Molexar: A Unified Multimodal Molecular Foundation Model for Drug Design

Haoyu Lin [*1] **Yiyan Liao** [*2] **Jinmei Pan** [3] **Xinliao Ling** [1] **Luhua Lai** [1 4 5 6] **Jianfeng Pei** [1]

Abstract

Molecular generation is a central challenge in drug discovery, requiring models that explore vast chemical space while satisfying diverse design constraints. We present Molexar, a unified multimodal molecular foundation model built on Fragment-SELFIES, a robust, fragment-aware molecular language with validity-preserving decoding and explicit fragment structure. A pretrained autoregressive decoder learns the Fragment-SELFIES syntax and molecular distribution; supervised fine-tuning (SFT) then trains the same decoder on condition-molecule pairs spanning scalar molecular properties, pharmacophore fingerprints, protein sequences, and binding pockets, injecting each condition by in-place replacement of value-token embeddings so that all generation modes share one autoregressive path. Molexar achieves strong efficiency at a small parameter count while matching or exceeding larger models. The pretrained model reaches 100% validity and high drug-likeness in unconditional and fragment-constrained generation; the SFT model follows single- and multi-property instructions and remains competitive on target-conditioned generation on the CrossDocked2020 test set. On MolGenBench, Molexar further generates molecules with favorable safety and potency. These results establish Molexar as a practical unified foundation for computational chemistry and drug-design workflows.

[*]Equal contribution [1]Center for Quantitative Biology, Academy for Advanced Interdisciplinary Studies, Peking University, Beijing 100871, China [2]School of Life Sciences, Peking University, Beijing 100871, China [3]Department of Pathology, The First Affiliated Hospital of Shantou University Medical College, Shantou, Guangdong 515041, China [4]BNLMS, College of Chemistry and Molecular Engineering, Peking University, Beijing 100871, China [5]Peking-Tsinghua Center for Life Sciences, Academy for Advanced Interdisciplinary Studies, Peking University, Beijing 100871, China [6]Peking University Chengdu Academy for Advanced Interdisciplinary Biotechnologies, Chengdu, Sichuan 610213, China. Correspondence to: Luhua Lai <lhlai@pku.edu.cn>, Jianfeng Pei <jfpei@pku.edu.cn>.

Preprint. June 25, 2026.

1. Introduction

Generative modeling has become a central component of modern drug discovery because it promises to search chemical spaces that are far larger than experimentally accessible libraries. Early molecular language models and variational or adversarial generators demonstrated that strings and graphs can be optimized toward desired molecular properties, including GPT-style SMILES generation with property and scaffold conditioning (Bagal et al., 2022). Practical molecular design, however, rarely involves a single objective. A useful generator must simultaneously produce valid molecules, preserve or modify fragments, control scalar properties, satisfy pharmacophore hypotheses, and exploit target information from protein sequences or three-dimensional binding pockets. These requirements have motivated a broad range of specialized models, including chemical language models for property-directed design, structure-based generators for protein pockets, pharmacophore-conditioned models, and fragment-aware representations.

Molecular representation is a recurring bottleneck. SMILES is compact and widely used, but not every token sequence maps to a valid molecule (Weininger, 1988). SELFIES addresses this by defining a robust string representation in which arbitrary SELFIES strings decode to valid molecules (Krenn et al., 2020; 2022). More recent fragment-aware representations, including Group SELFIES, SAFE, and t-SMILES, explicitly encode chemically meaningful substructures to improve generation, optimization, and fragment-constrained design (Cheng et al., 2023; Noutahi et al., 2024; Wu et al., 2024a). Fragment-level representations are particularly attractive for medicinal chemistry because scaffold decoration, linker design, and hit-to-lead optimization are often expressed as operations on fragments rather than isolated atoms.

At the same time, model architectures have diverged across design settings. Molecular foundation models such as ChemFM, GenMol, and SmileyLlama explore the use of large language or diffusion models for broad molecular generation tasks (Cai et al., 2026; Lee et al., 2025; Cavanagh et al., 2026). Protein- or target-aware models such as TamGen introduce target conditioning into chemical language models (Wu et al., 2024b). In structure-based drug design,

receptor-conditioned generators have evolved from direct 3D ligand construction methods such as DeepLigBuilder to grid-based models, autoregressive equivariant models, flow-based pocket generators, and diffusion models conditioned on protein pockets (Li et al., 2021; Ragoza et al., 2022; Luo et al., 2021; Liu et al., 2022; Peng et al., 2022; Zhang et al., 2023; Jiang et al., 2024; Guan et al., 2023a; Schneuing et al., 2024; Huang et al., 2024a; Guan et al., 2023b; Huang et al., 2024b; Lin et al., 2025; Qu et al., 2024). PocketXMol recently showed that an atom-level prompt and universal denoising framework can unify many pocket-interaction tasks, including docking, structure-based design, fragment operations, and peptide design (Peng et al., 2026). Other methods focus on pharmacophores, interactions, shapes, or electrostatics as design priors (Imrie et al., 2021; Skalic et al., 2019; Zhu et al., 2023; Xie et al., 2025; Peng et al., 2025; Zhung et al., 2024; Adams et al., 2025; Wang & Rajapakse, 2024). While these systems have advanced individual tasks, their heterogeneity makes it difficult to share representations, transfer pretraining, or compare conditioning strategies across modalities.

Evaluation has likewise become more demanding. General molecular generation benchmarks such as GuacaMol and MOSES emphasize distribution learning, validity, novelty, and goal-directed optimization (Brown et al., 2019; Polykovskiy et al., 2020). Structure-based evaluation depends on curated protein-ligand resources such as PDBbind, CASF, and CrossDocked2020 (Wang et al., 2004; Liu et al., 2015; Li et al., 2018; Su et al., 2019; Francoeur et al., 2020), while virtual-screening benchmarks such as DUD-E and DEKOIS 2.0 test enrichment against challenging decoys (Mysinger et al., 2012; Bauer et al., 2013). Recent PoseBusters and PoseCheck analyses show that docking or RMSD scores alone can hide chemically invalid or strained generated poses (Buttenschoen et al., 2024; Harris et al., 2023). Newer resources including Durian, POKMOL-3D, PLINDER, SAIR, and MolGenBench further emphasize leakage-aware splits, realistic protein-ligand systems, active recovery, and hit-to-lead applicability (Nie et al., 2025; Liu et al., 2024; Durairaj et al., 2024; Lemos et al., 2025; Cao et al., 2025). These trends motivate a model interface that can support multiple evaluation regimes without rebuilding the generator for each modality.

We present Molexar, a unified multimodal molecular foundation model designed to reuse one autoregressive molecular decoder across these settings. Molexar is based on the Gemma2 architecture (Gemma Team, 2024); however, its molecular language, condition interface, and training objective are specialized for molecular design. The model operates on Fragment-SELFIES, a BRICS-fragment molecular representation developed for this work that uses an explicit fragment-tree token grammar rather than corpus-specific fragment identifiers. Fragment-SELFIES encodes the BRICS fragment tree with a small set of structural tokens, represents fragment interiors using SELFIES-compatible tokens, and exposes dummy attachment sites that support fragment-constrained prompts. This design retains chemically meaningful fragment structure while avoiding opaque fragment IDs. The complete codec and model workflow are summarized in Figures 1 and 2.

The main design goal of Molexar is to make conditional molecular generation a prompt-embedding problem rather than an architecture-specific problem. All conditions are inserted into a fixed template as token-like values. During the forward pass, embeddings at selected value-token positions are replaced by projections of condition values. Discrete properties are encoded by one-hot embeddings, continuous properties by radial basis expansions, vector conditions by learned projections, and pocket graphs by a GVP encoder (Jing et al., 2021). This strategy allows unconditional pretraining, conditional supervised fine-tuning, and fragment continuation to share the same causal language modeling path while preserving key-value-cache-compatible generation.

This paper makes the following contributions. First, we introduce Fragment-SELFIES, a fragment-level molecular representation with an explicit fragment-tree token grammar that avoids corpus-specific fragment-ID tokens while supporting foundation-model pretraining and fragment-constrained generation. Second, we describe Molexar, a Gemma2-derived molecular language model that unifies unconditional and multimodal conditional generation through value-token embedding replacement. Third, we evaluate the same model across unconditional generation, inference efficiency, fragment-constrained design, property control, target-conditioned generation on CrossDocked2020 test set, and real-world applicability on MolGenBench, comparing against molecular language models, pharmacophore-guided generators, and structure-based drug design systems.

2. Related Work

2.1. Molecular String and Fragment Representations

SMILES remains the dominant textual molecular representation due to its compactness and software support (Weininger, 1988). Its fragility under arbitrary sequence perturbations, however, can make language-model decoding inefficient when validity is important. SELFIES was introduced to address this limitation by making molecular strings robust under arbitrary token sequences (Krenn et al., 2020; 2022). Group SELFIES extends the same philosophy with group-level tokens, allowing larger chemical units to be represented explicitly (Cheng et al., 2023). SAFE uses fragment strings connected by attachment labels and demonstrates strong performance in fragment-constrained generation and

molecular design (Noutahi et al., 2024). t-SMILES further explores tree-like fragment decomposition for ligand design (Wu et al., 2024a). Fragment-SELFIES follows this fragment-aware direction by explicitly serializing a BRICS fragment tree while encoding each fragment body with SELFIES-compatible tokens and non-atomic dummy attachment placeholders. This preserves chemically meaningful fragment structure for constrained prompts and avoids the fixed predefined group-token set required by Group SELFIES without representing fragments as opaque IDs.

2.2. Molecular Foundation and Language Models

MolGPT showed that a decoder-only Transformer trained by next-token prediction over SMILES can generate valid molecules and can be conditioned on scalar properties and Bemis-Murcko scaffolds (Bagal et al., 2022). More recent work has increasingly treated molecules as a domain for larger foundation models. ChemFM studies scaling-law-guided chemical pretraining over molecular strings and supports property-related downstream tasks (Cai et al., 2026). GenMol formulates a drug-discovery generalist through discrete diffusion over molecular sequences (Lee et al., 2025). SmileyLlama adapts large language models for directed chemical space exploration with natural-language-like property prompts (Cavanagh et al., 2026). MolFM aligns molecular graphs, biomedical text, and knowledge-graph context for multimodal molecular representation learning (Luo et al., 2023). TamGen combines a protein encoder and chemical decoder to generate molecules conditioned on target information (Wu et al., 2024b). These models show that molecular generation and representation learning can benefit from general sequence-modeling and multimodal pretraining machinery, but they often specialize their conditioning interface to a particular input form. Molexar instead uses a uniform value-token interface that can bind scalar properties, fingerprints, protein embeddings, and pocket embeddings to the same molecular decoder.

2.3. Structure-Based and Feature-Guided Generation

Structure-based drug design requires models to generate molecules that are chemically valid and compatible with a target binding site. Early deep generative approaches included DeepLigBuilder, which combines an end-to-end 3D ligand generator with Monte Carlo tree search to optimize molecules directly inside binding pockets, and grid- or density-based receptor-conditioned models (Li et al., 2021; Ragoza et al., 2022). Subsequent models used equivariant or geometric networks for autoregressive atom placement, as in SBDD, GraphBP, Pocket2Mol, ResGen, and PocketFlow (Luo et al., 2021; Liu et al., 2022; Peng et al., 2022; Zhang et al., 2023; Jiang et al., 2024). Diffusion-based models such as TargetDiff, DiffSBDD, PMDM, DecompDiff, IRDiff, and DiffBP generate full molecular structures conditioned on pocket geometry, decomposed priors, or reference-ligand interactions and have become important baselines for 3D SBDD (Guan et al., 2023a; Schneuing et al., 2024; Huang et al., 2024a; Guan et al., 2023b; Huang et al., 2024b; Lin et al., 2025). PocketXMol broadens this geometric line by representing tasks with atom-level prompts and training a universal denoiser across small-molecule and peptide tasks, yielding a unified 3D model for pocket-based generation and docking (Peng et al., 2026). MolCRAFT emphasizes conformational quality and highlights that high docking scores can be misleading when generated poses are strained or unrealistic (Qu et al., 2024). In parallel, DEVELOP, LigDream, PGMG, TransPharmer, PhoreGen, DeepICL, ShEPhERD, and PharmacoBridge demonstrate the value of pharmacophore, interaction, shape, electrostatic, and linker constraints (Imrie et al., 2021; Skalic et al., 2019; Zhu et al., 2023; Xie et al., 2025; Peng et al., 2025; Zhung et al., 2024; Adams et al., 2025; Wang & Rajapakse, 2024). Molexar is complementary to these methods: it is a molecular language model, but its condition interface is designed to absorb the same categories of target and feature information.

3. Methods

3.1. Fragment-SELFIES Molecular Language

Fragment-SELFIES represents a molecule as a BRICS fragment tree (Figure 1). A root fragment begins with `[Frag]`. An explicit connection is encoded by `[Attach:M]` followed by `[Frag@N]`: `[Attach:M]` selects the parent-side attachment site, and `[Frag@N]` starts the child fragment while selecting its child-side attachment site. This explicitly specifies the bonding relationship between the two fragments; `[pop]` then returns traversal to the parent. The body of each fragment is encoded by SELFIES-compatible tokens such as atom, branch, and ring tokens. Attachment sites are represented by non-atomic `[Dummy]` tokens, which are removed or reconnected during decoding. A `[SELFIES]`...`[ENDSELFIES]` fallback can encode molecules for which the BRICS-fragment form is not used.

This representation is designed for molecular language modeling. First, it does not rely on a corpus-specific fragment-ID token set at encode/decode time: it combines a small fixed set of structural tokens with SELFIES-compatible fragment-body tokens. Second, BRICS fragments expose chemically meaningful units, making fragment-level prompts natural. Third, adjacent root fragments can form an implicit connection, written schematically as `[Frag] A [Frag] B`. Unlike an explicit connection of the form `[Attach:M] [Frag@N]`, this form does not prescribe which attachment sites should bond. It is therefore designed for linker-design start strings: the user can provide two endpoint fragments as the initial string, and generation can continue by propos-

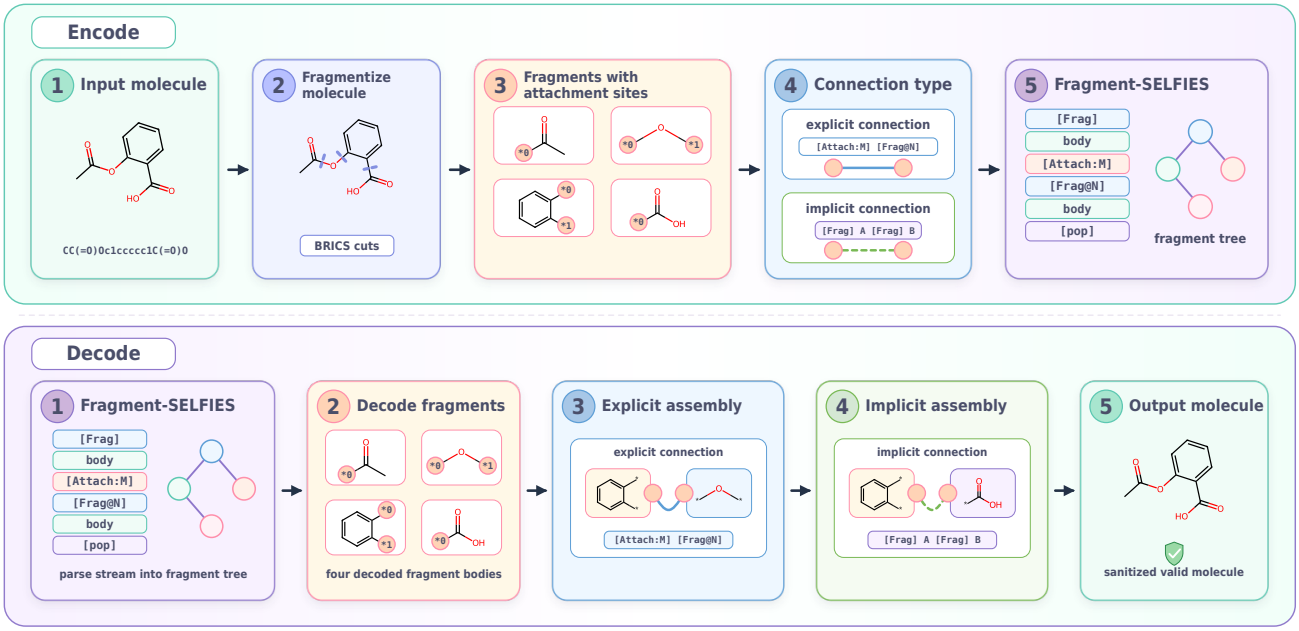


Figure 1. **Fragment-SELFIES encoding and decoding workflow.** A molecule or SMILES string is BRICS-fragmented into chemically meaningful fragments with dummy attachment sites. The encoder serializes the resulting fragment tree with root `[Frag]` tokens, parent-side `[Attach:M]` selectors, child-side `[Frag@N]` tokens, and `[pop]` returns while keeping each fragment body in SELFIES-compatible tokens. The connection-type panel distinguishes explicit connections, where `[Attach:M]` and `[Frag@N]` define the parent and child attachment sites, from implicit connections, schematically `[Frag] A [Frag] B`, where adjacent root fragments do not specify the bonding relationship. During decoding, explicit edges are reconnected directly, whereas implicit anchors are assembled through compatible unused dummy attachment sites, making the implicit form useful as a linker-design start string. The fallback `[SELFIES]...[ENDSELFIES]` block preserves molecules that are not represented by the BRICS-fragment form.

ing the missing linker fragments while decoding assembles compatible unused dummy attachment sites.

As a concrete example matching Figure 1, aspirin, `CC(=O)Oc1ccccc1C(=O)O`, is decomposed into four BRICS fragments: an acetyl fragment, an ester oxygen, an aryl ring, and a carboxyl fragment. One equivalent Fragment-SELFIES serialization is shown below, with line breaks added only for readability:

```
[Frag][C][CH0][=Branch1][C][=O][Dummy][Attach:0]
[Frag@0][Dummy][OH0][Dummy][Attach:1]
[Frag@0][Dummy][CH0][=C][C][=C][C]
[=CH0][Ring1][=Branch1][Dummy][pop][pop]
[Frag][O][=CH0][Branch1][C][O][Dummy]
```

The first three fragments are connected explicitly: acetyl attachment 0 connects to oxygen attachment 0, and oxygen attachment 1 connects to aryl attachment 0. The final carboxyl fragment starts with a new `[Frag]` root rather than an `[Attach:M] [Frag@N]` edge, so the aryl-carboxyl relationship is implicit. During decoding, the parser first reconstructs the explicit acetyl-oxygen-aryl tree, then assembles the adjacent carboxyl root through the remaining compatible dummy attachment sites. This string was checked with the Fragment-SELFIES implementation and strict decoding recovers the canonical aspirin SMILES `CC(=O)Oc1ccccc1C(=O)O`.

3.2. Molexar Sequence Template

Molexar uses a shared sequence template for pretraining, supervised fine-tuning, and inference:

```
<BOS><COND> conditions </COND><SEP>
<MOL> molecule </MOL><EOS>.
```

The condition block contains condition keys and placeholder value tokens. The molecular block contains a Fragment-SELFIES string. In unconditional pretraining, no condition value is active and the model learns the molecular grammar and distribution. In conditional fine-tuning and inference, selected placeholder value-token embeddings are replaced with encoded condition vectors, while all surrounding tokens and the autoregressive objective remain unchanged.

In the implementation, the condition block is an ordered sequence of key-token/value-token pairs. With zero-indexed token positions in the full template, the value placeholder for the ith condition slot is

$$p_i = 3 + 2i, \qquad i = 0, \ldots, 21, \tag{1}$$

where the first twelve slots correspond to molecular properties, pharmacophore fingerprint, protein-sequence embedding, and pocket-geometry embedding, and the remaining ten slots are reserved unused slots. Thus changing the ac-

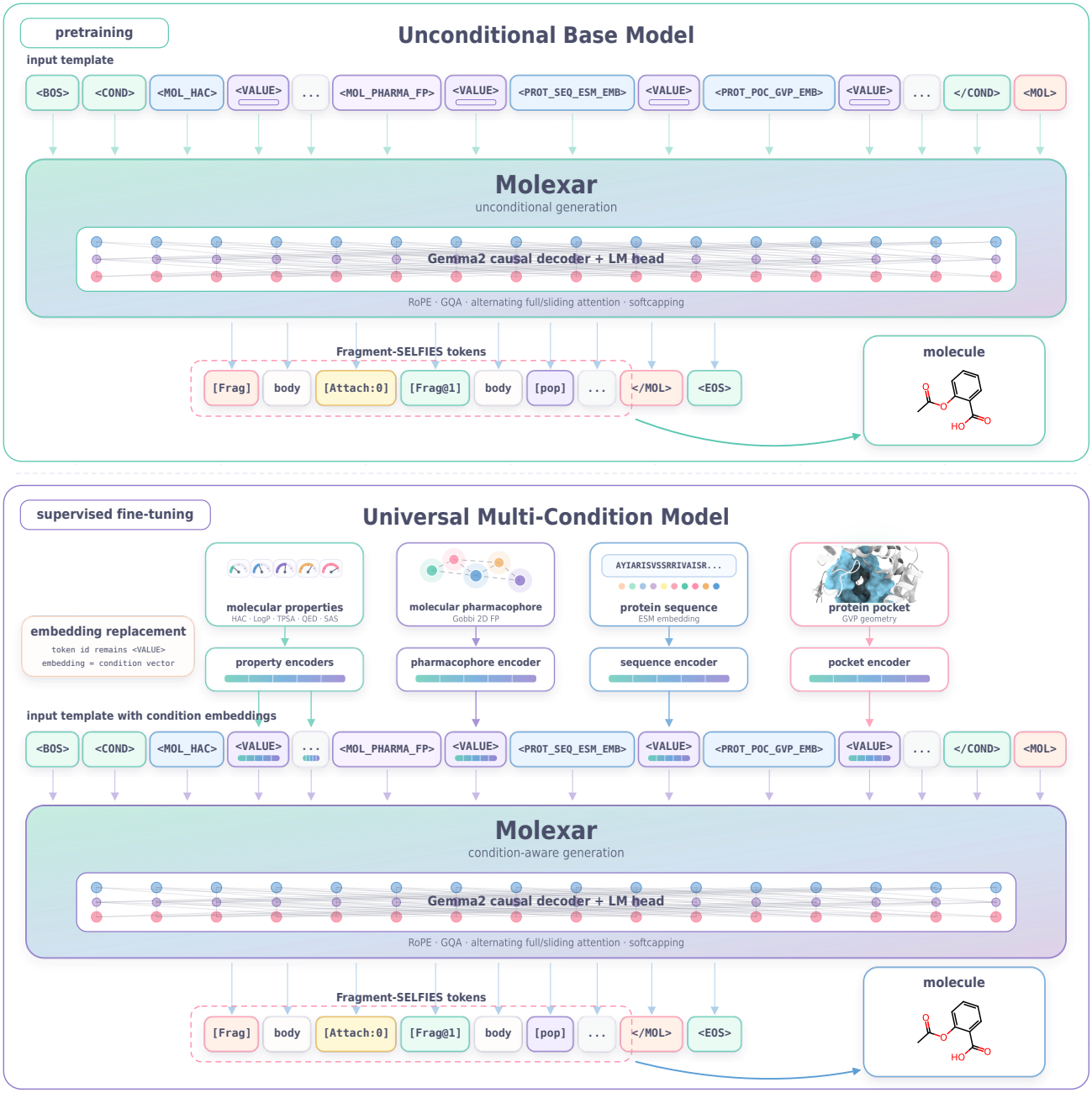


Figure 2. **Molexar architecture and training flow.** Stage 1 performs unconditional causal-language-model pretraining on Fragment-SELFIES strings with the unified prompt template and a Gemma2-style molecular decoder. Stage 2 performs universal multi-condition supervised fine-tuning: scalar property encoders, pharmacophore-fingerprint projections, protein-sequence embeddings, and pocket-geometry encoders produce vectors that replace designated <VALUE> token embeddings in the condition block. The autoregressive decoder, molecular output block, attention masks, and key-value-cache-compatible generation path are shared across unconditional generation, property- and pharmacophore-guided generation, target-aware generation, pocket-conditioning generation, and fragment continuation.

tive condition set changes only which fixed positions are replaced, not the token sequence length.

3.3. Backbone Architecture

The decoder is a Gemma2-style causal transformer with rotary positional embeddings, grouped-query attention, sliding-window/full-attention layer patterns, and logit soft-capping inherited from the Gemma2 design (Gemma Team,

2024). As shown in Figure 2, we train it as a molecular causal language model over Fragment-SELFIES tokens and reuse the same decoder for unconditional and conditional generation. The experimental model has 16 transformer layers, hidden size 256, intermediate size 640, four attention heads, one key-value head, maximum sequence length 256, sliding-window size 128, and a 127-token vocabulary. The tokenizer is a word-level tokenizer whose regular expression treats molecular and control tokens as atomic units. This keeps chemical tokens intact and the molecular token set interpretable.

Let $h_t^{(\ell)}$ denote the hidden state at position t in layer ℓ. For query head a and its grouped key-value head $g(a)$, attention uses the allowed causal set $\mathcal{A}_\ell(t)$, which is either all positions up to t in full-attention layers or the most recent 128 positions up to t in sliding-window layers. With $\tau_{\text{attn}} = 50$,

$$\lambda_{ts}^{(\ell,a)} = \tau_{\text{attn}} \tanh\left(\frac{(q_t^{(\ell,a)})^\top k_s^{(\ell,g(a))}}{\tau_{\text{attn}}\sqrt{d_h}}\right), \tag{2}$$

$$\alpha_{ts}^{(\ell,a)} = \text{softmax}_{s\in\mathcal{A}_\ell(t)}\left(\lambda_{ts}^{(\ell,a)}\right). \tag{3}$$

The attention output is $\sum_{s\in\mathcal{A}_\ell(t)} \alpha_{ts}^{(\ell,a)} v_s^{(\ell,g(a))}$, followed by the Gemma2 feed-forward, normalization, and residual updates.

3.4. Unified Condition Injection

Let $x_{0:T-1}$ be the tokenized template and let $E(x_t)$ be the learned input embedding at position t. For an active condition c_i with a designated value-token position p_i, Molexar first maps the raw condition to a feature vector

$$r_i(c_i) = \begin{cases} \text{onehot}_{\Omega_i}(c_i), & \text{discrete scalar,} \\ [\psi_{ij}(c_i)]_{j=1}^{K_i}, & \text{continuous scalar,} \\ c_i, & \text{direct vector condition,} \end{cases} \tag{4}$$

where Ω_i is the discrete allowable set, $\psi_{ij}(c_i) = \exp(-(c_i - \mu_{ij})^2/(2\sigma_i^2))$, and the RBF centers μ_{ij} are uniformly spaced over the configured property range. A two-layer projector then produces a hidden-size condition embedding

$$z_i = P_i(r_i(c_i)) = W_{i,2}\,\phi(W_{i,1} r_i(c_i) + b_{i,1}) + b_{i,2} \in \mathbb{R}^d, \tag{5}$$

with $d = 256$ and the GELU-tanh activation ϕ. The input embedding sequence is modified by

$$\tilde{e}_t = \begin{cases} z_i, & t = p_i \text{ for an active condition } i, \\ E(x_t), & \text{otherwise.} \end{cases} \tag{6}$$

The transformer receives $\tilde{e}_{0:T-1}$ instead of the original embeddings. Missing conditions are assigned out-of-range positions by the collator and therefore do not replace any embedding. This operation does not add tokens, does not alter attention masks, and does not require cross-attention. Therefore, generation remains compatible with the standard autoregressive key-value cache.

Molexar currently supports twelve built-in condition types. Discrete scalar conditions include heavy atom count (HAC), hydrogen-bond donor count (HBDC), hydrogen-bond acceptor count (HBAC), and rotatable bond count (RotBC). Continuous scalar conditions include molecular weight (WT, Da), LogP, topological polar surface area (TPSA), QED, and synthetic accessibility score (SAS), encoded through radial basis functions before projection. Vector conditions include a 2D pharmacophore fingerprint, a protein sequence embedding, and a pocket-geometry embedding. The protein-sequence condition uses mean-pooled ESMC-600M final embeddings with dimension 1152 (Candido et al., 2026). Additional unused condition slots are reserved for user-defined conditions.

3.5. Pocket Geometry Encoder

For pocket-conditioned generation, a protein pocket is converted to an atom graph. We parse no-hydrogen pocket PDB files, keep atoms within a 25 Å radius of the pocket center, cap each pocket at 425 atoms, and build a directed 8-nearest-neighbor graph. For centered atom coordinates $\rho_i \in \mathbb{R}^3$, node features are

$$\begin{aligned} s_i &= [\text{onehot}(a_i); 0] \in \mathbb{R}^{11}, \\ V_i &= [\rho_i, \rho_i/\|\rho_i\|_2, \mathbf{0}] \in \mathbb{R}^{3\times 3}, \end{aligned} \tag{7}$$

where a_i is the atom element. For each directed edge (i, j), edge features are the unit direction $u_{ij} = (\rho_j - \rho_i)/\|\rho_j - \rho_i\|_2$ and a 32-dimensional scalar vector consisting of 16 Gaussian distance features with centers on $[0, 8]$ and width 0.5, concatenated with a fixed edge-type indicator. A three-layer geometric vector perceptron encoder maps the graph to node states, and the pocket vector is their mean pooled scalar output,

$$g_{\text{pocket}} = \frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} W_{\text{out}}\left(h_i^{(3)}\right) \in \mathbb{R}^{256}. \tag{8}$$

This vector is projected by the pocket-condition projector and injected at the pocket-condition value token, keeping the molecular decoder unchanged while allowing the model to use three-dimensional pocket context.

3.6. Training Data

The base pretraining and molecule-context SFT corpus is derived from UniChem (Chambers et al., 2013). We canonicalize molecules as non-isomeric RDKit SMILES with explicit hydrogens removed and stereochemistry stripped, deduplicate them, and filter out malformed records, mixtures or reactions, RDKit sanitization failures, molecules

with more than 50 heavy atoms, radicals, molecular weight above 800 Da, QED below 0.3 (Bickerton et al., 2012), SAS above 5 (Ertl & Schuffenhauer, 2009), and PAINS (Baell & Holloway, 2010), Brenk (Brenk et al., 2008), or NIH (Jadhav et al., 2010; Doveston et al., 2015) structural alerts. The final UniChem-derived molecule list, with SAIR and PLINDER training-set ligand coverage included, is converted offline into randomized Fragment-SELFIES strings. The full molecule-context file contains 135,763,524 Fragment-SELFIES records, corresponding to 33,940,881 molecule-condition rows with four precomputed randomized folds. The aligned condition files contain the nine scalar molecular properties and a 2D pharmacophore fingerprint.

Target-context SFT uses protein-ligand pairs from SAIR and the PLINDER training set (Lemos et al., 2025; Durairaj et al., 2024). Ligands are normalized to no-hydrogen, no-stereochemistry SMILES, filtered and deduplicated, and converted to 10-fold randomized Fragment-SELFIES files. Protein-sequence context is stored as precomputed ESMC-600M embeddings, and pocket context is read from processed no-hydrogen pocket structures. To control leakage in CrossDocked2020 evaluation (Francoeur et al., 2020), we removed SAIR and PLINDER training-set pairs whose protein sequence had more than 30% identity to any CrossDocked2020 test protein, using MMseqs2 sequence search (Steinegger & Söding, 2017; Kallenborn et al., 2025). After this filtering, the target-context pool contains 573,463 SAIR pair records and 21,770 PLINDER training-set pair records.

3.7. Training Objectives

Molexar uses a two-stage recipe. In the first stage, the base model is pretrained on randomized Fragment-SELFIES strings using causal language modeling. The template includes the condition block, but no condition values are injected and the condition encoders and GVP pocket encoder are not updated by this objective. This stage teaches the decoder the Fragment-SELFIES syntax, fragment-tree traversal patterns, and molecular distribution. In the second stage, universal multi-condition SFT starts from the pretrained checkpoint and trains the decoder, condition encoders, and pocket encoder on condition-molecule pairs.

For both stages, the model predicts the next token from the shifted logits. If $h_t = F_\theta(\tilde{e}_{0:t})$ is the decoder state, the vocabulary logits use the implementation's final softcap

$$o_t = \tau_{\text{out}} \tanh\left(\frac{W_{\text{lm}} h_t}{\tau_{\text{out}}}\right), \qquad \tau_{\text{out}} = 30. \tag{9}$$

The training loss is

$$\mathcal{L}(\theta) = -\frac{1}{\sum_{b,t} m_{bt}} \sum_b \sum_{t=1}^{T-1} m_{bt} \log \text{softmax}(o_{b,t-1})_{x_{bt}}, \tag{10}$$

where $m_{bt} = 1$ when the label at position t is not ignored. Stage 1 uses the full shifted template labels. Stage 2 sets $m_{bt} = 0$ for the prefix through <MOL> and for padding tokens, so the loss is applied to the molecular continuation and closing tokens conditioned on the prompt and injected values.

Universal SFT mixes molecule-context and target-context samples in a 4:1 ratio. For molecule-context samples, conditions are drawn from the nine scalar properties and the 2D pharmacophore fingerprint. The default schedule samples one, two, or three active molecule-side conditions with probabilities 0.6, 0.3, and 0.1, respectively, and oversamples the pharmacophore-fingerprint condition with probability 0.5. Thus molecule-context SFT covers up to 175 single-, pair-, and triple-condition subsets; the property-only subsets account for 129 of these combinations. For target-context samples, sequence and pocket conditions are sampled first: when both are available, the dual sequence-plus-pocket condition is selected with probability 0.2 and otherwise one of the two target modalities is sampled. Ligand-side property or pharmacophore conditions can then be added with probability 0.2, capped at three active condition values in total. Both stages are trained with maximum sequence length 256, learning rate 2×10^{-4}, 2,000 warmup steps, batch size 1,000, bfloat16 mixed precision, and full-shard FSDP on eight H800 GPUs; Stage 1 uses two epochs and Stage 2 uses five epochs.

4. Experiments

4.1. Unconditional and Fragment-Constrained Generation

4.1.1. Unconditional Generation

We first evaluate unconditional generation, which probes the ability of the base model to produce valid molecules that span a broad region of chemical space. We sampled 10,000 molecules from our pretrained base model Molexar and assessed them with four metrics: validity, uniqueness, diversity, and quality. Validity is the fraction of generated strings that are parseable by RDKit; uniqueness is the fraction of distinct molecules among the valid ones; and diversity is the mean pairwise Tanimoto distance over ECFP4 fingerprints of the valid molecules. Quality is the fraction of molecules that are simultaneously drug-like and synthesizable, defined as drug-likeness QED ≥ 0.6 (Bickerton et al., 2012) and synthetic accessibility SAS ≤ 4 (Ertl & Schuffenhauer, 2009), following Jin et al. (2020) and Lee et al. (2025). We compare against SAFE-GPT (Noutahi et al., 2024), GenMol (Lee et al., 2025), and ChemFM-1B and ChemFM-3B (Cai et al., 2026), all of which are language models trained on molecular string representations.

As shown in Table 1, Molexar attains the best overall pro-

Table 1. **Unconditional molecule generation results on 10,000 samples. Bold**: best; underline: second best.

Method	Validity	Uniqueness	Diversity	Quality
SAFE-GPT	0.9351	**1.0000**	0.8783	0.5465
GenMol	0.9997	0.8132	0.8324	0.7428
ChemFM-1B	0.9939	**1.0000**	**0.9001**	0.4157
ChemFM-3B	0.9945	**1.0000**	**0.9001**	0.4090
Molexar	**1.0000**	0.9997	0.8824	**0.8326**

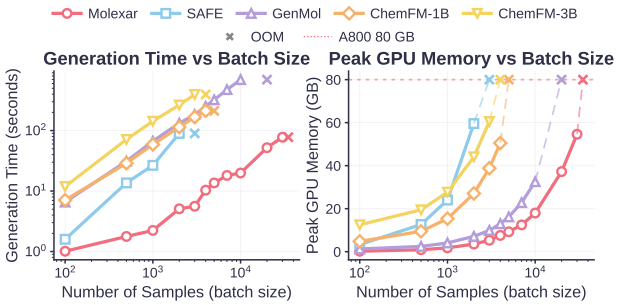


Figure 3. **Inference efficiency under unconditional generation.** Generation time (left) and peak GPU memory (right) versus batch size. Crosses (×): first OOM batch size; dotted line: 80 GB capacity.

file, combining the highest quality with strong diversity and near-perfect uniqueness. By construction, Molexar decodes generated Fragment-SELFIES strings through a validity-oriented codec with RDKit sanitization and non-strict recovery paths, yielding 100% validity in our samples. This validity profile holds across all subsequent experiments and illustrates a concrete advantage of Fragment-SELFIES over SAFE (Noutahi et al., 2024) and SMILES (Weininger, 1988). The agreement between the sampled molecules and the training distribution is examined in Section A. Example generated molecules are shown in Figure 13.

We further benchmark inference latency and peak GPU memory for all methods on a single NVIDIA A800-SXM4-80GB GPU in FP32 (Figure 3). Molexar attains the lowest latency and memory footprint across all batch sizes, scaling to far larger batches before saturating the GPU. It samples up to 30,000 molecules within a single batch in under 80 s, whereas the baselines typically run out of memory once the batch size exceeds 10,000. This efficiency follows from Molexar's small parameter count and the Gemma2 backbone (Gemma Team, 2024).

4.1.2. Fragment-Constrained Generation

Fragment-constrained generation grows new fragments onto one or more user-specified input fragments. Using the benchmark of Noutahi et al. (2024), which provides ten known drugs together with their fragment decompositions, we evaluate five tasks: linker design, scaffold morphing, motif extension, scaffold decoration, and superstructure generation. We characterize the generated molecules with validity, uniqueness, diversity, and quality, as well as distance, the average Tanimoto distance between the original and the generated molecules. For each drug and each task we generated 100 molecules and compared against SAFE-GPT (Noutahi et al., 2024) and GenMol (Lee et al., 2025). Note that linker design and scaffold morphing use the same generation procedure for both Molexar and GenMol.

As shown in Table 2, Molexar outperforms both baselines in validity on every task and in quality on most tasks, indicating that it produces more drug-like molecules and is well suited to realistic drug-design settings. Examples of Molexar generations for Eliglustat are shown in Figure 14.

Table 2. **Fragment-constrained molecule generation results.** The results are the means and the standard deviations of three runs. The best results are highlighted in **bold**.

Method	Task	Validity (%)	Uniqueness (%)	Quality (%)	Diversity	Distance
SAFE-GPT	Linker design	46.7 ± 1.7	**72.5 ± 8.7**	15.6 ± 1.7	0.497 ± 0.024	0.509 ± 0.033
	Scaffold morphing	49.2 ± 2.4	**76.5 ± 1.1**	13.7 ± 1.4	0.513 ± 0.005	0.540 ± 0.002
	Motif extension	91.5 ± 6.4	69.9 ± 3.1	19.7 ± 3.3	0.542 ± 0.010	0.670 ± 0.006
	Scaffold decoration	81.6 ± 6.9	79.0 ± 3.2	9.4 ± 2.7	0.576 ± 0.008	0.617 ± 0.005
	Superstructure generation	97.4 ± 0.4	86.1 ± 3.7	18.0 ± 3.9	0.573 ± 0.021	**0.785 ± 0.026**
GenMol	Linker design	93.9 ± 2.9	71.7 ± 1.1	13.9 ± 0.4	**0.567 ± 0.001**	**0.555 ± 0.002**
	Scaffold morphing	93.9 ± 2.9	71.7 ± 1.1	13.9 ± 0.4	**0.567 ± 0.001**	**0.555 ± 0.002**
	Motif extension	82.2 ± 0.1	57.6 ± 1.9	16.5 ± 0.5	**0.630 ± 0.002**	**0.677 ± 0.003**
	Scaffold decoration	98.0 ± 0.2	86.1 ± 1.1	**32.5 ± 1.1**	**0.598 ± 0.002**	**0.626 ± 0.002**
	Superstructure generation	98.2 ± 0.5	**87.6 ± 4.1**	29.5 ± 2.8	**0.613 ± 0.024**	0.757 ± 0.015
Molexar	Linker design	**100.0 ± 0.0**	55.1 ± 0.3	**25.2 ± 1.7**	0.536 ± 0.004	0.529 ± 0.002
	Scaffold morphing	**100.0 ± 0.0**	55.1 ± 0.3	**25.2 ± 1.7**	0.536 ± 0.004	0.529 ± 0.002
	Motif extension	**100.0 ± 0.0**	**79.3 ± 1.8**	**46.8 ± 1.6**	0.621 ± 0.004	0.652 ± 0.000
	Scaffold decoration	**100.0 ± 0.0**	**96.3 ± 0.5**	16.6 ± 0.4	0.580 ± 0.005	0.601 ± 0.003
	Superstructure generation	**100.0 ± 0.0**	70.5 ± 5.8	**31.9 ± 0.3**	0.487 ± 0.016	0.746 ± 0.016

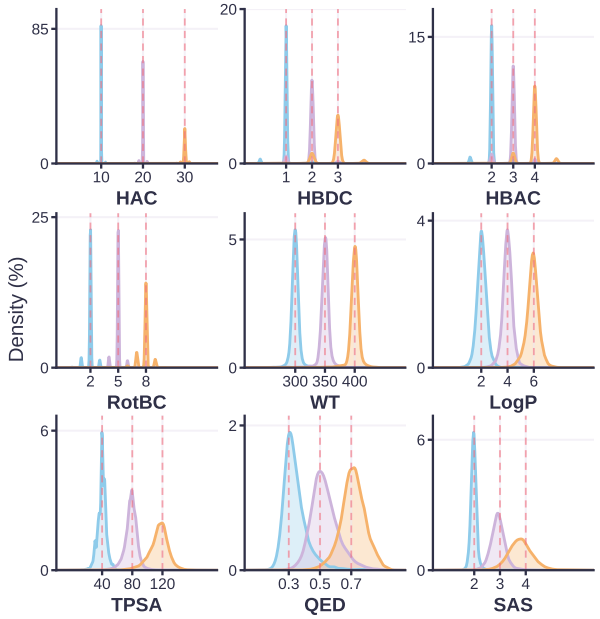


Figure 4. **Single-property conditioning across nine properties.** Each panel shows the property distribution of molecules generated under three target values (dashed red lines); from left to right and top to bottom: HAC, HBDC, HBAC, RotBC (discrete), WT, LogP, TPSA, QED, SAS (continuous).

4.2. Property-Controlled Generation

Conditioning in prior molecular generators is often restricted: some methods require additional, goal-specific reinforcement learning to steer generation toward a target property (Noutahi et al., 2024), while others condition on only a small number of continuous properties (Cai et al., 2026). Through supervised fine-tuning (SFT) against a universal objective, Molexar instead conditions generation on nine continuous or discrete properties and supports single-, dual-, and triple-property prompts. We note that the model is explicitly trained on these combinations: the SFT schedule samples all 129 property-only subsets (9 singletons, 36 pairs, and 84 triples).

4.2.1. Single-Property Conditioning

For each of the nine properties we selected three target values and generated 10,000 molecules per target, as shown in Figure 4. Molexar exhibits strong conditioning ability for both discrete and continuous properties: the generated molecules form a single sharp peak around each target value, with the mode of the distribution coinciding with the requested target, demonstrating reliable single-property instruction following.

We note that the degree of single-property obedience is influenced by the abundance of the corresponding property values in the training set; Section B reports the relationship between the obedience rate and training-set abundance.

4.2.2. Multi-Property Conditioning

Molexar also supports conditioning on combinations of properties. This is important because many drug-design objectives are defined by several properties jointly rather than in isolation, as in the rule of three for fragment-based discovery, Lipinski's rule of five, and the beyond-rule-of-five (bRo5) regime for oral drugs (Congreve et al., 2003; Lipinski et al., 2001; Doak et al., 2014). Because the number of possible combinations is large, we evaluate three representative scenarios, shown in Figure 5.

The first scenario fixes HAC and varies RotBC (Figure 5, left), testing whether the model can produce molecules of different flexibility at a constant size. The generated molecules follow the requested targets closely, and they shift visibly from fused-ring scaffolds toward longer flexible chains, indicating that Molexar captures the joint distribution rather than each property in isolation.

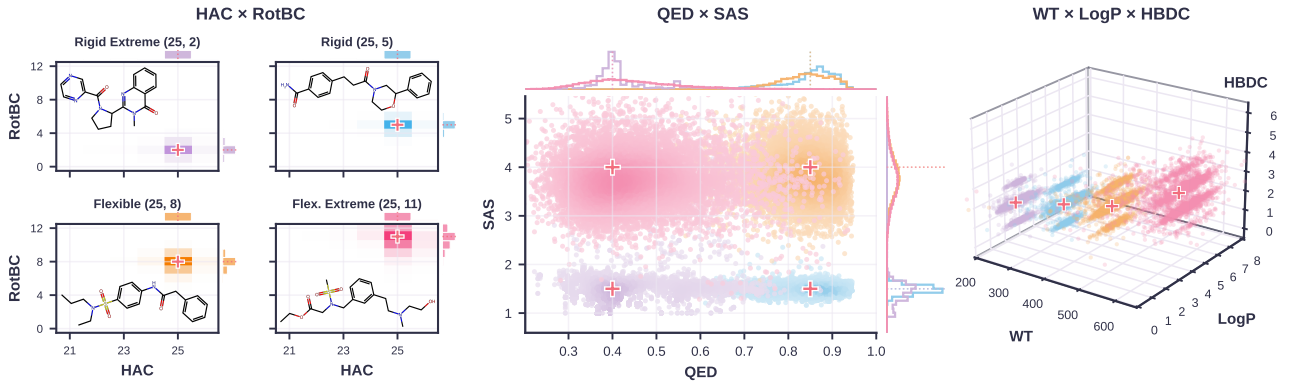


Figure 5. **Multi-property conditioning under realistic design scenarios. Left:** size-flexibility decoupling, jointly conditioning on {HAC, RotBC} at rigid-extreme {25, 2}, rigid {25, 5}, flexible {25, 8}, and flexible-extreme {25, 11}. **Middle:** QED-SAS trade-off, conditioning on {QED, SAS} at {0.85, 1.5}, {0.85, 4.0}, {0.40, 1.5}, and {0.40, 4.0}. **Right:** drug-development regimes defined by {WT, LogP, HBDC}: fragment-like {250, 1.5, 2}, lead-like {350, 2.5, 2}, drug-like {450, 3.5, 2}, and beyond-rule-of-five (bRo5) {600, 4.5, 3}.

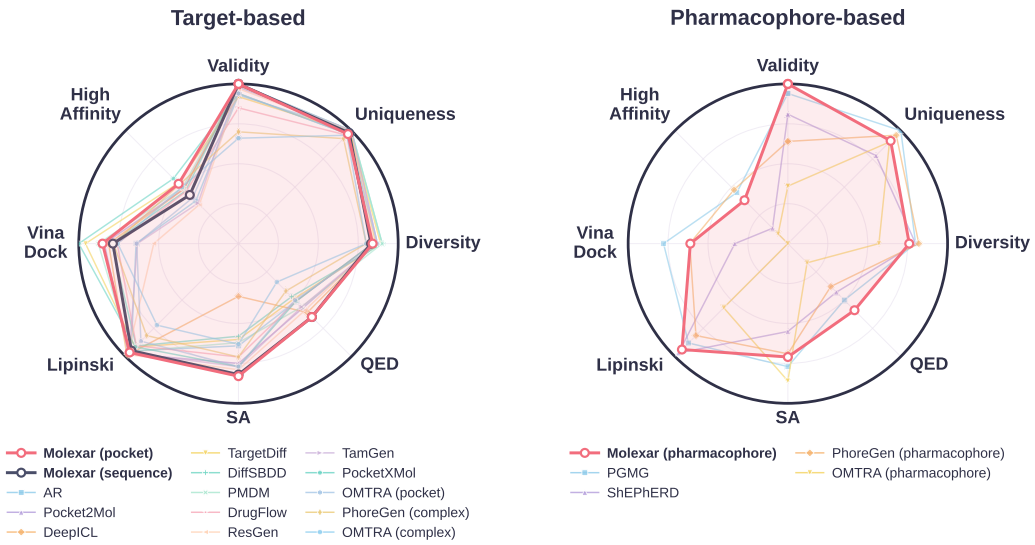


Figure 6. **Target-conditioned molecular generation on the CrossDocked2020 test set. Left:** generation metrics for methods conditioned on target structure or sequence (some methods additionally take a reference-ligand pharmacophore as input). **Right:** generation conditioned only on the reference-ligand pharmacophore. For Molexar, we report molecules generated under each of three conditioning modes: protein sequence, binding pocket, and ligand pharmacophore.

The second scenario combines QED and SAS (Figure 5, middle) to test whether the model can produce molecules that are both drug-like and easy to synthesize. We sampled the four corners of the QED-SAS plane, including settings in which one or both targets are placed at unfavorable values. The model follows all four settings, and obedience is highest in the most useful corner, namely high QED together with low SAS. This suggests that the training corpus is well matched to drug-like chemistry.

The third scenario uses three properties, WT, LogP, and HBDC, to trace a rule-of-five drug-likeness trajectory, simulating the shift in chemical space along a drug-discovery pipeline from fragment-like to bRo5 regimes (Figure 5, right). Molexar tracks the requested WT, LogP, and HBDC closely across the four regimes, and the generated molecules respect the rule-of-three, rule-of-five, and bRo5 criteria that define them, demonstrating its drug-generation capability. The bRo5 regime is matched slightly less tightly than the others, consistent with a training corpus filtered to drug-like molecules (see Methods). This experiment is a proof of concept that Molexar can populate molecular libraries spanning different stages of drug development. The representative molecules are shown in Figure 15.

We note that the conditioned properties are not independent. Correlations among them mean that some combinations correspond to molecules that are scarce or absent in the training set, which can reduce obedience for those settings. We report the pairwise property correlations of the training set in Section C for reference.

4.3. Target-Conditioned Generation

To assess whether Molexar is target-aware, we evaluate it on the CrossDocked2020 test set (Francoeur et al., 2020). The baselines fall into three groups: pocket-only methods (AR (Luo et al., 2021), Pocket2Mol (Peng et al., 2022), DeepICL (Zhung et al., 2024), TargetDiff (Guan et al., 2023a), DiffSBDD (Schneuing et al., 2024), PMDM (Huang et al., 2024a), DrugFlow (Schneuing et al., 2025), ResGen (Zhang et al., 2023), TamGen (Wu et al., 2024b), PocketXMol (Peng et al., 2026), and OMTRA (Dunn et al., 2025) (pocket)); pharmacophore-only methods that condition on a reference-ligand pharmacophore (PGMG (Zhu et al., 2023), ShEPhERD (Adams et al., 2025), PhoreGen (Peng et al., 2025) (pharmacophore), and OMTRA (pharmacophore)); and methods that condition jointly on the pocket and the pharmacophore (PhoreGen (complex) and OMTRA (complex)). Some methods support more than one of these settings (PhoreGen and OMTRA). Owing to its flexible conditioning interface, Molexar is evaluated under three settings: pocket geometry (Molexar (pocket)), protein sequence (Molexar (sequence)), and reference-ligand pharmacophore (Molexar (pharmacophore)). To prevent data leakage, we removed training samples with high sequence homology ($> 30\%$) to the CrossDocked2020 test set. For each of the 100 complexes in the test set we generated 100 molecules and report validity, uniqueness, diversity, QED, SA (rescaled from the original 1–10 scale to 0–1, where higher is better), Lipinski's rule of five, the Vina docking score (Trott & Olson, 2010), and the fraction of molecules whose Vina docking score is better than that of the reference ligand (high affinity).

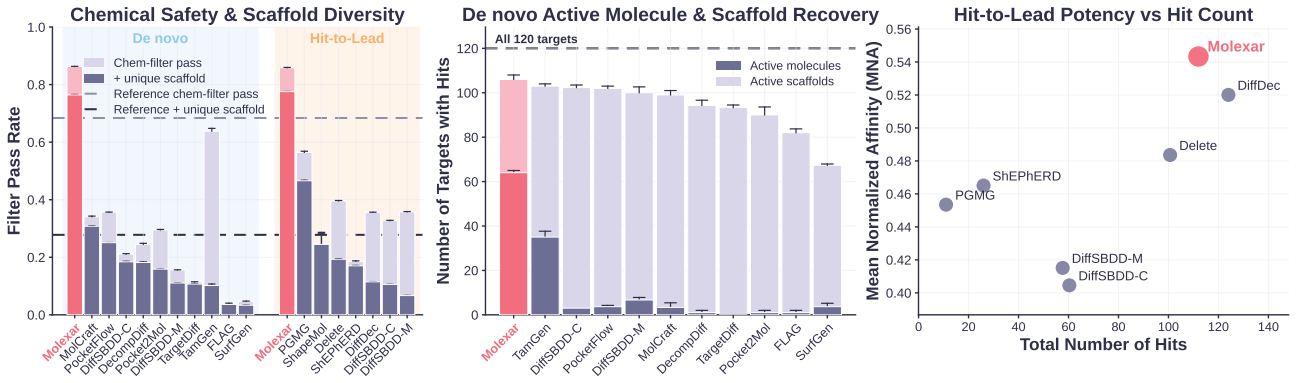


Figure 7. **Real-world applicability results on MolGenBench. Left:** fraction of generated molecules that pass the chemical filters (dark) and the fraction of molecules that also contribute a unique Bemis-Murcko scaffold (Bemis & Murcko, 1996) (light). **Middle:** evaluation of molecular generative models on identifying active molecules and scaffolds through *de novo* generation. **Right:** evaluation of hit-to-lead (H2L) optimization, measured by the potency of generated active molecules and the number of hits.

As shown in Figure 6, Molexar achieves a well-balanced profile across all metrics. Under every conditioning setting, the molecules generated by Molexar rank among the best in SA and QED, indicating favorable drug-likeness. Although it is a language model, Molexar (pocket) attains better Vina docking scores than many structure-based models and clearly outperforms the language-model baseline TamGen, ranking third in high-affinity ratio among the target-based methods. Despite the strict homology-based deduplication, Molexar (sequence) still achieves strong Vina scores, reflecting the out-of-distribution generalization afforded by multi-condition SFT. For pharmacophore conditioning, Molexar performs comparably to existing models. Detailed per-metric results on CrossDocked2020 are reported in Tables 4 and 5. Overall, Molexar demonstrates a balanced and strong ability to generate potentially high-affinity molecules for specific targets.

4.4. Real-World Applicability on MolGenBench

To evaluate the practical applicability of Molexar, we test it on MolGenBench (Cao et al., 2025). The benchmark comprises two tasks, *de novo* generation and hit-to-lead (H2L) optimization, and assesses both generation across 120 protein targets and the recovery of ground-truth drug molecules. We follow its protocol strictly: for *de novo* generation we condition on the pocket, and for the H2L task we condition jointly on the pocket and the reference-ligand pharmacophore.

We focus on the practical drug-likeness of the generated molecules. Viable drugs must be sufficiently safe and free of unstable or reactive groups, and MolGenBench applies a panel of industry-standard chemical filters (Schuffenhauer et al., 2020; Bruns & Watson, 2012); under these filters, only about 70% of the reference active molecules and about 30% of their scaffolds pass. As shown in Figure 7 (left), Molexar reaches a molecule pass rate of about 85% and a scaffold pass rate of about 75%, indicating high safety and diversity. For the *de novo* task (Figure 7, middle), Molexar exactly recovers the active-molecule SMILES for more than half of the targets and nearly all of the active scaffolds. For the H2L task (Figure 7, right), Molexar discovers the second-largest number of active molecules while achieving the highest potency, demonstrating its ability to generate both a greater quantity of active molecules and compounds with superior bioactivity. Additionally, results on general molecular properties are reported in Figure 12, where Molexar ranks among the top methods. These results indicate that Molexar is an effective model for generating safe, high-affinity molecules in realistic drug-discovery settings.

5. Conclusions

In this work, we present Molexar, a unified molecular foundation model that treats diverse design constraints as values in a single autoregressive prompt. Molexar is built on two components: Fragment-SELFIES, an explicit BRICS-fragment molecular language with validity-preserving decoding and fragment continuation, and a value-token embedding replacement mechanism that binds scalar properties, pharmacophore fingerprints, protein-sequence embeddings, and pocket-geometry embeddings to a single Gemma2-style decoder. This removes the need for task-specific generators and cross-attention, letting unconditional sampling, property control, pharmacophore guidance, target and pocket conditioning, and fragment-constrained generation share one sequence template and a cache-compatible decoding path.

This unified interface comes at no cost to generation quality. In unconditional generation, Molexar attains 100% validity,

near-perfect uniqueness, high diversity, and the best quality score among molecular-language baselines, all at the lowest latency and memory use; it preserves perfect validity under fragment constraints and follows single-, dual-, and triple-property instructions from fragment-like to bRo5 regimes. Conditioned on protein sequence, pocket geometry, or pharmacophores on CrossDocked2020, it generates chemically favorable molecules with competitive docking metrics under leakage-controlled training, and on MolGenBench it combines high chemical-filter pass rates with strong active-molecule and scaffold recovery and H2L potency. Together, these results establish Molexar as a compact, efficient foundation model for multimodal molecular design, and future work should test prospective, experimentally validated design cycles.

Impact Statement

This paper presents Molexar, a unified molecular foundation model whose goal is to advance molecule generation. By handling diverse design conditions within a single compact and fast model that produces drug-like, target-aware candidates, Molexar lowers the cost of exploring chemical space and holds broad potential for drug discovery.

Code Availability

The source code for Molexar is publicly available at https://github.com/fairydance/Molexar, and the implementation of Fragment-SELFIES is available at https://github.com/fairydance/Fragment-SELFIES. Additional information and resources for both projects are available at https://molexar.com.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (grant nos. 22033001 and T2321001), the National Key R&D Program of China (grant no. 2023YFF1205103), the Chinese Academy of Medical Sciences (grant no. 2021-I2M-5-014), and Anhui's Plans for Major Provincial Science & Technology Projects (grant no. 202303a07020009). We thank the team managing the high-performance computing platform at the Peking-Tsinghua Center for Life Sciences, Peking University, for providing computational resources.

References

Adams, K., Abeywardane, K., Fromer, J., and Coley, C. W. ShEPhERD: Diffusing shape, electrostatics, and pharmacophores for bioisosteric drug design. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=KSLkFYHlYg.

Baell, J. B. and Holloway, G. A. New substructure filters for removal of pan assay interference compounds (PAINS) from screening libraries and for their exclusion in bioassays. *Journal of Medicinal Chemistry*, 53 (7):2719–2740, 2010. doi: 10.1021/jm901137j. URL https://doi.org/10.1021/jm901137j.

Bagal, V., Aggarwal, R., Vinod, P. K., and Priyakumar, U. D. MolGPT: Molecular generation using a transformer-decoder model. *Journal of Chemical Information and Modeling*, 62(9):2064–2076, 2022. doi: 10.1021/acs.jcim.1c00600. URL https://doi.org/10.1021/acs.jcim.1c00600.

Bauer, M. R., Ibrahim, T. M., Vogel, S. M., and Boeckler, F. M. Evaluation and optimization of virtual screening workflows with DEKOIS 2.0: A public library of challenging docking benchmark sets. *Journal of Chemical Information and Modeling*, 53(6):1447–1462, 2013. doi: 10.1021/ci400115b. URL https://doi.org/10.1021/ci400115b.

Bemis, G. W. and Murcko, M. A. The properties of known drugs. 1. molecular frameworks. *Journal of Medicinal Chemistry*, 39(15):2887–2893, 1996. doi: 10.1021/jm9602928. URL https://doi.org/10.1021/jm9602928.

Bickerton, G. R., Paolini, G. V., Besnard, J., Muresan, S., and Hopkins, A. L. Quantifying the chemical beauty of drugs. *Nature Chemistry*, 4(2):90–98, 2012. doi: 10.1038/nchem.1243. URL https://doi.org/10.1038/nchem.1243.

Brenk, R., Schipani, A., James, D., Krasowski, A., Gilbert, I. H., Frearson, J., and Wyatt, P. G. Lessons learnt from assembling screening libraries for drug discovery for neglected diseases. *ChemMedChem*, 3(3):435–444, 2008. doi: 10.1002/cmdc.200700139. URL https://doi.org/10.1002/cmdc.200700139.

Brown, N., Fiscato, M., Segler, M. H. S., and Vaucher, A. C. GuacaMol: Benchmarking models for de novo molecular design. *Journal of Chemical Information and Modeling*, 59(3):1096–1108, 2019. doi: 10.1021/acs.jcim.8b00839. URL https://doi.org/10.1021/acs.jcim.8b00839.

Bruns, R. F. and Watson, I. A. Rules for identifying potentially reactive or promiscuous compounds. *Journal of Medicinal Chemistry*, 55(22):9763–9772, 2012. doi: 10.1021/jm301008n. URL https://doi.org/10.1021/jm301008n.

Buttenschoen, M., Morris, G. M., and Deane, C. M. PoseBusters: AI-based docking methods fail to generate physically valid poses or generalise to novel sequences. *Chemical Science*, 15(9):3130–3139, 2024. doi: 10.1039/D3SC04185A. URL https://doi.org/10.1039/D3SC04185A.

Cai, F., Zacour, K., Zhu, T., Tzeng, T.-R., Duan, Y., Liu, L., Pilla, S., Li, G., and Luo, F. ChemFM as a scaling law guided foundation model pre-trained on informative chemicals. *Communications Chemistry*, 9(1):3, 2026. doi: 10.1038/s42004-025-01793-8. URL https://doi.org/10.1038/s42004-025-01793-8.

Candido, S., Hayes, T., Derry, A., Rao, R., Lin, Z., Verkuil, R., Wu, B. Z., Lee, J. S., Bruguera, E. S., Keval, J. A., Kopylov, M., Pak, J. E., Wu, W., Thomas, N., Mataraso, S., Hsu, A., Trotman-Grant, A. C., Fatras, K., dos Santos Costa, A., Badkundri, R., Akın, H., Oktay, D., Deaton, J., Montabana, E., Sitwala, H., Yu, Y., Wiggert, M., Carlin, D. A., Goering, A. W., Blazejewski, T., Sandora, M., Hla, M., Jia, T. Z., Kloker, L. H., Sofroniew, N. J., Uehara, M., Pannu, J., Bachas, S., Liu, D. S., Sercu, T., and Rives, A. Language modeling materializes a world model of protein biology, 2026. URL https://doi.org/10.64898/2026.06.03.729735. bioRxiv preprint.

Cao, D., Fan, Z., Yu, J., Chen, M., Jiang, X., Sheng, X., Wang, X., Zeng, C., Luo, X., Teng, D., and Zheng, M. Benchmarking real-world applicability of molecular generative models from de novo design to lead optimization with MolGenBench, 2025. URL https://doi.org/10.1101/2025.11.03.686215. bioRxiv preprint.

Cavanagh, J. M., Sun, K., Gritsevskiy, A., Bagni, D., Wang, Y., Bannister, T. D., and Head-Gordon, T. SmileyLlama: Modifying large language models for directed chemical space exploration. *Nature Computational Science*, 2026. doi: 10.1038/s43588-026-00986-y. URL https://doi.org/10.1038/s43588-026-00986-y. Published online 11 May 2026.

Chambers, J., Davies, M., Gaulton, A., Hersey, A., Velankar, S., Petryszak, R., Hastings, J., Bellis, L., McGlinchey, S., and Overington, J. P. UniChem: A unified chemical structure cross-referencing and identifier tracking system. *Journal of Cheminformatics*, 5(1):3, 2013. doi: 10.1186/1758-2946-5-3. URL https://doi.org/10.1186/1758-2946-5-3.

Cheng, A. H., Cai, A., Miret, S., Malkomes, G., Phielipp, M., and Aspuru-Guzik, A. Group SELFIES: A robust fragment-based molecular string representation. *Digital Discovery*, 2(3):748–758, 2023. doi: 10.1039/D3DD00012E. URL https://doi.org/10.1039/D3DD00012E.

Congreve, M., Carr, R., Murray, C., and Jhoti, H. A 'rule of three' for fragment-based lead discovery? *Drug Discovery Today*, 8(19):876–877, 2003. doi: 10.1016/S1359-6446(03)02831-9. URL https://doi.org/10.1016/S1359-6446(03)02831-9.

Doak, B. C., Over, B., Giordanetto, F., and Kihlberg, J. Oral druggable space beyond the rule of 5: Insights from drugs and clinical candidates. *Chemistry & Biology*, 21(9):1115–1142, 2014. doi: 10.1016/j.chembiol.2014.08.013. URL https://doi.org/10.1016/j.chembiol.2014.08.013.

Doveston, R. G., Tosatti, P., Dow, M., Foley, D. J., Li, H. Y., Campbell, A. J., House, D., Churcher, I., Marsden, S. P., and Nelson, A. A unified lead-oriented synthesis of over fifty molecular scaffolds. *Organic & Biomolecular Chemistry*, 13(3):859–865, 2015. doi: 10.1039/C4OB02287D. URL https://doi.org/10.1039/C4OB02287D.

Dunn, I., Toft, L., Katz, T., Gupta, J., Shah, R., Hettiarachchi, R., and Koes, D. R. OMTRA: A multitask generative model for structure-based drug design, 2025. URL https://arxiv.org/abs/2512.05080. arXiv preprint arXiv:2512.05080.

Durairaj, J., Adeshina, Y., Cao, Z., Zhang, X., Oleinikovas, V., Duignan, T., McClure, Z., Robin, X., Studer, G., Kovtun, D., Rossi, E., Zhou, G., Veccham, S., Isert, C., Peng, Y., Sundareson, P., Akdel, M., Corso, G., Stärk, H., Tauriello, G., Carpenter, Z., Bronstein, M., Kucukbenli, E., Schwede, T., and Naef, L. PLINDER: The protein-ligand interactions dataset and evaluation resource, 2024. URL https://doi.org/10.1101/2024.07.17.603955. bioRxiv preprint.

Ertl, P. and Schuffenhauer, A. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of Cheminformatics*, 1(1):8, 2009. doi: 10.1186/1758-2946-1-8. URL https://doi.org/10.1186/1758-2946-1-8.

Francoeur, P. G., Masuda, T., Sunseri, J., Jia, A., Iovanisci, R. B., Snyder, I., and Koes, D. R. Three-dimensional convolutional neural networks and a cross-docked data set for structure-based drug design. *Journal of Chemical Information and Modeling*, 60(9):4200–4215, 2020. doi: 10.1021/acs.jcim.0c00411. URL https://doi.org/10.1021/acs.jcim.0c00411.

Gemma Team. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118. arXiv preprint arXiv:2408.00118.

Guan, J., Qian, W. W., Peng, X., Su, Y., Peng, J., and Ma, J. 3D equivariant diffusion for target-aware molecule generation and affinity prediction. In *The Eleventh International Conference on Learning Representations*, 2023a. URL https://openreview.net/forum?id=kJqXEPXMsE0.

Guan, J., Zhou, X., Yang, Y., Bao, Y., Peng, J., Ma, J., Liu, Q., Wang, L., and Gu, Q. DecompDiff: Diffusion models with decomposed priors for structure-based drug design. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 11827–11846. PMLR, 2023b. URL https://proceedings.mlr.press/v202/guan23a.html.

Harris, C., Didi, K., Jamasb, A. R., Joshi, C. K., Mathis, S. V., Lio, P., and Blundell, T. Benchmarking generated poses: How rational is structure-based drug design with generative models?, 2023. URL https://arxiv.org/abs/2308.07413. arXiv preprint arXiv:2308.07413.

Huang, L., Xu, T., Yu, Y., Zhao, P., Chen, X., Han, J., Xie, Z., Li, H., Zhong, W., Wong, K.-C., and Zhang, H. A dual diffusion model enables 3D molecule generation and lead optimization based on target pockets. *Nature Communications*, 15(1):2657, 2024a. doi: 10.1038/s41467-024-46569-1. URL https://doi.org/10.1038/s41467-024-46569-1.

Huang, Z., Yang, L., Zhou, X., Qin, C., Yu, Y., Zheng, X., Zhou, Z., Zhang, W., Wang, Y., and Yang, W. Interaction-based retrieval-augmented diffusion models for protein-specific 3D molecule generation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 20348–20364. PMLR, 2024b. URL https://proceedings.mlr.press/v235/huang24ab.html.

Imrie, F., Hadfield, T. E., Bradley, A. R., and Deane, C. M. Deep generative design with 3D pharmacophoric constraints. *Chemical Science*, 12(43):14577–14589, 2021. doi: 10.1039/D1SC02436A. URL https://doi.org/10.1039/D1SC02436A.

Jadhav, A., Ferreira, R. S., Klumpp, C., Mott, B. T., Austin, C. P., Inglese, J., Thomas, C. J., Maloney, D. J., Shoichet, B. K., and Simeonov, A. Quantitative analyses of aggregation, autofluorescence, and reactivity artifacts in a screen for inhibitors of a thiol protease. *Journal of Medicinal Chemistry*, 53(1):37–51, 2010. doi: 10.1021/jm901070c. URL https://doi.org/10.1021/jm901070c.

Jiang, Y., Zhang, G., You, J., Zhang, H., Yao, R., Xie, H., Zhang, L., Xia, Z., Dai, M., Wu, Y., Li, L., and Yang, S. PocketFlow is a data-and-knowledge-driven structure-based molecular generative model. *Nature Machine Intelligence*, 6(3):326–337, 2024. doi: 10.1038/s42256-024-00808-8. URL https://doi.org/10.1038/s42256-024-00808-8.

Jin, W., Barzilay, R., and Jaakkola, T. Multi-objective molecule generation using interpretable substructures. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 4849–4859. PMLR, 2020. URL https://proceedings.mlr.press/v119/jin20b.html.

Jing, B., Eismann, S., Suriana, P., Townshend, R. J. L., and Dror, R. O. Learning from protein structure with geometric vector perceptrons. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=1YLJDvSx6J4.

Kallenborn, F., Chacon, A., Hundt, C., Sirelkhatim, H., Didi, K., Cha, S., Dallago, C., Mirdita, M., Schmidt, B., and Steinegger, M. Gpu-accelerated homology search with MMseqs2. *Nature Methods*, 22(10):2024–2027, 2025. doi: 10.1038/s41592-025-02819-8. URL https://doi.org/10.1038/s41592-025-02819-8.

Krenn, M., Häse, F., Nigam, A., Friederich, P., and Aspuru-Guzik, A. Self-referencing embedded strings (SELFIES): A 100% robust molecular string representation. *Machine Learning: Science and Technology*, 1(4):045024, 2020. doi: 10.1088/2632-2153/aba947. URL https://doi.org/10.1088/2632-2153/aba947.

Krenn, M., Ai, Q., Barthel, S., Carson, N., Frei, A., Frey, N. C., Friederich, P., Gaudin, T., Gayle, A. A., Jablonka, K. M., Lameiro, R. F., Lemm, D., Lo, A., Moosavi, S. M., Nápoles-Duarte, J. M., Nigam, A., Pollice, R., Rajan, K., Schatzschneider, U., Schwaller, P., Skreta, M., Smit, B., Strieth-Kalthoff, F., Sun, C., Tom, G., von Rudorff, G. F., Wang, A., White, A. D., Young, A., Yu, R., and Aspuru-Guzik, A. SELFIES and the future of molecular string representations. *Patterns*, 3(10):100588, 2022. doi: 10.1016/j.patter.2022.100588. URL https://doi.org/10.1016/j.patter.2022.100588.

Lee, S., Kreis, K., Veccham, S. P., Liu, M., Reidenbach, D., Peng, Y., Paliwal, S. G., Nie, W., and Vahdat, A. GenMol: A drug discovery generalist with discrete diffusion. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 33205–33226. PMLR, 2025. URL https://proceedings.mlr.press/v267/lee25o.html.

Lemos, P., Beckwith, Z., Bandi, S., van Damme, M., Crivelli-Decker, J., Shields, B. J., Merth, T., Jha, P. K.,

De Mitri, N., Callahan, T. J., Nish, A., Abruzzo, P., Salomon-Ferrer, R., and Ganahl, M. SAIR: Enabling deep learning for protein-ligand interactions with a synthetic structural dataset, 2025. URL https://doi.org/10.1101/2025.06.17.660168. bioRxiv preprint.

Li, Y., Su, M., Liu, Z., Li, J., Liu, J., Han, L., and Wang, R. Assessing protein-ligand interaction scoring functions with the CASF-2013 benchmark. *Nature Protocols*, 13(4):666–680, 2018. doi: 10.1038/nprot.2017.114. URL https://doi.org/10.1038/nprot.2017.114.

Li, Y., Pei, J., and Lai, L. Structure-based de novo drug design using 3D deep generative models. *Chemical Science*, 12(41):13664–13675, 2021. doi: 10.1039/D1SC04444C. URL https://doi.org/10.1039/D1SC04444C.

Lin, H., Huang, Y., Zhang, O., Ma, S., Liu, M., Li, X., Wu, L., Wang, J., Hou, T., and Li, S. Z. DiffBP: Generative diffusion of 3D molecules for target protein binding. *Chemical Science*, 16(3):1417–1431, 2025. doi: 10.1039/D4SC05894A. URL https://doi.org/10.1039/D4SC05894A.

Lipinski, C. A., Lombardo, F., Dominy, B. W., and Feeney, P. J. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Advanced Drug Delivery Reviews*, 46(1–3):3–26, 2001. doi: 10.1016/S0169-409X(00)00129-0. URL https://doi.org/10.1016/S0169-409X(00)00129-0.

Liu, H., Qin, Y., Niu, Z., Xu, M., Wu, J., Xiao, X., Lei, J., Ran, T., and Chen, H. How good are current pocket-based 3D generative models? the benchmark set and evaluation of protein pocket-based 3D molecular generative models. *Journal of Chemical Information and Modeling*, 64(24):9260–9275, 2024. doi: 10.1021/acs.jcim.4c01598. URL https://doi.org/10.1021/acs.jcim.4c01598.

Liu, M., Luo, Y., Uchino, K., Maruhashi, K., and Ji, S. Generating 3D molecules for target protein binding. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 13912–13924. PMLR, 2022. URL https://proceedings.mlr.press/v162/liu22m.html.

Liu, Z., Li, Y., Han, L., Li, J., Liu, J., Zhao, Z., Nie, W., Liu, Y., and Wang, R. PDB-wide collection of binding data: Current status of the PDBbind database. *Bioinformatics*, 31(3):405–412, 2015. doi: 10.1093/bioinformatics/btu626. URL https://doi.org/10.1093/bioinformatics/btu626.

Luo, S., Guan, J., Ma, J., and Peng, J. A 3D generative model for structure-based drug design. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 6229–6239. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/314450613369e0ee72d0da7f6fee773c-Paper.pdf.

Luo, Y., Yang, K., Hong, M., Liu, X. Y., and Nie, Z. MolFM: A multimodal molecular foundation model, 2023. URL https://arxiv.org/abs/2307.09484. arXiv preprint arXiv:2307.09484.

Mysinger, M. M., Carchia, M., Irwin, J. J., and Shoichet, B. K. Directory of useful decoys, enhanced (DUD-E): Better ligands and decoys for better benchmarking. *Journal of Medicinal Chemistry*, 55(14):6582–6594, 2012. doi: 10.1021/jm300687e. URL https://doi.org/10.1021/jm300687e.

Nie, D., Zhao, H., Zhang, O., Weng, G., Zhang, H., Jin, J., Lin, H., Huang, Y., Liu, L., Li, D., Hou, T., and Kang, Y. Durian: A comprehensive benchmark for structure-based 3D molecular generation. *Journal of Chemical Information and Modeling*, 65(1):173–186, 2025. doi: 10.1021/acs.jcim.4c02232. URL https://doi.org/10.1021/acs.jcim.4c02232.

Noutahi, E., Gabellini, C., Craig, M., Lim, J. S. C., and Tossou, P. Gotta be SAFE: A new framework for molecular design. *Digital Discovery*, 3(4):796–804, 2024. doi: 10.1039/D4DD00019F. URL https://doi.org/10.1039/D4DD00019F.

Peng, J., Yu, J.-L., Yang, Z.-B., Chen, Y.-T., Wei, S.-Q., Meng, F.-B., Wang, Y.-G., Huang, X.-T., and Li, G.-B. Pharmacophore-oriented 3D molecular generation toward efficient feature-customized drug discovery. *Nature Computational Science*, 5(10):898–914, 2025. doi: 10.1038/s43588-025-00850-5. URL https://doi.org/10.1038/s43588-025-00850-5.

Peng, X., Luo, S., Guan, J., Xie, Q., Peng, J., and Ma, J. Pocket2Mol: Efficient molecular sampling based on 3D protein pockets. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 17644–17655. PMLR, 2022. URL https://proceedings.mlr.press/v162/peng22b.html.

Peng, X., Guo, R., Guo, F., Wang, Z., Sun, J., Guan, J., Jia, Y., Xu, Y., Huang, Y., Zhang, M., Peng, J., Wang, X., Han, C., Wang, Z., and Ma, J. Unified modeling of 3D molecular generation via atomic interactions with

PocketXMol. *Cell*, 189(7):1904–1922.e28, 2026. doi: 10.1016/j.cell.2026.01.003. URL https://doi.org/10.1016/j.cell.2026.01.003.

Polykovskiy, D., Zhebrak, A., Sanchez-Lengeling, B., Golovanov, S., Tatanov, O., Belyaev, S., Kurbanov, R., Artamonov, A., Aladinskiy, V., Veselov, M., Kadurin, A., Johansson, S., Chen, H., Nikolenko, S., Aspuru-Guzik, A., and Zhavoronkov, A. Molecular sets (MOSES): A benchmarking platform for molecular generation models. *Frontiers in Pharmacology*, 11:565644, 2020. doi: 10.3389/fphar.2020.565644. URL https://doi.org/10.3389/fphar.2020.565644.

Qu, Y., Qiu, K., Song, Y., Gong, J., Han, J., Zheng, M., Zhou, H., and Ma, W.-Y. MolCRAFT: Structure-based drug design in continuous parameter space. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 41749–41768. PMLR, 2024. URL https://proceedings.mlr.press/v235/qu24a.html.

Ragoza, M., Masuda, T., and Koes, D. R. Generating 3D molecules conditional on receptor binding sites with deep generative models. *Chemical Science*, 13(9):2701–2713, 2022. doi: 10.1039/D1SC05976A. URL https://doi.org/10.1039/D1SC05976A.

Schneuing, A., Harris, C., Du, Y., Didi, K., Jamasb, A., Igashov, I., Du, W., Gomes, C., Blundell, T. L., Lio, P., Welling, M., Bronstein, M., and Correia, B. Structure-based drug design with equivariant diffusion models. *Nature Computational Science*, 4(12):899–909, 2024. doi: 10.1038/s43588-024-00737-x. URL https://doi.org/10.1038/s43588-024-00737-x.

Schneuing, A., Igashov, I., Dobbelstein, A. W., Castiglione, T., Bronstein, M. M., and Correia, B. Multi-domain distribution learning for de novo drug design. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=g3VCIM94ke.

Schuffenhauer, A., Schneider, N., Hintermann, S., Auld, D., Blank, J., Cotesta, S., Engeloch, C., Fechner, N., Gaul, C., Giovannoni, J., Jansen, J., Joslin, J., Krastel, P., Lounkine, E., Manchester, J., Monovich, L. G., Pellicciioli, A. P., Schwarze, M., Shultz, M. D., Stiefl, N., and Baeschlin, D. K. Evolution of Novartis' small molecule screening deck design. *Journal of Medicinal Chemistry*, 63(23):14425–14447, 2020. doi: 10.1021/acs.jmedchem.0c01332. URL https://doi.org/10.1021/acs.jmedchem.0c01332.

Skalic, M., Jiménez, J., Sabbadin, D., and De Fabritiis, G. Shape-based generative modeling for de novo drug design. *Journal of Chemical Information and Modeling*, 59(3):1205–1214, 2019. doi: 10.1021/acs.jcim.8b00706. URL https://doi.org/10.1021/acs.jcim.8b00706.

Steinegger, M. and Söding, J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature Biotechnology*, 35(11):1026–1028, 2017. doi: 10.1038/nbt.3988. URL https://doi.org/10.1038/nbt.3988.

Su, M., Yang, Q., Du, Y., Feng, G., Liu, Z., Li, Y., and Wang, R. Comparative assessment of scoring functions: The CASF-2016 update. *Journal of Chemical Information and Modeling*, 59(2):895–913, 2019. doi: 10.1021/acs.jcim.8b00545. URL https://doi.org/10.1021/acs.jcim.8b00545.

Trott, O. and Olson, A. J. AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of Computational Chemistry*, 31(2):455–461, 2010. doi: 10.1002/jcc.21334. URL https://doi.org/10.1002/jcc.21334.

Wang, C. and Rajapakse, J. C. Pharmacophore-guided de novo drug design with diffusion bridge, 2024. URL https://arxiv.org/abs/2412.19812. arXiv preprint arXiv:2412.19812.

Wang, R., Fang, X., Lu, Y., and Wang, S. The PDBbind database: Collection of binding affinities for protein-ligand complexes with known three-dimensional structures. *Journal of Medicinal Chemistry*, 47(12):2977–2980, 2004. doi: 10.1021/jm030580l. URL https://doi.org/10.1021/jm030580l.

Weininger, D. 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. doi: 10.1021/ci00057a005. URL https://doi.org/10.1021/ci00057a005.

Wu, J.-N., Wang, T., Chen, Y., Tang, L.-J., Wu, H.-L., and Yu, R.-Q. t-SMILES: A fragment-based molecular representation framework for de novo ligand design. *Nature Communications*, 15(1):4993, 2024a. doi: 10.1038/s41467-024-49388-6. URL https://doi.org/10.1038/s41467-024-49388-6.

Wu, K., Xia, Y., Deng, P., Liu, R., Zhang, Y., Guo, H., Cui, Y., Pei, Q., Wu, L., Xie, S., Chen, S., Lu, X., Hu, S., Wu, J., Chan, C.-K., Chen, S., Zhou, L., Yu, N., Chen, E., Liu, H., Guo, J., Qin, T., and Liu, T.-Y. TamGen: Drug design with target-aware molecule generation through a chemical language model. *Nature Communications*, 15(1):9360,

2024b. doi: 10.1038/s41467-024-53632-4. URL https://doi.org/10.1038/s41467-024-53632-4.

Xie, W., Zhang, J., Xie, Q., Gong, C., Ren, Y., Xie, J., Sun, Q., Xu, Y., Lai, L., and Pei, J. Accelerating discovery of bioactive ligands with pharmacophore-informed generative models. *Nature Communications*, 16(1):2391, 2025. doi: 10.1038/s41467-025-56349-0. URL https://doi.org/10.1038/s41467-025-56349-0.

Zhang, O., Zhang, J., Jin, J., Zhang, X., Hu, R., Shen, C., Cao, H., Du, H., Kang, Y., Deng, Y., Liu, F., Chen, G., Hsieh, C.-Y., and Hou, T. ResGen is a pocket-aware 3D molecular generation model based on parallel multiscale modelling. *Nature Machine Intelligence*, 5(9):1020–1030, 2023. doi: 10.1038/s42256-023-00712-7. URL https://doi.org/10.1038/s42256-023-00712-7.

Zhu, H., Zhou, R., Cao, D., Tang, J., and Li, M. A pharmacophore-guided deep learning approach for bioactive molecular generation. *Nature Communications*, 14(1):6234, 2023. doi: 10.1038/s41467-023-41454-9. URL https://doi.org/10.1038/s41467-023-41454-9.

Zhung, W., Kim, H., and Kim, W. Y. 3D molecular generative framework for interaction-guided drug design. *Nature Communications*, 15(1):2688, 2024. doi: 10.1038/s41467-024-47011-2. URL https://doi.org/10.1038/s41467-024-47011-2.

A. Alignment of Unconditionally Generated Molecules with the Training Distribution

To verify that unconditional sampling reproduces the chemistry of the pretraining corpus rather than collapsing onto a narrow region or drifting into unrealistic chemical space, we compare 10,000 molecules sampled from Molexar against the training set along two complementary views. Figure 8 overlays the marginal distributions of nine molecular properties, and Figure 9 compares the two populations in a shared low-dimensional embedding of their structural fingerprints. Across both views the generated molecules closely track the training distribution, indicating faithful distribution learning without mode collapse.

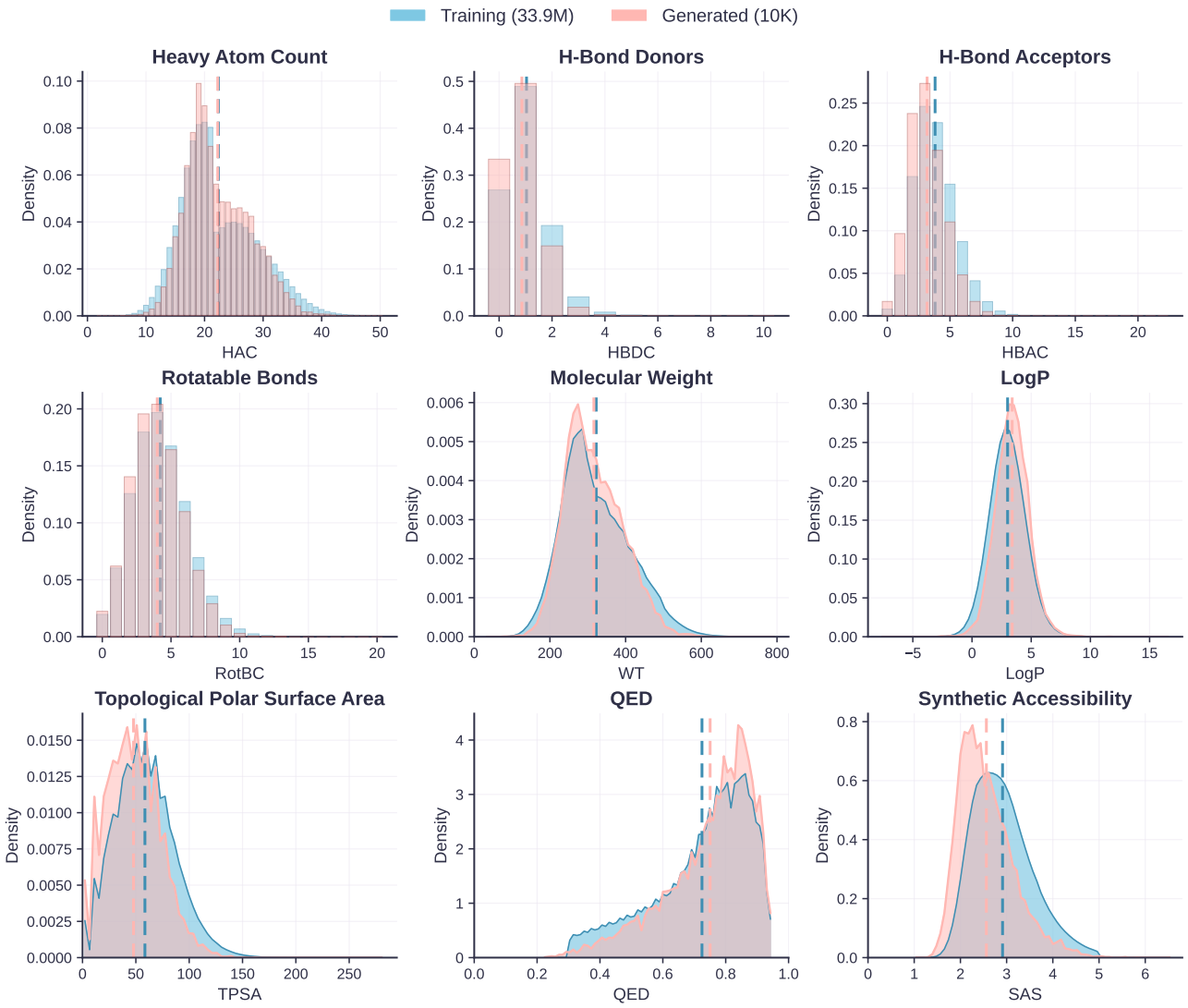


Figure 8. **Marginal property distributions of unconditionally generated molecules versus the training set.** Each panel overlays the normalized histogram and kernel-density estimate of one molecular property for 10,000 molecules sampled from Molexar (pink) and for the 33.9M-molecule training corpus (green); dashed vertical lines mark the respective means.

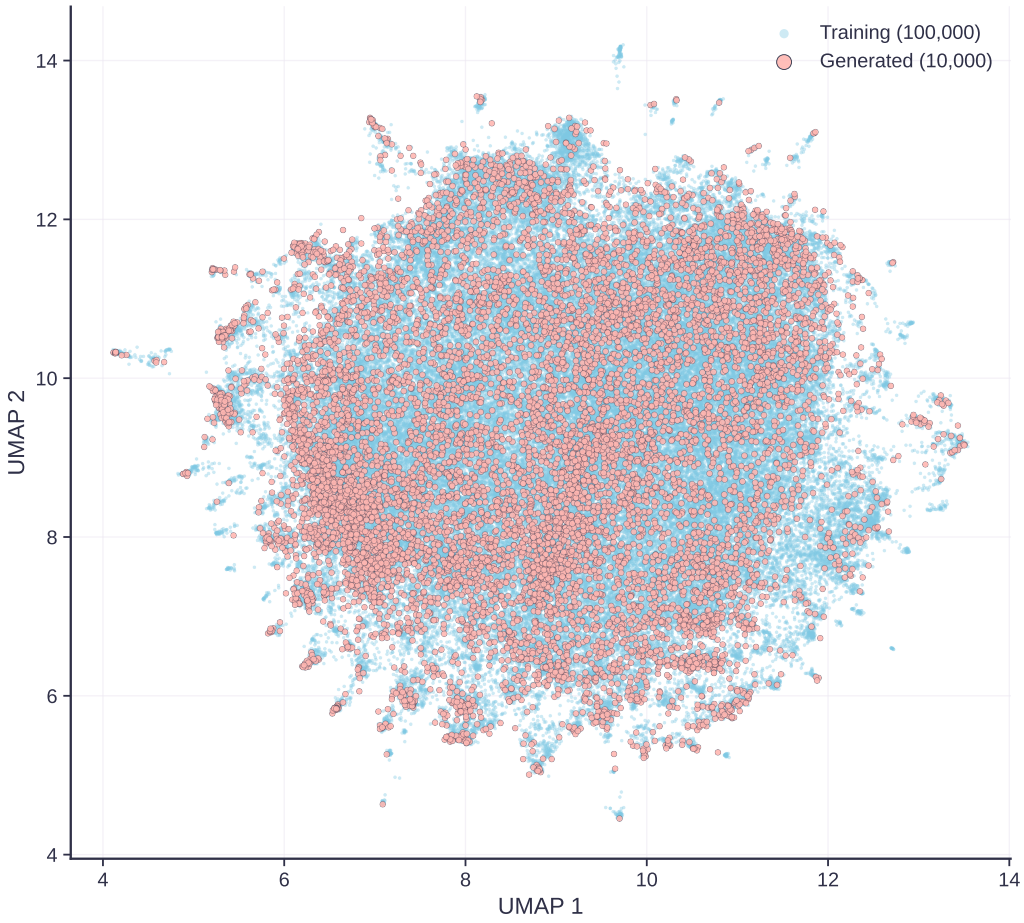


Figure 9. **UMAP embedding of ECFP4 fingerprints for 10,000 unconditionally generated molecules (pink) and 100,000 molecules subsampled from the training set (green).**

B. Single-Property Obedience versus Training-Set Abundance

To quantify how reliably single-property conditioning obeys the requested instruction, and in particular whether obedience is retained for target values that are scarce in the training set, we conducted a detailed analysis of the per-property obedience rate as a function of training-set abundance. For discrete properties, we enumerated all available sampling points and defined the obedience rate as the fraction of generated molecules whose property value exactly matches the requested point. For continuous properties, we spaced the sampling points by the RBF encoding step size and defined the obedience rate as the fraction of molecules falling within a tolerance window of the target, where the tolerance is set to that same step size. The sampling ranges, step counts, and tolerances for all nine properties are listed in Table 3, and we generated 1,000 molecules at each sampling point.

Figure 10 reports the obedience rate at each sampling point together with the abundance of the corresponding property value in the training set. The obedience rate is correlated with training-set abundance, but the conditioning also generalizes: relative to the densest region of the training data, Molexar retains a high obedience rate even at sampling points where training data are scarce.

Table 3. **Single-property conditioning configurations.** Discrete properties use one-hot encoding (exact match); continuous properties use radial basis function (RBF) encoding. The tolerance is the obedience window half-width: a generated molecule counts as matching when its property lies within $\pm$tolerance of the target.

Property	Method	Range	Steps	Tolerance
Heavy Atom Count (HAC)	one-hot	$[2, 50]$	49	exact
H-Bond Donors (HBDC)	one-hot	$[0, 10]$	11	exact
H-Bond Acceptors (HBAC)	one-hot	$[0, 22]$	23	exact
Rotatable Bonds (RotBC)	one-hot	$[0, 20]$	21	exact
Molecular Weight (WT, Da)	RBF	$[30, 750]$	128	± 5.67
LogP	RBF	$[-6, 12]$	96	± 0.19
Topological Polar Surface Area (TPSA)	RBF	$[0, 200]$	96	± 2.11
QED	RBF	$[0.3, 1.0]$	64	± 0.011
Synthetic Accessibility (SAS)	RBF	$[1.0, 5.0]$	64	± 0.063

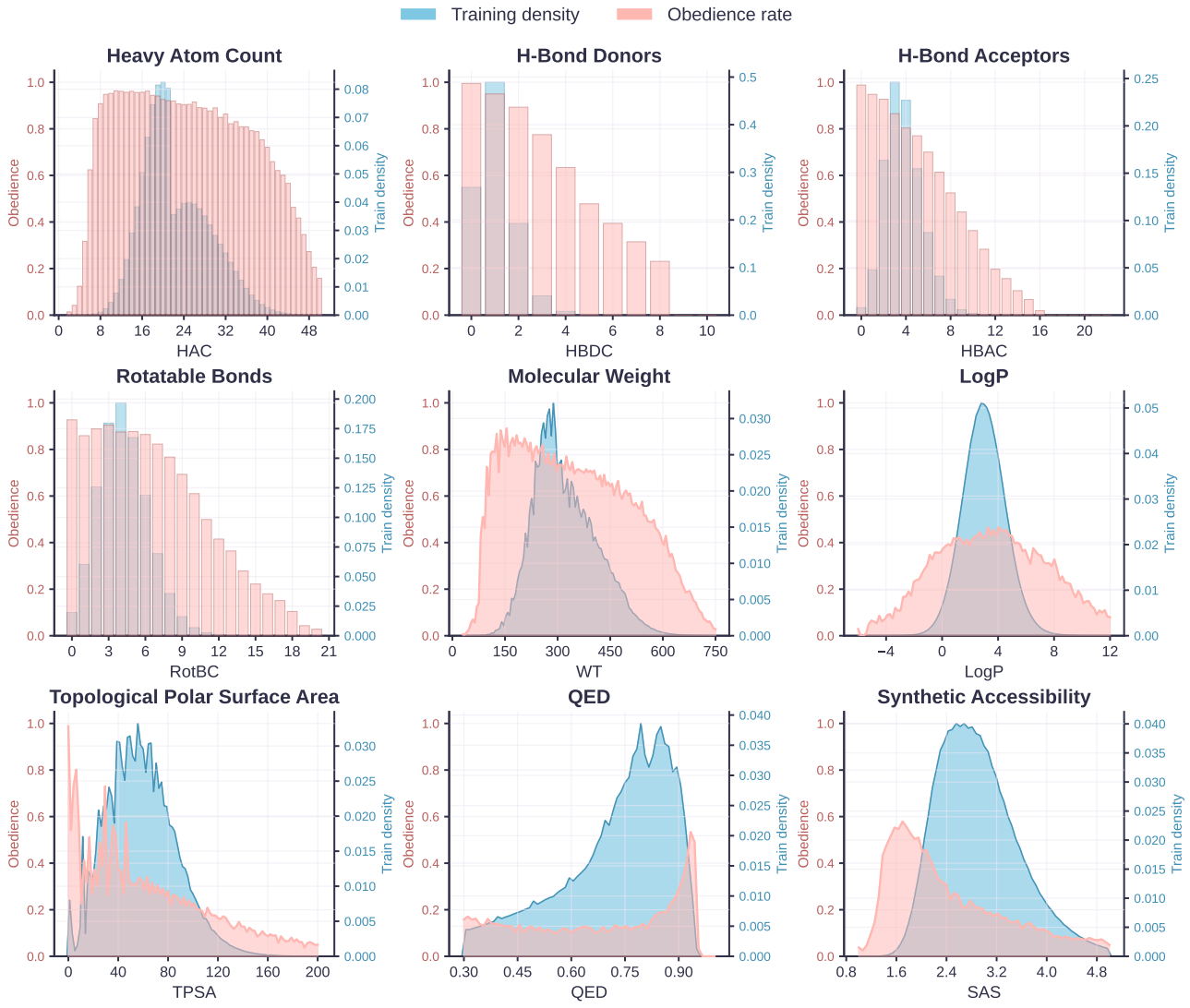


Figure 10. **Single-property obedience rate versus training-set abundance for the nine conditioned properties.** In each panel, the pink bars show the obedience rate at each sampling point (left axis) and the green histogram shows the abundance of that property value in the training set (right axis).

C. Correlation between Molecular Properties in the Training Set

Because different molecular properties can be correlated, and such correlations may affect the model's behavior under multi-property conditioning, we computed the pairwise Pearson correlation coefficients among the nine conditioning properties in the training set and report them in Figure 11. The figure shows the degree of correlation among the properties and provides context for interpreting the model's behavior under multi-property conditioning.

The correlations affect our chosen tasks as follows. HAC and RotBC are relatively strongly correlated ($r = 0.48$), yet Molexar maintains largely independent control over the two in this setting, consistent with its strong obedience. QED and SAS are nearly uncorrelated ($r = -0.09$), indicating that they vary relatively independently in the training set. WT and LogP are more strongly correlated ($r = 0.53$), and the positively correlated target values we chose in the corresponding test follow this trend.

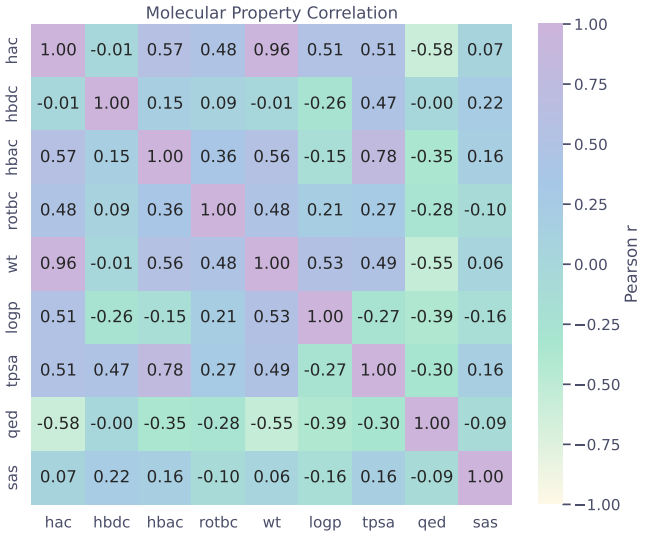


Figure 11. **Pairwise Pearson correlation coefficients (r) between the nine conditioning properties in the training set.**

D. Target-Conditioned Generation on the CrossDocked2020 Benchmark

We provide the detailed per-metric results behind Figure 6 in Tables 4 and 5.

Table 4. **Pocket- and sequence-conditioned generation results.**

Method	Valid.	Uniq.	Div.	QED	SA	Lip.	Vina	HA
AR	0.93	**1.00**	0.84	0.51	0.64	4.75	-6.87	37.8
Pocket2Mol	0.98*	**1.00**	0.87*	0.57	0.75	4.88*	-7.39	48.4
DeepICL	0.99	0.99	0.86	0.61	0.33	**4.94**	-7.32	48.3
TargetDiff	0.92	0.99	0.89	0.49	0.60	4.58	-7.70	53.5
DiffSBDD	**1.00**	**1.00**	**0.90**	0.47	0.58	4.53	-7.27	46.8
PMDM	0.99	**1.00**	**0.90**	0.56	0.62	4.78	-7.49*	49.5
DrugFlow	0.85	0.96	0.84	0.51	0.71	4.52	-7.18	46.3
ResGen	0.97	**1.00**	0.85	0.59*	0.79*	4.91	-6.58	34.6
TamGen	**1.00**	**1.00**	0.87*	0.56	0.77	4.83	-6.85	35.8
PocketXMol	0.94	0.97	0.84	0.51	0.77	4.57	**-7.81**	**57.6**
OMTRA (pocket)	0.94	0.97	0.87*	0.50	0.77	4.32	-6.86	40.6
PhoreGen (complex)	0.70	0.93	0.80	0.42	0.71	4.07	-7.24	52.3
OMTRA (complex)	0.66	0.96	0.80	0.34	0.63	3.61	-7.17	50.0
Molexar (sequence)	**1.00**	0.98*	0.83	**0.65**	0.82	4.74	-7.25	43.1
Molexar (pocket)	**1.00**	0.97	0.84	**0.65**	**0.83**	4.82	-7.42	53.0*
Reference	1.00	-	-	0.48	0.73	4.27	-7.45	-

Table 5. **Pharmacophore-conditioned generation results.**

Method	Valid.	Uniq.	Div.	QED	SA	Lip.	Vina	HA
PGMG	0.94	**1.00**	0.80	0.50	0.77	4.40*	**-7.23**	45.2
ShEPhERD	0.81*	0.78	0.80	0.43*	0.55	**4.84**	-6.06	13.6
PhoreGen (pharmacophore)	0.64	0.96	**0.82**	0.38	0.69	4.07	-6.80	**47.7**
OMTRA (pharmacophore)	0.36	0.90	0.57	0.17	**0.86**	2.84	-5.19	8.6
Molexar (pharmacophore)	**1.00**	0.91*	0.76*	**0.59**	0.71*	4.69	-6.79*	38.4*
Reference	**1.00**	-	-	0.48	0.73	4.27	-7.45	-

Blue : pocket-only; Red : pharmacophore-only; Purple : complex (pocket+pharmacophore).
Valid.: Validity; Uniq.: Uniqueness.; Div.: Diversity; SA: Normalized synthetic accessibility; Lip.: Lipinski's rule of five; Vina: Vina docking score (docking values greater than 0 are excluded); HA: High affinity ratio.
Top 3 results are highlighted with **bold** text, underlined text, and *, respectively.

E. General Properties of Molexar on MolGenBench

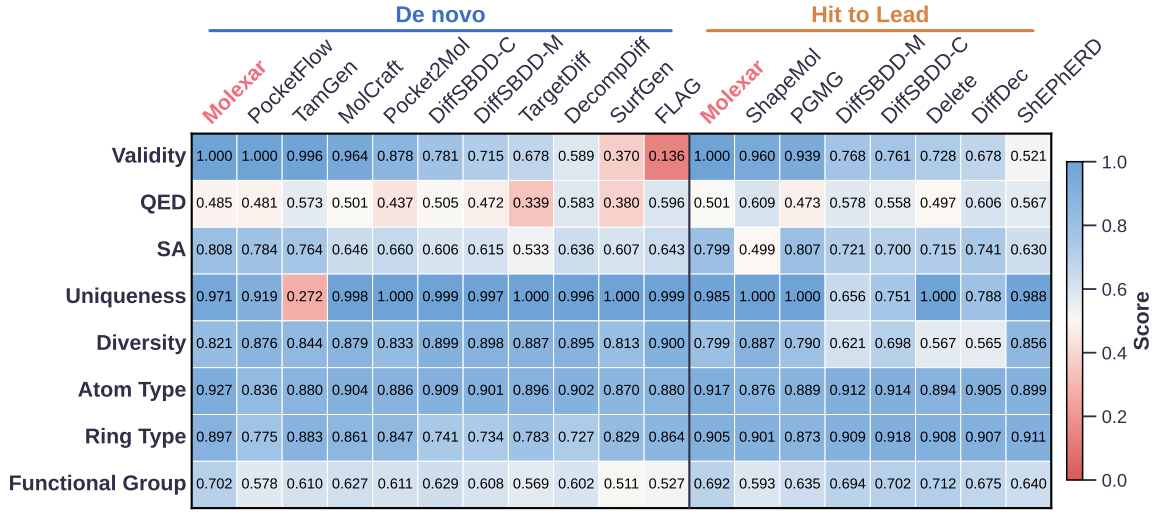


Figure 12. **General properties of Molexar on MolGenBench.** Evaluation results of basic molecular properties, including validity, QED, SA score, uniqueness, diversity, and motif distributions (atom types, ring types, functional groups) with reference actives. Values represent the mean from three independent replicates; all metrics are normalized to a 0–1 scale, where higher values indicate better performance.

F. Examples of Generated Molecules

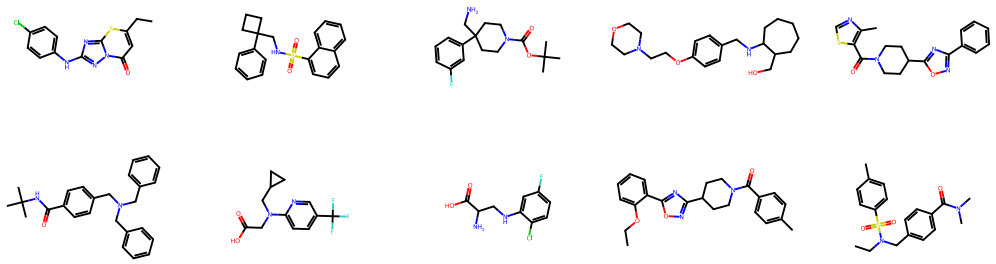


Figure 13. **Examples of generated molecules on *de novo* generation.**

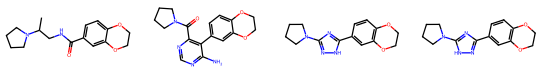
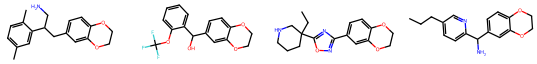
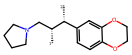
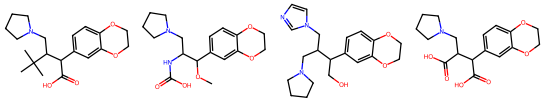
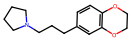
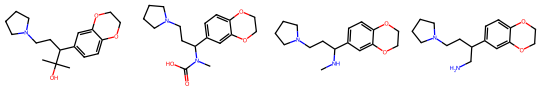
Task	Input	Generated molecules
Linker design & Scaffold morphing		
Motif extension		
Scaffold decoration		
Superstructure generation		

Figure 14. **Examples of generated molecules on fragment-constrained generation of Eliglustat.**

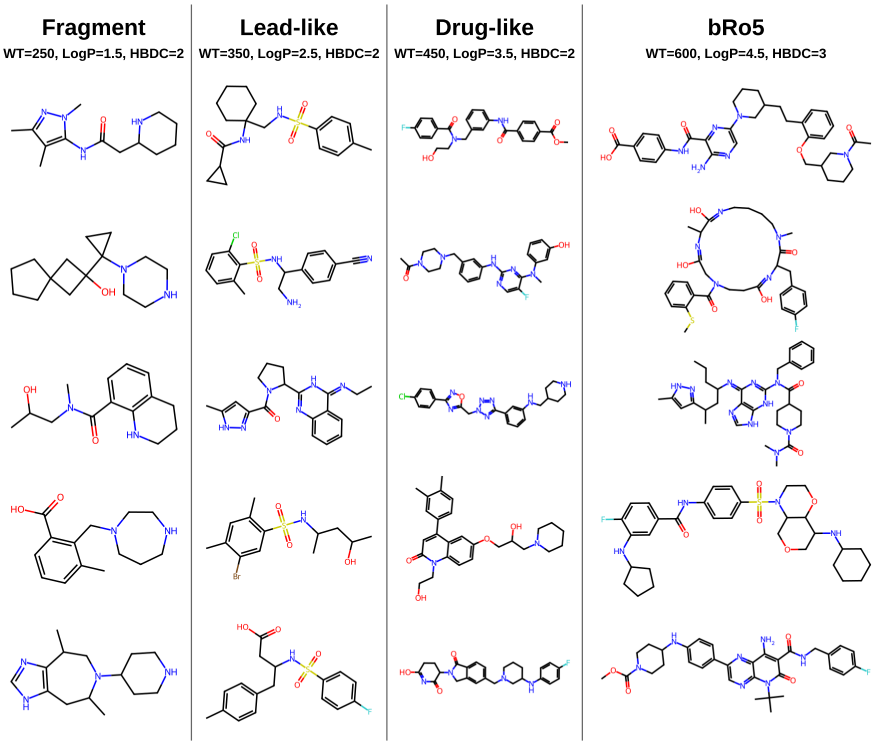


Figure 15. **Examples of molecules generated under the realistic drug-regime multi-property settings.**