986 at beam 1200, which is usually enough for application scenarios. This indicates that our generator is not validity-bottlenecked by search depth within its intended operational range; increasing beam primarily expands the candidate pool but does not harm chemical correctness.

Evaluation of the Prompted Generation Mechanism of MMPT-FM. We evaluate controllability through a masked infilling task, where MMPT-FM must complete user-specified structural templates with chemically valid variables. We sample 50 variables from each of ChEMBL and PMV17, create three masked templates per variable, and perform prompted generation with 1,000 beams and a length margin of 7. As shown in Table 3, MMPT-FM achieves high ground-truth recovery even under structural constraints: at $K = 1$, recall reaches 58.0% on ChEMBL and 46.0% on PMV17, and by $K = 20$ it approaches saturation on both datasets. At larger K, the model also produces many unique valid candidates, indicating that masked structural prompting enables both controllable and diverse MMPT generation.

Analysis of Distribution Steering via RAG.

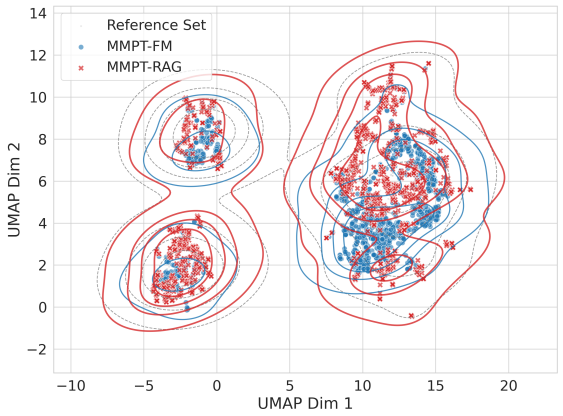


Figure 4: UMAP visualization of MMPT-FM and MMPT-RAG's chemical landscape on PMV17. The grey shaded areas represent the reference dataset's distribution. Compared to FM inference (blue), the MMPT-RAG framework (red) populates structural voids where the foundation model is sparse or absent.

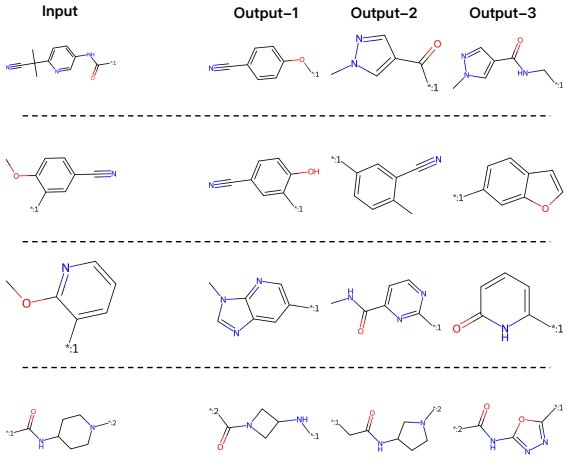


Figure 5: Examples of MMPT-FM generations. In each row, the left structure is the input variable, and the structures on the right are generated outputs.

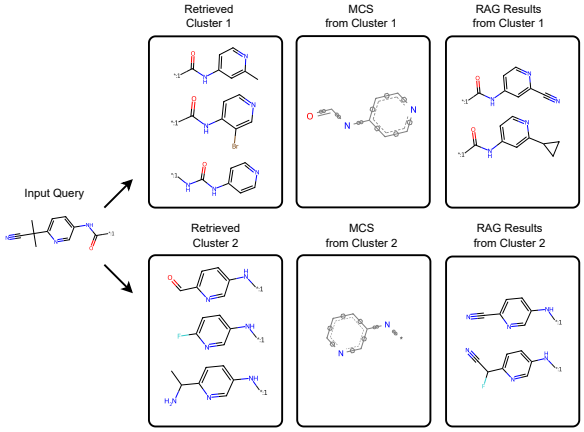


Figure 6: Examples of MMPT-RAG generations. Retrieved variables are clustered, an MCS is extracted per cluster, and generation is conditioned on each cluster's template (MCS).

To investigate how retrieval augmentation steers the generative process toward target chemical domains, we visualize the global distribution of generated analogs against the reference patent dataset (PMV17). We compare the union of outputs from 50 random unique inputs generated via vanilla FM inference versus the MMPT-RAG framework, using the PMV17 dataset as the reference. For clarity, the RAG visualization highlights the additional coverage contributed beyond vanilla FM outputs, illustrating the complementary effect of retrieval augmentation. As illustrated in Fig. 4, MMPT-RAG (red) expands into multiple structural regions that remain underexplored by the vanilla foundation model (blue). The vanilla FM tends to concentrate in high-probability general regions, leaving several reference clusters sparsely covered. In contrast, retrieval augmentation encourages the model to populate these underrepresented areas, effectively filling distributional gaps. This shift indicates that RAG enhances coverage of project-relevant chemical space by guiding generation toward regions of the reference dataset.

Property Evaluation and Extension for Optimization. To assess whether generated variables translate into realistic drug-like molecules, we reconstruct full product molecules and evaluate their synthetic accessibility, drug-likeness, and logP across all three tasks (Appendix D). Additionally, we demonstrate that the framework can be extended for property-conditioned generation through simple methods like post-hoc reranking and fine-tuning (Appendix E).

Hyperparameter Sensitivity. We additionally examine the hyperparameter sensitivity of MMPT-RAG in Appendix C. Results show that moderate increases in all three parameters improve performance with diminishing returns.

5.4 Qualitative Evaluation

To complement the quantitative results, we present representative qualitative examples in Fig. 5 and Fig. 6. Fig. 5 shows that MMPT-FM generates multiple diverse and chemically plausible variants for each input variable, preserving valid valence patterns, realistic functional groups, ring systems, and multiple attachment points, which indicates its ability to handle context-dependent transformations across substitution sites. Fig. 6 further illustrates

MMPT-RAG on the same input as the first example in Fig. 5. Structurally similar historical variables are retrieved and clustered by shared substructures; for each cluster, the extracted Maximum Common Substructure (MCS) serves as a template capturing its dominant transformation pattern. Conditioned on these cluster-specific templates, MMPT-RAG generates outputs that reflect the corresponding cluster patterns while remaining chemically valid and diverse, showing that RAG steers generation toward specific transformation families without sacrificing chemical plausibility.

6 Conclusion

In this work, we reframed analog generation as a transformation-to-transformation task grounded in Matched Molecular Pair Transformations (MMPTs), explicitly modeling the localized edits that encode medicinal chemistry intuition. We trained **MMPT-FM** on large-scale transformation data to generate variable substructures that balance chemical plausibility and structural novelty. We further introduced **MMPT-RAG**, which uses external reference datasets to steer generation toward project-specific motifs. Experiments on general chemical corpora and time-split patent series show improved diversity and validity, as well as recovery of prospective ligands in realistic discovery scenarios. Overall, this work establishes MMPTs as a first-class generative abstraction for combining machine learning with medicinal chemistry expertise.

7 Limitations

Our approach relies on the availability and coverage of large historical transformation datasets, and its performance may vary in underrepresented chemical domains. While we demonstrate that property-conditioned generation is feasible through post-hoc reranking and fine-tuning, direct integration of multi-objective optimization into the MMPT generation process itself remains an important direction for future work. AI assistant was used for Coding and refining this draft.

8 Acknowledgement

This work was supported in part by the National Science Foundation (NSF) under Grant Nos. 2432418, 2414115, 2403312, 2318831, and 2521631; the National Institutes of Health (NIH) under Grant Nos. R01AG089806 and R01CA297856; and a Sony Research Award. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NSF, NIH, or other funding organizations.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Stanislav Andreev, Tatu Pantsar, Ahmed El-Gokha, Francesco Ansideri, Mark Kudolo, Débora Bublitz Anton, Giulia Sita, Jenny Romasco, Christian Geibel, Michael Lämmerhofer, and 1 others. 2020. Discovery and evaluation of enantiopure 9 h-pyrimido [4, 5-b] indoles as nanomolar gsk-3β inhibitors with improved metabolic stability. *International Journal of Molecular Sciences*, 21(21):7823.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and 1 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Binghong Chen, Tianzhe Wang, Chengtao Li, Hanjun Dai, and Le Song. 2021. Molecule optimization by explainable evolution. In *International conference on learning representation (ICLR)*.

Oh-Hyeon Choung, Riccardo Vianello, Marwin Segler, Nikolaus Stiefl, and José Jiménez-Luna. 2023. Extracting medicinal chemistry intuition via preference machine learning. *Nature Communications*, 14(1):6651.

Dimitrios Christofidellis, Giorgio Giannone, Jannis Born, Ole Winther, Teodoro Laino, and Matteo Manica. 2023. Unifying molecular and textual representations via multi-task language modelling. In *International Conference on Machine Learning*, pages 6140–6157. PMLR.

Andrew Dalke, Jerome Hert, and Christian Kramer. 2018. mmpdb: An open-source matched molecular pair platform for large multiproperty data sets. *Journal of chemical information and modeling*, 58(5):902–910.

Inc. Daylight Chemical Information Systems. 2019. Smarts: a language for describing molecular patterns.

Alexander G Dossetter, Edward J Griffen, and Andrew G Leach. 2013. Matched molecular pair analysis in drug discovery. *Drug Discovery Today*, 18(15-16):724–731.

Janos Fischer and C Robin Ganellin. 2010. Analogue-based drug discovery. *Chemistry International–Newsmagazine for IUPAC*, 32(4):12–15.

Anna Gaulton, Louisa J Bellis, A Patricia Bento, Jon Chambers, Mark Davies, Anne Hersey, Yvonne Light, Shaun McGlinchey, David Michalovich, Bissan Al-Lazikani, and 1 others. 2012. Chembl: a large-scale bioactivity database for drug discovery. *Nucleic acids research*, 40(D1):D1100–D1107.

Shiva Ghaemi, Amanda Consylman, Bo Pan, Alice Wu, Ashley Petersen, Gabe Chang, Diana McDonough, Mark Forman, Elise L Bezold, William M Wuest, and 1 others. 2026. Topology-aware generation and activity-based filtering: A computational-experimental framework for data-scarce quaternary ammonium compound discovery. *Journal of Chemical Information and Modeling*, 66(6):3117–3128.

Ed Griffen, Andrew G Leach, Graeme R Robb, and Daniel J Warner. 2011. Matched molecular pairs as a medicinal chemistry tool: miniperspective. *Journal of medicinal chemistry*, 54(22):7739–7750.

Jiazhen He, Eva Nittinger, Christian Tyrchan, Werngard Czechtizky, Atanas Patronov, Esben Jannik Bjerrum, and Ola Engkvist. 2022. Transformer-based molecular optimization beyond matched molecular pairs. *Journal of cheminformatics*, 14(1):18.

Jiazhen He, Huifang You, Emil Sandström, Eva Nittinger, Esben Jannik Bjerrum, Christian Tyrchan, Werngard Czechtizky, and Ola Engkvist. 2021. Molecular optimization by capturing chemist's intuition using deep neural networks. *Journal of cheminformatics*, 13:1–17.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. 2020. Multi-objective molecule generation using interpretable substructures. In *International conference on machine learning*, pages 4849–4859. PMLR.

Wengong Jin, Kevin Yang, Regina Barzilay, and Tommi Jaakkola. 2018. Learning multimodal graph-to-graph translation for molecular optimization. *arXiv preprint arXiv:1812.01070*.

Lou Jost. 2006. Entropy and diversity. *Oikos*, 113(2):363–375.

Greg Landrum. 2013. Rdkit: Open-source cheminformatics. https://www.rdkit.org.

Jaechang Lim, Sang-Yeon Hwang, Seokhyun Moon, Seungsu Kim, and Woo Youn Kim. 2020. Scaffold-based molecular design with a graph generative model. *Chemical science*, 11(4):1153–1164.

Hannes H Loeffler, Jiazhen He, Alessandro Tibo, Jon Paul Janet, Alexey Voronov, Lewis H Mervin, and Ola Engkvist. 2024. Reinvent 4: modern ai–driven generative molecule design. *Journal of Cheminformatics*, 16(1):20.

Jieyu Lu and Yingkai Zhang. 2022. Unified deep learning model for multitask reaction predictions with explanation. *Journal of chemical information and modeling*, 62(6):1376–1387.

Yu A Malkov and Dmitry A Yashunin. 2018. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE transactions on pattern analysis and machine intelligence*, 42(4):824–836.

Leland McInnes, John Healy, Steve Astels, and 1 others. 2017. hdbscan: Hierarchical density based clustering. *J. Open Source Softw.*, 2(11):205.

NA Meanwell. 2017. A synopsis of the properties and applications of heteroaromatic rings in medicinal chemistry. In *Advances in Heterocyclic Chemistry*, volume 123, pages 245–361. Elsevier.

Harry L. Morgan. 1965. 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.

Emmanuel Noutahi, Hadrien Mary, Kyle M. Kovary, Shawn Whitfield, Julien St-Laurent, Honoré Hounwanou, and Michael Craig. 2025. datamol-io/medchem: Molecular filtering for drug discovery (v2.0.5-alpha).

Marcus Olivecrona, Thomas Blaschke, Ola Engkvist, and Hongming Chen. 2017. Molecular de-novo design through deep reinforcement learning. *Journal of cheminformatics*, 9(1):48.

Rıza Ozcelik, Helena Brinkmann, Emanuele Criscuolo, and Francesca Grisoni. 2025. Generative deep learning for de novo drug design - a chemical space odyssey. *Journal of Chemical Information and Modeling*, 65(14):7352–7372.

Bo Pan, Shiva Ghaemi, Amanda J Consylman, Zihao Zhao, Ashley Petersen, Alice Wu, Gabriel Chang, Diana McDonough, Mark A Forman, Elise L Bezold, and 1 others. 2025. An instructible chemist-ai alignment framework for generating quaternary ammonium compound structures. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 4740–4750.

Hao-Wei Pang, Peter Zhiping Zhang, Bo Pan, Liang Zhao, Xiang Yu, and Liying Zhang. 2026. Scalable and generalizable analog design via learning medicinal chemistry intuition from matched molecular pair transformations. *Journal of Chemical Information and Modeling*, 66(15):8908–8922.

George Papadatos, Muhammad Alkarouri, Valerie J Gillet, Peter Willett, Visakan Kadirkamanathan, Christopher N Luscombe, Gianpaolo Bravi, Nicola J Richmond, Stephen D Pickett, Jameed Hussain, and 1 others. 2010. Lead optimization using matched molecular pairs: inclusion of contextual information for enhanced prediction of herg inhibition, solubility, and lipophilicity. *Journal of chemical information and modeling*, 50(10):1872–1886.

Yang Qiao, Bo Pan, Hao-Wei Pang, Peter Zhiping Zhang, Liying Zhang, and Liang Zhao. 2026. Molworld: Molecule world models for actionable molecular optimization. *arXiv preprint arXiv:2605.08954*.

John W Raymond, Eleanor J Gardiner, and Peter Willett. 2002. Rascal: Calculation of graph similarity using maximum common edge subgraphs. *The Computer Journal*, 45(6):631–644.

Alfréd Rényi. 1961. On measures of entropy and information. In *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability, volume 1: contributions to the theory of statistics*, volume 4, pages 547–562. University of California Press.

Valerie W Shurtleff, Mark E Layton, Craig A Parish, James J Perkins, John D Schreier, Yunyi Wang, Gregory C Adam, Nadine Alvarez, Soheila Bahmanjah, Carolyn M Bahnck-Teets, and 1 others. 2024. Invention of mk-7845, a sars-cov-2 3cl protease inhibitor employing a novel difluorinated glutamine mimic. *Journal of Medicinal Chemistry*, 67(5):3935–3958.

Jiangming Sun, Nina Jeliazkova, Vladimir Chupakhin, Jose-Felipe Golib-Dzib, Ola Engkvist, Lars Carlsson, Jörg Wegner, Hugo Ceulemans, Ivan Georgiev, Vedrin Jeliazkov, and 1 others. 2017. Excape-db: an integrated large scale dataset facilitating big data analysis in chemogenomics. *Journal of cheminformatics*, 9(1):17.

Alessandro Tibo, Jiazhen He, Jon Paul Janet, Eva Nittinger, and Ola Engkvist. 2024. Exhaustive local chemical space exploration using a transformer model. *Nature Communications*, 15(1):7315.

Emma P Tysinger, Brajesh K Rai, and Anton V Sinitskiy. 2023. Can we quickly learn to "translate" bioactive molecules with transformer models? *Journal of Chemical Information and Modeling*, 63(6):1734–1744.

Binh Vu, Romyr Dominique, and Hongju Li. 2017. Methods and compounds for restoring mutant p53 function.

Binh Vu, Romyr Dominique, Hongju Li, Bruce Fahr, and Andrew Good. 2021. Methods and compounds for restoring mutant p53 function.

Pat Walters. 2018. rd_filters.

Haorui Wang, Marta Skreta, Cher-Tian Ser, Wenhao Gao, Lingkai Kong, Felix Strieth-Kalthoff, Chenru Duan, Yuchen Zhuang, Yue Yu, Yanqiao Zhu, and 1 others. 2024. Efficient evolutionary search over chemical space with large language models. *arXiv preprint arXiv:2406.16976*.

Ronald J Williams and David Zipser. 1989. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280.

Zhenxing Wu, Odin Zhang, Xiaorui Wang, Li Fu, Huifeng Zhao, Jike Wang, Hongyan Du, Dejun Jiang, Yafeng Deng, Dongsheng Cao, and 1 others. 2024. Leveraging language model for advanced multiproperty molecular optimization via prompt engineering. *Nature Machine Intelligence*, pages 1–11.

Ziyi Yang, Shaohua Shi, Li Fu, Aiping Lu, Tingjun Hou, and Dongsheng Cao. 2023. Matched molecular pair analysis in drug discovery: methods and recent applications. *Journal of Medicinal Chemistry*, 66(7):4361–4377.

Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec. 2018. Graph convolutional policy network for goal-directed molecular graph generation. *Advances in neural information processing systems*, 31.

Botao Yu, Frazier N Baker, Ziqi Chen, Xia Ning, and Huan Sun. 2024. Llasmol: Advancing large language models for chemistry with a large-scale, comprehensive, high-quality instruction tuning dataset. *arXiv preprint arXiv:2402.09391*.

Odin Zhang, Haitao Lin, Hui Zhang, Huifeng Zhao, Yufei Huang, Yuansheng Huang, Dejun Jiang, Chang-Yu Hsieh, Peichen Pan, and Tingjun Hou. 2024. Deep lead optimization: Leveraging generative AI for structural modification. *arXiv preprint arXiv:2404.19230*.

A Implementation Details

Foundation model training and inference. To obtain the foundation model, we leverage the ChemT5 model (Christofidellis et al., 2023), an encoder-decoder transformer pretrained on chemical datasets, as our base model. ChemT5 contains approximately 220 million parameters and has been fine-tuned for tasks in cheminformatics. It consists of 12 layers, 12 attention heads, 220M parameters, and processes input sequences up to 512 tokens (Christofidellis et al., 2023). We employed standard supervised training to fine-tune all parameters of the base model. Teacher forcing (Williams and Zipser, 1989) is incorporated to improve training stability. The training was conducted with a batch size of 64 on each device, and a learning rate of 5e-4. We use an early stop strategy with a tolerance of 2 epochs based on the evaluation loss. Utilizing four NVIDIA A6000 GPUs (48 GB each), the training process required approximately 70 hours to complete. During inference with the foundation model, for each input, we use beam search to produce 1000 outputs with a maximum length of 50.

Retriever. We pre-build a nearest-neighbor index with HNSW (Malkov and Yashunin, 2018) and query it with cosine distance over the Morgan Fingerprint (Morgan, 1965) embedding space. At inference time, we first retrieve at most top 500 nearest input variables, then expand each input into its associated label set. To ensure compatibility with the query, we filter candidates by the number of wild atoms (i.e., attachment points) to match the query variable. We then compute Morgan fingerprints and re-rank retrieved label candidates by Tanimoto similarity between the query and candidate labels, retaining the top set for downstream RAG steps.

Clustering, Template Construction. We cluster the retrieved labels in structure space using a shared-substructure clustering method, where we first compute pairwise similarities between retrieved outputs using the size of the RDKit maximum common substructure (MCS) normalized by the smaller molecule. We then perform agglomerative hierarchical clustering (average linkage) on the corresponding distance matrix, and cut the dendrogram at a similarity threshold of 0.70 to obtain clusters. To avoid overly coarse motifs, any cluster exceeding 10 molecules is recursively split using the same linkage procedure until all clusters satisfy the size constraint. For each retained cluster, we compute a Maximum Common Substructure (MCS), whose resulting SMARTS string serves as the cluster-invariant template.

We further convert this template into a partially specified output constraint by masking atoms outside the invariant scaffold. Concretely, we apply substructure masking to produce a template string where masked spans are denoted by a special masking character. We then convert each masked span into a single `<BLANK>` placeholder and perform span infilling using the generative model, as introduced in the prompted generation section.

Prompted Generation via Mask Infilling. The infilling search is controlled by 11 maximum new tokens per blank, 200 maximum total candidate continuations, and 200 top candidates scored. The final RAG outputs are the union of valid, RDKit-parsable infilled candidates across templates. We exclude any duplicate sequences.

Implementation of Compared Methods. For Retrieval, we first retrieve at most 50 input variables from the reference dataset that are most similar to the test query based on fingerprint similarity. We then collect all corresponding output variables paired with these retrieved inputs. If the resulting candidate set exceeds 1000 outputs, we retain the 1000 variables that are most similar to the test query to ensure a fair comparison in terms of candidate size. REINVENT4 (LibINVENT) follows a different formulation, generating variables conditioned on a scaffold rather than directly modeling MMPT-style variable-to-variable transformations. To align the setting with our task, we use the constant fragment identified by MMPDB as the scaffold input and generate up to 1000 candidate variables for each query.

For consistency, all methods, including MMPT-FM and MMPT-RAG, generate 1000 candidates per input.

B Dataset Statistics

Training data. We use MMPDB (Dalke et al., 2018) to extract MMPs from ChEMBL (Gaulton et al., 2012) with a maximum variable ratio of 33% (`max-variable-ratio = 0.33`). Molecules are first filtered using the `rule_of_druglike_soft` criterion (Noutahi et al., 2025), retaining only compounds with molecular weight $\geq$ 200 Da and removing structural alerts (Walters, 2018). After filtering, 800,714 compounds remain, which yield

361,706 drug-like compounds with valid MMPs. The resulting MMP pairs total approximately 4M, from which we extract 2,638,199 unique variable-to-variable transformations. The vocabulary contains 262,448 unique variables (148,488 unique inputs; 217,882 unique outputs). The ChEMBL held-out test set comprises 2,349 unique input scaffolds and 2,639 ground-truth transformations. We use 90% of these transformations for training and reserve the remaining 10% as the held-out evaluation set.

Patent data. The PMV17 patent dataset contains 62 molecules and 152 R-groups, yielding 16,831 ground-truth transformations within the patent series. For the cross-patent evaluation task (PMV17 $\rightarrow$ PMV21), the target patent (PMV21) contains 286 R-groups with a total of 5,395 ground-truth cross-patent transformations.

C Hyperparameter Sensitivity Analysis

To better understand the behavior of MMPT-RAG, we perform a sensitivity analysis on three hyperparameters that control generation quality: the number of retrieved clusters expanded, i.e., generate using its MCS (default 10), the number of variables generated per cluster (default 50), and the mask-infilling length range used during sequence completion (default = original length before masking ± 7). We vary one hyperparameter at a time while keeping the others fixed: clusters $3, 5, 10, 20$, variables per cluster $10, 25, 50, 100$, and mask range $[1, 3], [1, 5], [1, 7], [1, 9]$. Figure 7 shows that moderate increases in all three parameters improve performance, while larger values provide diminishing returns. Expanding more clusters improves results by exposing more diverse retrieved patterns and then stabilizes around 10 to 20 clusters. Increasing the number of variables per cluster expands search coverage but saturates near 50 samples. Widening the mask-infilling range consistently helps up to $[1, 7]$, after which additional flexibility yields little gain. Based on these trends, we recommend 10 clusters, 50 variables per cluster, and a mask range of $[1, 7]$ to balance performance and computation.

D Property Evaluation

To evaluate whether the generated variables translate into realistic full molecules, we reconstruct full product molecules by applying the generated variable to the constant scaffold and compute three widely-used molecular property metrics: synthetic accessibility (SA, lower is easier to synthesize), quantitative estimate of drug-likeness (QED, higher is more drug-like), and octanol-water partition coefficient (logP). We compare properties of reconstructed molecules against the ground-truth molecules from the dataset across all three tasks.

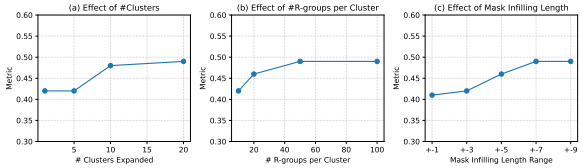


Figure 7: Hyperparameter Study. (a) the number of clusters to expand, (b) the number of variables to generate for each cluster, (c) the range of mask length to fill.

As shown in Table 4, both MMPT-FM and MMPT-RAG produce molecules with synthetic accessibility, drug-likeness, and logP values close to those of ground-truth molecules across all three tasks. The property distributions overlap substantially with ground truth, indicating that the generated transformations preserve favorable physicochemical profiles. In particular, SA scores indicate high synthetic feasibility, and QED values confirm drug-like character. These results demonstrate that the proposed framework generates not only structurally valid and novel MMPTs but also practically relevant molecules suitable for downstream medicinal chemistry consideration.

E Property Optimization via Post-hoc Reranking and Fine-Tuning

While the core MMPT framework is designed as a structure-level generator, it can be extended to support property-conditioned generation through two lightweight mechanisms: post-hoc reranking and fine-tuning. We evaluate these on a solubility-annotated subset of ChEMBL comprising 500 held-out molecular pairs with associated aqueous solubility measurements. For each input variable, we generate 100 candidates using (1) the base MMPT-FM and (2) a solubility-finetuned variant trained on approximately 266K MMPTs derived from solubility-increasing transformations. Full product molecules are reconstructed, and aqueous solubility is predicted using a RandomForest regressor. We report the mean predicted solubility change ($\Delta \log_{10}$ molarity) across inputs.

As shown in Table 5, the base MMPT-FM without property guidance produces a near-zero average solubility change across all candidates. Sim-

Table 4: Property evaluation of reconstructed full molecules.

Method	Task 1			Task 2			Task 3		
	SA↓	QED↑	logP	SA↓	QED↑	logP	SA↓	QED↑	logP
MMPT-FM	2.48±0.58	0.63±0.18	3.50±1.29	3.83±0.45	0.31±0.13	4.52±1.30	3.37±0.26	0.35±0.10	5.10±0.97
MMPT-RAG	2.99±0.60	0.59±0.20	3.35±1.36	3.95±0.42	0.32±0.15	4.07±1.35	3.76±0.36	0.30±0.10	4.79±1.00
GT	2.28±0.49	0.71±0.14	3.02±1.15	3.20±0.29	0.41±0.11	4.62±0.82	3.25±0.22	0.41±0.07	4.79±0.68

Table 5: Property-conditioned generation results. Average predicted solubility change ($\Delta\log_{10}$ molarity) under different selection strategies.

Selection	MMPT-FM	Sol-Finetuned
All candidates (avg)	-0.11	$+0.55$
Top-1 (rerank)	$+0.78$	$+1.55$
Top-5 (rerank)	$+0.58$	$+1.39$
Top-10 (rerank)	$+0.47$	$+1.31$
Top-50 (rerank)	$+0.04$	$+0.98$

ple post-hoc reranking by the property predictor yields substantial improvements, with Top-1 reaching $+0.78$ log units. Fine-tuning on solubility-increasing MMPTs further elevates the baseline to $+0.55$ and Top-1 to $+1.55$, demonstrating that the model can learn property-guided transformation patterns directly from data. These results confirm that our framework supports lightweight property optimization while maintaining its core structure-generation capabilities.

F Proof of Theorem 4.1

Proof. According to the workflow defined in Eq. 5, the MMPT-RAG framework constructs the global generation as a mixture of cluster-conditioned distributions with weights $\tilde{\pi}_k$. Summing over the variables y:

$$\begin{aligned} p_{\text{RAG}}(y \mid x) &= \sum_{k=1}^{K} \tilde{\pi}_k p(y \mid x, T_k) \\ &= \sum_{k=1}^{K} \tilde{\pi}_k \big[(1-\alpha_k) p_\theta(y \mid x) + \\ &\quad \alpha_k p(y \mid T_k)\big] \\ &= p_\theta(y \mid x) \sum_{k=1}^{K} \tilde{\pi}_k (1-\alpha_k) + \\ &\quad \sum_{k=1}^{K} \tilde{\pi}_k \alpha_k p(y \mid T_k). \end{aligned}$$

Using $\sum \tilde{\pi}_k = 1$ and the definition $\bar{\alpha} = \sum \tilde{\pi}_k \alpha_k$, the first term simplifies to $(1-\bar{\alpha}) p_\theta(y \mid x)$. The second term, by multiplying and dividing by $\bar{\alpha}$, recovers the effective reference $p^*_{\text{ref}}(y \mid x)$. Thus:

$$p_{\text{RAG}}(y \mid x) = (1-\bar{\alpha}) p_\theta(y \mid x) + \bar{\alpha} p^*_{\text{ref}}(y \mid x).$$

□