SEQUENCE-CONDITIONED LIGAND GENERATION WITH COMPACT PROTEIN PREFIXES

Anonymous authors
Paper under double-blind review

ABSTRACT

Designing ligands from protein sequences could extend drug discovery to targets without well-characterized binding-pocket structures. We develop a compact autoregressive Transformer that generates ligand SMILES from frozen ESM3 residue embeddings and a separate affinity input. A learned projection and fixed segment pooling provide regional protein summaries while keeping the decoder's protein-conditioning length independent of protein length. On a 42-target panel, selected ligands generated using the target's own prefix achieve higher mean predicted affinity scores and RF3 interface confidence than those generated using another protein's prefix. Mean RF3 interface predicted TM-score (ipTM) is 0.920 versus 0.829 under the same candidate-selection rule. Our generator has 6.85 million parameters and achieves RF3 interface confidence comparable to PCMol and ProtoBind-Diff on their respective comparison panels. It also yields more distinct, valid, connected molecules for the same number of generations. With cached protein features on the same GPU, our implementation generates candidates 4.3 to 6.1 times faster than the released baseline implementations and uses about one-ninth of their peak allocated GPU memory.

1 INTRODUCTION

Finding ligands that bind to a chosen protein is a key step in drug discovery. Protein-conditioned generative models use protein information to guide the molecular design (Bernatavicius et al., 2024; Mistryukova et al., 2025). We focus on the sequence-only setting, where the target information is obtained from the protein's amino-acid sequence. This allows ligand generation without a binding-pocket structure, but the model must infer suitable ligand features without directly observing the pocket's shape or the spatial arrangement of its residues (Guan et al., 2023). The challenge is to learn this protein–ligand relationship from sequence-derived representations.

Protein language models learn residue representations that encode structural and functional information from amino-acid sequences (Rives et al., 2021). Reusing these representations gives a ligand generator access to protein information learned beyond its paired protein–ligand training data. Passing one token per residue would make the decoder's conditioning length vary across targets and increase attention costs for longer proteins (Vaswani et al., 2017). A fixed number of conditioning tokens gives the decoder the same protein-input length across targets. The challenge is to retain useful target information within this fixed token budget. Locality-aware pooling methods that combine regional summaries with attention have improved performance across protein prediction tasks (Hoang & Singh, 2025). Motivated by these findings, we hypothesize that a compact prefix of regional residue summaries can provide useful target context for ligand generation while keeping the conditioning sequence short.

We develop a compact sequence-conditioned ligand generator that connects frozen ESM3 residue embeddings (Hayes et al., 2025) to a decoder-only Transformer, following the molecular sequence generation approach of MolGPT (Bagal et al., 2022). A learned projection and fixed segment pooling produce a protein prefix whose tokens summarize different contiguous sequence regions. The frozen protein features are computed once per target and reused during generator training and sampling. The decoder generates ligand SMILES (Simplified Molecular Input Line Entry System) strings (Weininger, 1988) from this protein prefix and a separate scalar affinity input.

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

For each target protein, we compare two types of protein–ligand pairs. Matched pairs contain selected ligands generated using the target's own prefix, while mismatched pairs contain those generated using another protein's prefix. The matched pairs achieve higher mean predicted affinity scores and RF3 interface confidence (Table 1). This shows that the protein prefix supplies useful target information for ligand generation. Selected generated ligands also achieve higher mean RF3 interface confidence than molecular-weight-matched ligands measured against other proteins. Our 6.85-million-parameter generator achieves comparable RF3 interface confidence to PCMol and ProtoBind-Diff on their respective comparison panels. Our implementation generates candidates 4.3–6.1 times faster and uses about one-ninth of the baselines' peak allocated GPU memory in cached-feature benchmarks on the same GPU.

2 METHODS

We develop a sequence-conditioned ligand generator that uses pretrained protein representations to guide molecular design without a binding-pocket structure. A learned projection and fixed segment pooling convert frozen ESM3 residue features into a compact protein prefix. This prefix supplies regional protein summaries to an autoregressive SMILES decoder. A separate scalar prefix specifies the requested affinity label, allowing the decoder to use both protein context and the requested value when predicting molecular tokens. Figure 1 summarizes generation and the subsequent candidate selection and structural evaluation.

2.1 SEQUENCE-DERIVED PROTEIN PREFIX

We extract contextual residue features with pretrained ESM3 and keep the encoder frozen throughout generator training (Hayes et al., 2025). Features are computed once per target and cached for reuse during training and repeated sampling. We supply only the amino-acid sequence for feature extraction, although ESM3 was pretrained with sequence, structure, and function modalities.

A single global average combines all residues into one vector. We instead average projected residue features within separate sequence segments, giving the decoder one summary per segment. Locality-aware pooling has improved performance on other protein prediction tasks (Hoang & Singh, 2025), motivating this regional representation. We use the same number of segments for every protein, so the decoder receives the same number of protein-conditioning tokens regardless of protein length.

Let $\mathbf{P} \in \mathbb{R}^{L\times 1536}$ contain the cached residue features, excluding special tokens, with at most $L_{\max} = 2048$ residues retained. We first project each residue to the decoder width $d = 256$:

$$\mathbf{h}_i = \mathrm{LN}_{\mathrm{in}}(\mathbf{W}_p\mathbf{P}_i + \mathbf{b}_p), \qquad i = 1, \ldots, L, \tag{1}$$

where $\mathbf{W}_p \in \mathbb{R}^{256\times 1536}$ and $\mathbf{b}_p \in \mathbb{R}^{256}$ are learned parameters, and LN denotes layer normalization. We divide the sequence into K contiguous segments:

$$S_k = \left\{ i : \left\lfloor \frac{(i-1)K}{L} \right\rfloor = k - 1 \right\}, \qquad k = 1, \ldots, K.$$

Each protein-prefix token is the normalized mean of the projected features in one segment:

$$\mathbf{c}_k = \mathrm{LN}_{\mathrm{out}}\left(\frac{\sum_{i\in S_k} \mathbf{h}_i}{\max(1, |S_k|)} \right), \qquad k = 1, \ldots, K. \tag{2}$$

Padding residues are excluded from these averages. Segment membership is fixed by residue position, while the projection and layer-normalization parameters are learned with the decoder. ESM3 encodes each residue in the context of the input sequence before we pool these features into each segment.

We use $K = 16$ to provide multiple regional summaries within a short conditioning prefix. With one additional affinity token and a molecular stream of length T, the decoder processes $T + 17$ positions. For fixed T, its dense attention matrices therefore remain the same size as protein length varies (Vaswani et al., 2017). Protein length still affects residue-feature extraction, projection, and pooling. In Appendix B.2, we report reduced-budget pilots that informed the pooling choice. The number of prefix tokens can be adjusted to suit the desired performance-computation tradeoff.

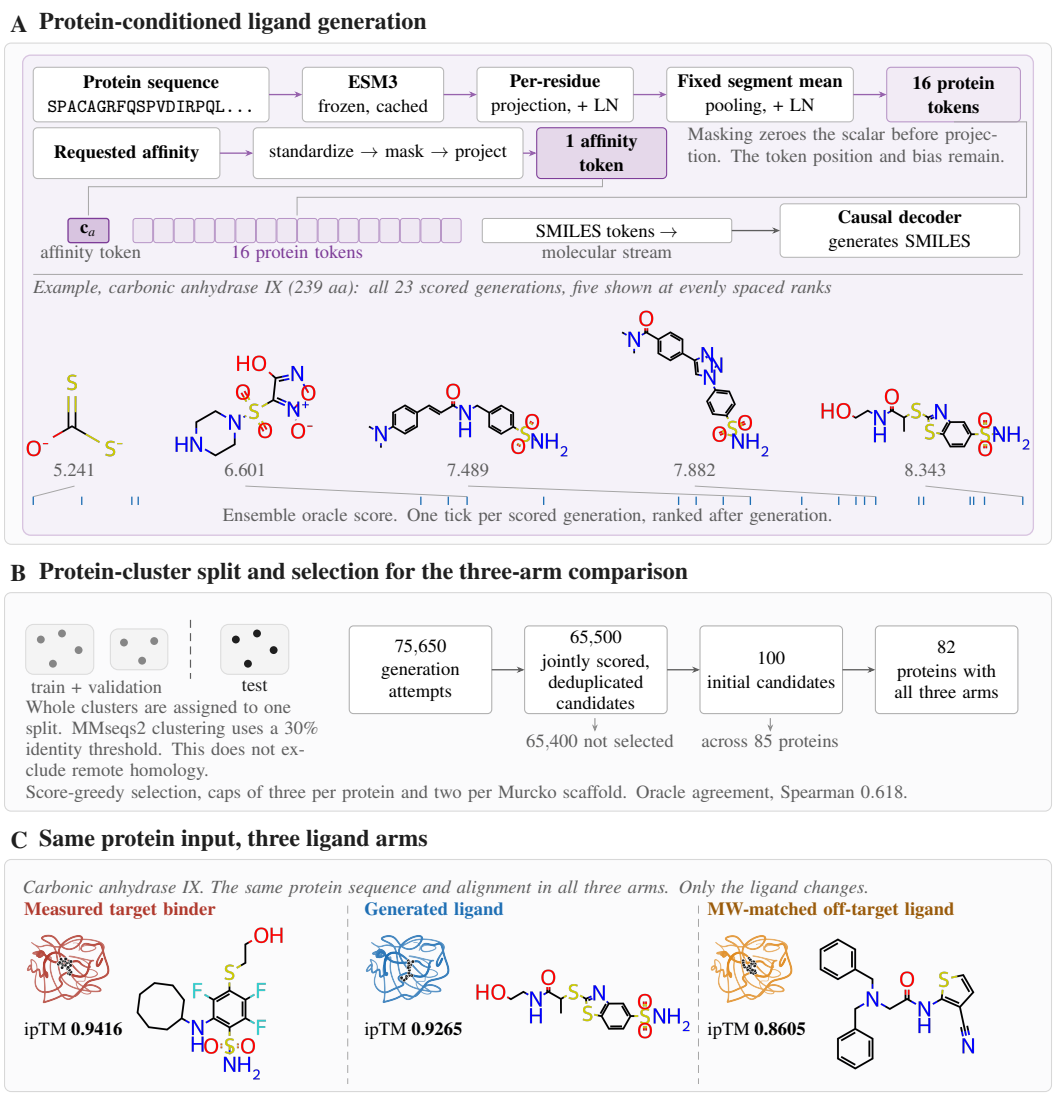


Figure 1: Sequence-conditioned ligand generation and structural evaluation. (A) Frozen ESM3 features form a compact protein prefix for SMILES generation with a separate affinity input. (B) Protein-cluster splitting and candidate selection for the original three-arm screen, including SMILES deduplication across targets. Protein-prefix and baseline comparisons instead select one ligand per target and condition. (C) A measured target binder, a generated ligand candidate, and a molecular-weight-matched off-target ligand share the same protein sequence and alignment. Among the 82 evaluated targets, we selected the target that minimizes the sum of absolute differences between its three ipTM scores and the corresponding group means in Table 15. ipTM measures interface confidence, not binding affinity. Binding is unmeasured for the generated candidate, and the off-target ligand is not a confirmed non-binder.

2.2 Conditional Molecular Decoder and Affinity Input

We generate ligand candidates as SMILES strings so that candidate generation does not require predicting atomic coordinates for every molecule. Prior studies have reported faster sampling with molecular string generators than with direct 3D generators evaluated in their benchmarks (Wu et al., 2024; Dollar et al., 2023). We use a decoder-only causal Transformer, following the generation approach of MolGPT (Bagal et al., 2022). At each molecular position, the decoder attends to the conditioning prefix and preceding molecular tokens to predict the next token.

A separate affinity token lets us change the requested value while keeping the protein representation fixed. The scalar label is $a = -\log_{10}[\text{affinity in M}]$. Because the source collection combines assay

endpoints, we call it a *pAff proxy* and describe its composition in Appendix A.3. Using the training-label mean μ_a and standard deviation σ_a, we construct

$$\widetilde{a} = \frac{a - \mu_a}{\sigma_a + \epsilon}, \qquad \mathbf{c}_a = \mathbf{w}_a(m\widetilde{a}) + \mathbf{b}_a, \qquad m \in \{0, 1\}, \tag{3}$$

where $m = 1$ supplies the normalized request and $m = 0$ sets it to zero before projection. The affinity-token position remains present in both cases. Appendix D.3 describes the masking and permutation controls.

The conditioning sequence places the affinity token before the protein tokens:

$$\mathbf{c} = [\mathbf{c}_a, \mathbf{c}_1, \ldots, \mathbf{c}_{16}], \tag{4}$$

followed by the molecular-token stream. Learned type embeddings distinguish conditioning positions from molecular positions, and molecular tokens receive learned position embeddings. The causal attention mask allows each molecular token to attend to the full conditioning prefix and earlier molecular tokens.

For molecule tokens $x_{1:T}$, we minimize the teacher-forced next-token loss

$$\mathcal{L}(\theta) = -\sum_{t=1}^{T} \log p_\theta(x_t \mid x_{<t}, \mathbf{c}), \tag{5}$$

with a start token initializing the molecular stream. Here, T includes padding positions in the shifted molecular training sequence. The decoder and conditioning projections are trained jointly, while the cached ESM3 features remain fixed. The loss is computed over molecular positions, including padding, and excludes conditioning positions. Architecture and training settings are provided in Appendix B.

2.3 Candidate Prioritization

We rank generated ligand candidates before structural evaluation to concentrate complex predictions on a smaller set of promising candidates. In our development experiments, affinity prediction was less accurate under the protein-cluster split than under a protein-random split. We therefore refined the affinity models used to rank generated candidates (Appendix C.1). Two affinity-prediction models serve as oracles: DeepPurpose's AttentiveFP-based model (Huang et al., 2020; Xiong et al., 2020; Wang et al., 2019) and a ConPLex-based regressor (Singh et al., 2023). Both predict a pAff-proxy score for a protein–ligand pair. Appendix C.2 describes their training and our ConPLex adaptation. For candidates scored by both models, we use their mean as the ranking score:

$$s_{\text{ens}} = \tfrac{1}{2}(s_{\text{DGL}} + s_{\text{ConPLex}}). \tag{6}$$

These oracles are applied after sampling to prioritize candidates for complex prediction. They do not update the generator. We describe the candidate-selection rule alongside each experiment in Results, with detailed protocols in the corresponding appendices.

3 Results

We first test whether the protein prefix provides useful target information, then compare generated ligands with measured-ligand controls and other generators.

3.1 Data and Generation Setting

We use the public `jglaser/binding_affinity` dataset hosted on Hugging Face (Glaser, n.d.). The collection combines records from BindingDB, PDBbind-cn, BioLiP, and Binding MOAD. Each record contains a protein amino-acid sequence, a canonical ligand SMILES string, and a pAff label. Our processed dataset contains 1,836,729 records covering 22,974 distinct protein sequences and 980,310 distinct ligand SMILES. Appendix A describes data preparation, and Table 4 summarizes the split composition.

We assign whole MMseqs2 clusters at a 30% sequence-identity threshold (Steinegger & Söding, 2017) to training, validation, and test sets. The generator and both affinity oracles use this split.

Table 1: Matched and mismatched protein–ligand pairs on 42 targets. Scores are for the oracle-selected ligand in each condition. RF3 ipTM uses the sample selected by RF3 ranking score. Differences are matched minus mismatched, with paired 95% target-bootstrap intervals. Wins count targets on which matched scores are higher.

Metric	Matched	Mismatched	Δ [95% CI]	Wins
Ensemble oracle	8.1245	7.5153	0.6092 [0.5006, 0.7187]	40/42
RF3 ipTM	0.9197	0.8290	0.0907 [0.0602, 0.1211]	39/42

Cluster separation does not exclude all remote homology or prior exposure during ESM3 pretraining. Model development also included checks on the test split, as documented in Appendix B.

The final protein-conditioned checkpoint generates 25 outputs for each of $3{,}026$ test proteins, giving $75{,}650$ generations. Generation counts include invalid and blank outputs. Chemical validity is 88.81%, with $65{,}456$ unique valid canonical outputs. Among these unique valid molecules, 99.47% are absent from the training set. Every test protein has at least one valid output.

3.2 Effect of the Protein Prefix

For each of 42 target proteins, we compare matched and mismatched protein–ligand pairs. Matched pairs contain ligands generated using the target's own prefix. Mismatched pairs contain ligands generated using another protein's prefix, chosen from a different homology cluster and matched approximately by sequence length.

For each target, we generate 25 outputs under each condition, using the same affinity condition and decoding settings. We score both candidate sets against the intended target using the same two affinity oracles and select the ligand with the highest mean oracle score from each set. Then RF3 (Corley et al., 2025) evaluates both selected ligands with the same target sequence, alignment, and inference settings. Appendix D.1 describes the fixed target panel, prefix assignment, and evaluation protocol. The oracles predict a mixed-endpoint proxy and are not direct measurements of generated-ligand affinity.

Matched pairs achieve higher scores at both stages (Table 1). Their mean ensemble oracle score is 8.1245, compared with 7.5153 for mismatched pairs, with higher scores on 40 of 42 targets. Mean RF3 ipTM is also higher, at 0.9197 versus 0.8290, favoring matched pairs on 39 targets.

The advantage remains after both candidate sets undergo the same target-aware selection. These results show that the compact protein prefix provides useful target information for ligand generation under the protein-cluster evaluation setting.

3.3 Three-Arm Structural Comparison

The multi-target screen yields 100 generated protein–ligand complexes across 84 proteins after 3D input preparation. Candidates are selected by oracle score with caps on the number per protein and scaffold. Appendix C.3 describes the initial selection and candidate replacements.

We compare generated ligands with two measured-ligand controls for the same protein: a measured target binder and a molecular-weight-matched ligand measured against another protein in a different homology cluster. The latter serves as an off-target measured control, not a verified non-binder. RF3 uses the same target sequence and alignment across all three arms. We analyze the 82 proteins with complete results.

Within the 100 selected generated complexes, ensemble oracle score has little association with subsequent RF3 interface confidence. The Spearman correlation is 0.035, and mean ipTM differs by about 0.005 between the upper and lower halves of the ensemble ranking. Thus, within this selected shortlist, higher oracle scores do not consistently correspond to higher RF3 confidence.

In the three-arm comparison, generated ligands achieve higher RF3 interface confidence than the off-target measured controls, while measured target binders score highest (Figure 2). Mean ipTM is 0.9456 for measured target binders, 0.9140 for generated ligands, and 0.8619 for off-target measured

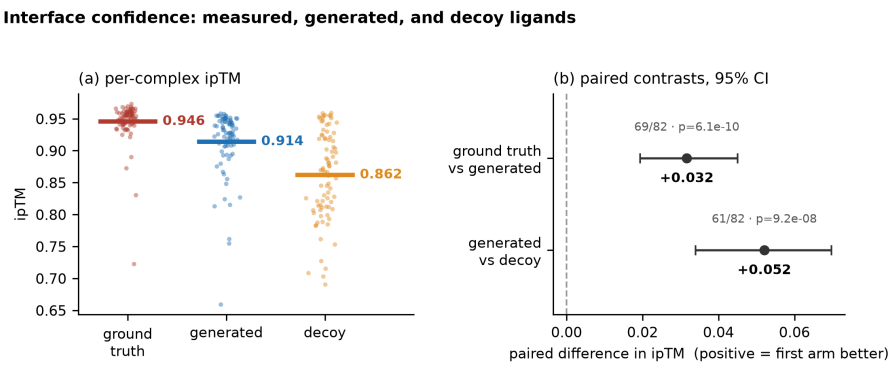


Figure 2: RF3 interface confidence for three ligand arms on the same 82 proteins. Each triplet keeps the protein and alignment fixed and changes the ligand. (a) Per-complex ipTM and arm means. (b) Paired differences with bootstrap 95% intervals, with positive values favoring the first-named arm. Generated ligands score below measured target binders and above molecular-weight-matched ligands of other proteins. The latter are off-target measured controls, not verified non-binders.

Table 2: RF3 comparison of candidates from public pretrained generators. For each target and generator, one ligand is selected by the shared oracle-based rule. Ours and Baseline report mean ipTM (↑) for the RF3 sample chosen by ranking score. Each row uses its own 42-target panel. Δ is ours minus baseline, with a paired 95% target-bootstrap interval. Wins count targets with higher ipTM for our model. Absolute means across rows are not directly comparable.

Comparison	Ours	Baseline	Δ [95% CI]	Wins
PCMol	0.7987	0.8095	−0.0108 [−0.0437, 0.0207]	18/42
ProtoBind-Diff	0.9197	0.9113	0.0083 [−0.0068, 0.0224]	22/42

controls. Generated ligands exceed the controls on 61 of 82 proteins, with a mean paired difference of 0.0521. Measured target binders exceed generated ligands on 69 proteins, with a mean difference of 0.0317. Contact-restricted iPAE gives the same ordering. Table 15 and Appendix E provide the full structural summaries. The generator therefore proposes ligands that RF3 rates as more compatible with their intended proteins than the molecular-weight-matched off-target controls.

3.4 Comparisons with Other Generators

Public pretrained generators. We compare our model with public PCMol (Bernatavicius et al., 2024) and ProtoBind-Diff (Mistryukova et al., 2025) checkpoints using their released implementations without retraining. PCMol uses structure-derived AlphaFold representations, whereas ProtoBind-Diff combines ESM2 sequence features with a molecular diffusion language model. Each generator receives 25 generations per target. We select one ligand using the same ensemble oracle score and evaluate it with the common RF3 protocol.

Each baseline comparison uses a separate 42-target panel. The PCMol panel contains targets supported by its released representations, while ProtoBind-Diff uses the protein-prefix control panel. Absence of PCMol training overlap could not be verified, and all 42 ProtoBind-Diff panel targets have homologs in that baseline's public training data. Appendix F reports checkpoints, panel construction, and overlap audit.

Our model achieves mean RF3 ipTM of 0.7987 versus 0.8095 for PCMol and 0.9197 versus 0.9113 for ProtoBind-Diff (Table 2). It achieves comparable RF3 interface confidence on both comparison panels.

Generator size and generation cost. Our ligand generator has 6.85 million parameters, compared with 101.34 million for PCMol and 159.53 million for ProtoBind-Diff (Table 3). These counts include conditioning modules and exclude the protein feature extractors. With cached protein features on the same NVIDIA A100 40 GB GPU, our implementation generates candidates 4.34 times faster

Table 3: Generator size and cached-feature generation cost. Parameter counts exclude protein feature extractors. Time and peak PyTorch-allocated GPU memory are normalized to our model (1.00) within each comparison panel. Baseline entries summarize six paired targets as geometric means of ratios of within-target trial medians, with 95% target-bootstrap intervals. Lower relative costs are better. ProtoBind-Diff timing includes per-call inference setup and model device placement. Measurements use the tested native implementations.

Model	Generator parameters (millions) ↓	Relative time ↓ (ours = 1)	Relative peak memory ↓ (ours = 1)
Ours (reference)	6.85	1.00	1.00
PCMol	101.34	4.34 [3.91, 4.77]	8.62 [8.09, 9.22]
ProtoBind-Diff	159.53	6.13 [5.39, 6.63]	9.24 [8.79, 9.53]

than PCMol and 6.13 times faster than ProtoBind-Diff, using about one-ninth of their peak allocated GPU memory.

Each benchmark uses six targets from the corresponding comparison panel, with each measured call producing 25 outputs. We measure the native generation path through raw decoded strings, excluding protein-feature extraction and downstream scoring. Our model has lower median generation time and peak allocated memory on every benchmark target. Appendix F.4 describes the benchmark protocol, and Figure 5 shows the target-level measurements.

Candidate yield and practical use. Our model also returns more distinct valid, connected molecules with a budget of 25 generations per target. Across the full 42-target comparison panels, these molecules account for 90.95% of our model's outputs versus 86.10% of PCMol's, and 88.95% versus 65.33% in the ProtoBind-Diff comparison (Table 16). All outputs are processed by the same RDKit procedure (Landrum et al., 2026b). The percentages use all outputs as the denominator, including invalid and blank strings.

Released structure-generator outputs. An exploratory comparison with released Pocket2Mol (Peng et al., 2022) and TargetDiff (Guan et al., 2023) molecules also gives similar RF3 interface confidence. For each target and method, a fixed hash rule selects five valid, connected ligands for evaluation under common RF3 settings. The 12-target panel contains nine training targets and three validation targets for our model, and the upstream generation budgets differ across methods.

Even without binding-pocket coordinates as a generator input, our model produces candidates with mean RF3 interface confidence numerically close to that of the released pocket-conditioned outputs on this panel. This supports sequence-conditioned generation as a way to propose ligands before a binding-pocket structure is available, with complex structure prediction applied afterward.

Appendix G reports the target-level results, selection protocol, and separate docking coverage and preparation limits.

Taken together, the results show that our compact protein prefix provides useful target information for ligand generation. Our model achieves RF3 interface confidence comparable to PCMol and ProtoBind-Diff, produces more distinct valid candidates from the same generation budget, and uses less time and GPU memory in the cached-feature benchmarks. These advantages make it a practical generator for proposing ligands from protein sequences.

4 Discussion

4.1 Protein Conditioning and Design Choices

The matched–mismatched comparison supports using a compact protein prefix to guide ligand generation (Table 1). The prefix retains useful target information after residue features are reduced to 16 regional summaries. This allows the molecular decoder to use sequence-derived protein information without receiving one conditioning token per residue. Keeping ESM3 frozen also lets us reuse protein features across training and repeated sampling. The resulting workflow uses the molecular decoder to build candidate pools and reserves complex structure prediction for selected ligands.

4.2 Structural Evidence and External Comparisons

Selected generated ligands achieve higher mean RF3 interface confidence than molecular-weight-matched ligands measured against other proteins, while measured target binders score highest (Figure 2). With the protein and alignment held fixed, this ordering supports greater predicted compatibility of the generated ligands with the intended target than of the off-target controls. On the 12-target public-output panel, selected ligands from our model also achieve mean RF3 interface confidence comparable to Pocket2Mol and TargetDiff under common rescoring (Appendix G).

On the tested panels, the compact generator combines comparable RF3 interface confidence with lower generation cost. The lower time and memory requirements of its cached-feature implementation leave more resources for additional sampling or downstream structural evaluation (Tables 2 and 3). Its higher yield of distinct valid, connected molecules also means that the same number of generations provides more candidates for screening (Table 16). These results support the model as an efficient first stage for target-directed ligand design.

4.3 Limitations and Future Work

RF3 ipTM measures predicted interface confidence rather than binding affinity. In the three-arm comparison, the off-target controls are not verified non-binders, and molecular-weight matching leaves other chemical differences uncontrolled. The affinity oracles combine labels from different assay endpoints and also select candidates, so their scores are not an independent validation of the selected molecules. Biochemical assays against intended and control targets are needed to establish binding affinity and selectivity. The docking analysis of released outputs also requires more complete coverage and reliable native-ligand redocking before supporting a broader comparison (Appendix G).

The structural comparisons cover selected target panels, and the nominal test split was consulted during model development (Section 3.1). Cluster-disjoint splitting does not exclude all remote homology or exposure during protein-encoder pretraining. All 42 targets in the ProtoBind-Diff comparison have homologs in its public training data. Overlap with PCMol's training data remains unverified (Appendix F). The 12 targets used for public-output rescoring belong to our training or validation splits (Appendix G). Evaluation on targets unused during model development and repetition across generation seeds would establish how well the observed advantages extend beyond these panels.

The prefix intervention tests the use of target information rather than the contribution of each architectural component. We chose $K = 16$ under our training budget. The reduced-budget pilots do not establish an optimal token count, and the predecessor comparison changes more than the pooling scheme (Appendices B.2 and B.3). The affinity sweep shows a response in oracle-predicted pAff (Appendix D.2), while the RF3 controls leave the structural benefit of the requested value unresolved (Appendix D.3). The masking control changes that value while retaining the token position and projection bias, so it does not test removal of the affinity token. We also did not compare ensemble ranking with single-oracle or random selection.

The generation-cost benchmark covers six targets per baseline with cached protein features. It compares native implementations that retain different precision and inference settings. Generator parameter counts exclude the protein encoders, and the measured costs exclude protein-feature extraction, training, and downstream scoring (Appendix F.4). Feature extraction, residue projection, and pooling still depend on protein length even though the decoder receives a fixed number of protein tokens. Full-workflow benchmarks across protein lengths and comparisons with pocket-conditioned generators would clarify the broader efficiency and scaling of this approach.

5 Conclusion

We developed a compact autoregressive ligand generator that connects frozen ESM3 residue features to a molecular decoder through a short protein prefix, without requiring a binding-pocket structure. Selected ligands generated using the intended protein's prefix achieve higher mean RF3 interface confidence than those generated using another protein's prefix. On their respective comparison panels, our model achieves RF3 interface confidence comparable to PCMol and ProtoBind-Diff with a substantially smaller generator. Its implementation uses less generation time and peak allocated

GPU memory in cached-feature benchmarks, and its candidate pools contain more distinct, valid, connected molecules for the same number of generations. These results support a compact sequence-conditioned generator for building target-directed candidate pools before structural evaluation.

Ethics Statement

This computational study uses public protein–ligand records and involves no human participants or animal experiments. Generated molecules and predicted complexes are hypotheses for research, not validated therapeutics. They should not be used for clinical, environmental, or biological deployment without appropriate synthesis, safety, activity, and dual-use review.

Reproducibility Statement

Sections 2 and 3 describe the model and the main experiments. The appendices detail the dataset and preprocessing (Appendix A), training and pooling choices (Appendix B), affinity oracles and candidate selection (Appendix C), conditioning experiments (Appendix D), and three-arm structural evaluation (Appendix E). Appendix F reports the public pretrained generator comparisons, including candidate yield and generation cost. Appendix G describes the common evaluation of released Pocket2Mol and TargetDiff ligands.

AI Use Statement

Anthropic Claude was used to implement and debug evaluation and analysis codes. Anthropic Claude also assisted with the preparation of Figure 1. OpenAI ChatGPT was used for manuscript organization, and editing. No synthetic data or reported result was generated by either agent. All reported quantitative results were computed from recorded training, generation, oracle, and folding runs. The authors reviewed the assisted code and text, checked quantitative claims against run artifacts, and take responsibility for the paper.

References

Viraj Bagal, Rishal Aggarwal, P. K. Vinod, et al. 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.

Andrius Bernatavicius, Martin Šícho, Antonius P. A. Janssen, et al. AlphaFold meets de novo drug design: Leveraging structural protein information in multitarget molecular generative models. *Journal of Chemical Information and Modeling*, 64(21):8113–8122, 2024. doi: 10.1021/acs.jcim.4c00309.

G. Richard Bickerton, Gaia V. Paolini, Jérémy Besnard, et al. Quantifying the chemical beauty of drugs. *Nature Chemistry*, 4(2):90–98, 2012. doi: 10.1038/nchem.1243.

Nathaniel Corley, Simon Mathis, Rohith Krishna, et al. Accelerating biomolecular modeling with AtomWorks and RF3. *bioRxiv*, 2025. doi: 10.1101/2025.08.14.670328. Preprint.

Orion Dollar, Nisarg Joshi, Jim Pfaendtner, et al. Efficient 3D molecular design with an E(3) invariant transformer VAE. *The Journal of Physical Chemistry A*, 127(37):7844–7852, 2023. doi: 10.1021/acs.jpca.3c04188.

Jerome Eberhardt, Diogo Santos-Martins, Andreas F. Tillack, et al. AutoDock Vina 1.2.0: New docking methods, expanded force field, and Python bindings. *Journal of Chemical Information and Modeling*, 61(8):3891–3898, 2021. doi: 10.1021/acs.jcim.1c00203.

Peter Ertl and Ansgar Schuffenhauer. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of Cheminformatics*, 1:8, 2009. doi: 10.1186/1758-2946-1-8.

Jens Glaser. jglaser/binding_affinity. Hugging Face dataset repository, n.d. URL https://huggingface.co/datasets/jglaser/binding_affinity. Accessed September 24, 2026.

Jiaqi Guan, Wesley Wei Qian, Xingang Peng, et al. 3D equivariant diffusion for target-aware molecule generation and affinity prediction. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=kJqXEPXMsE0.

Thomas Hayes, Roshan Rao, Halil Akin, et al. Simulating 500 million years of evolution with a language model. *Science*, 387(6736):850–858, 2025. doi: 10.1126/science.ads0018.

Minh Hoang and Mona Singh. Locality-aware pooling enhances protein language model performance across varied applications. *Bioinformatics*, 41(Supplement 1):i217–i226, 2025. doi: 10.1093/bioinformatics/btaf178.

Kexin Huang, Tianfan Fu, Lucas M. Glass, et al. DeepPurpose: A deep learning library for drug–target interaction prediction. *Bioinformatics*, 36(22–23):5545–5547, 2020. doi: 10.1093/bioinformatics/btaa1005.

Greg Landrum, Paolo Tosco, Brian Kelley, et al. rdkit/rdkit: 2025_09_6 (Q3 2025) Release. Zenodo software release, 2026a. doi: 10.5281/zenodo.18797641. RDKit version 2025.09.6.

Greg Landrum, Paolo Tosco, Brian Kelley, et al. rdkit/rdkit: 2026_03_4 (Q1 2026) Release. Zenodo software release, 2026b. doi: 10.5281/zenodo.21291217. RDKit version 2026.03.4.

Lukia Mistryukova, Vladimir Manuilov, Konstantin Avchaciov, et al. ProtoBind-Diff: A structure-free diffusion language model for protein sequence-conditioned ligand design. *bioRxiv*, 2025. doi: 10.1101/2025.06.16.659955. Preprint.

Xingang Peng, Shitong Luo, Jiaqi Guan, et al. 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, 2022. URL https://proceedings.mlr.press/v162/peng22b.html.

Alexander Rives, Joshua Meier, Tom Sercu, et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, 2021. doi: 10.1073/pnas.2016239118.

Rohit Singh, Samuel Sledzieski, Bryan Bryson, et al. Contrastive learning in protein language space predicts interactions between drugs and protein targets. *Proceedings of the National Academy of Sciences*, 120(24):e2220778120, 2023. doi: 10.1073/pnas.2220778120.

Martin Steinegger and Johannes Söding. 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.

Oleg Trott and Arthur J. Olson. 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.

Julia K. Varga, Sergey Ovchinnikov, and Ora Schueler-Furman. actifpTM: A refined confidence metric of AlphaFold2 predictions involving flexible regions. *Bioinformatics*, 41(3):btaf107, 2025. doi: 10.1093/bioinformatics/btaf107.

Ashish Vaswani, Noam Shazeer, Niki Parmar, et al. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Minjie Wang, Da Zheng, Zihao Ye, et al. Deep graph library: A graph-centric, highly-performant package for graph neural networks. *arXiv preprint arXiv:1909.01315*, 2019. doi: 10.48550/arXiv.1909.01315. URL https://arxiv.org/abs/1909.01315v2.

David Weininger. SMILES, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, 1988. doi: 10.1021/ci00057a005.

Kehan Wu, Yingce Xia, Pan Deng, et al. TamGen: Drug design with target-aware molecule generation through a chemical language model. *Nature Communications*, 15:9360, 2024. doi: 10.1038/s41467-024-53632-4.

Zhaoping Xiong, Dingyan Wang, Xiaohong Liu, et al. Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism. *Journal of Medicinal Chemistry*, 63(16):8749–8760, 2020. doi: 10.1021/acs.jmedchem.9b00959.

A Protein–Ligand Dataset and Preprocessing

A.1 Data Source and Composition

We use the public `jglaser/binding_affinity` collection hosted on Hugging Face (Glaser, n.d.). The collection combines records from BindingDB, PDBbind-cn, BioLiP, and Binding MOAD. We prepare the training and evaluation data from this collection using the released protein sequences, canonical ligand SMILES, and affinity labels.

The processed dataset contains $1{,}836{,}729$ records covering $22{,}974$ distinct protein sequences and $980{,}310$ distinct ligand SMILES. A record contains one protein sequence, one ligand SMILES string, and one affinity label. Multiple records can describe the same protein–ligand pair. Table 4 gives the composition of each split.

Across distinct protein sequences, the median length is 313 residues, with an interquartile range of 224 to 456 and a full range of 9 to $7{,}182$. These lengths describe the stored sequences before truncation for model input. Ligand counts use distinct canonical SMILES strings from the processed collection.

Table 4: Dataset composition under the protein-cluster split. Protein counts use distinct amino-acid sequences, and ligand counts use distinct canonical SMILES strings. Records include repeated protein–ligand pairs. Ligands can occur in more than one split, so the total ligand count is the size of their union.

Split	Records	Protein sequences	Ligand SMILES
Training	1,363,657	17,616	761,556
Validation	254,903	2,332	163,262
Test	218,169	3,026	136,466
Total	1,836,729	22,974	980,310

A.2 Preprocessing and Protein-Cluster Splits

Our preprocessing selects the protein sequence, canonical SMILES, and negative-log molar affinity fields from the public collection. The processed records contain nonempty sequence and SMILES strings and finite affinity labels. Repeated records are retained as separate observations.

We cluster protein sequences with MMseqs2 at a 30% sequence-identity threshold (Steinegger & Söding, 2017) and assign whole clusters to training, validation, or test sets. All records for a protein therefore remain in the same split. The partition contains $3{,}953$ training clusters, 493 validation clusters, and 493 test clusters, with no cluster shared across splits. The split is defined by protein clusters, and ligand structures can overlap across partitions.

The generator and final affinity oracles use the same fixed split assignments. Oracle development under protein-random and protein-cluster splits is reported in Appendix C. For the generator, cached residue features retain at most $2{,}048$ residues per protein, as described in Section 2.

A.3 Affinity Labels and Repeated Records

The affinity label is expressed on a negative-log molar scale:

$$a = -\log_{10}(\text{affinity in M}) = 6 - \log_{10}(\text{affinity in } \mu\text{M}).$$

Higher values correspond to lower reported affinity concentrations. We use these labels as pAff proxies because the processed records combine assay endpoints without retaining the endpoint type for each record. The oracles predict these values, and the generator uses them as affinity requests.

Grouping records by exact protein sequence and canonical SMILES gives 1,569,572 distinct protein–ligand pairs. Of these, 188,759 pairs occur more than once. For each repeated pair, we calculate the range between its largest and smallest pAff labels. The median range is 0.62 pAff units, and the 90th percentile is 2.23 (Table 5). Repeated labels can reflect differences in assay endpoints and conditions as well as measurement variation.

Table 5: Repeated records in the processed dataset. A pair is defined by an exact protein sequence and canonical SMILES string. The repeated-pair percentage uses all distinct pairs as its denominator. The repeated-record percentage uses all dataset records. Label ranges are calculated within pairs that occur at least twice.

Metric	**Value**
Distinct protein–ligand pairs	1,569,572
Pairs with at least two records	188,759 (12.0%)
Records belonging to repeated pairs	455,916 (24.8%)
Maximum records for one pair	94
Median pAff range among repeated pairs	0.62
90th percentile of pAff range among repeated pairs	2.23

B Model Training and Pooling Choices

B.1 Generator Configuration and Training

The generator combines frozen ESM3 residue features with a causal molecular decoder. Table 6 gives the configuration used for the full protein-conditioned model. We train the decoder, conditioning projections, and layer normalization parameters while keeping the cached ESM3 features fixed. The checkpoint is selected by validation loss, with the evaluated checkpoint taken from epoch 6.

Table 6: Configuration of the protein-conditioned generator.

Setting	Value
Transformer layers	8
Attention heads	8
Hidden width	256
Feed-forward width	1,024
Activation	GELU
Protein-prefix tokens	16
Affinity tokens	1
Molecular tokenizer	MolGPT-style SMILES tokenizer
Optimizer	AdamW
Batch size	256
Peak learning rate	6×10^{-4}
Learning-rate schedule	Token-based warmup and cosine decay
Training schedule	30-epoch horizon
Evaluated checkpoint	Epoch 6, selected by validation loss

Affinity requests are standardized using the training-label mean and standard deviation. The shifted next-token loss is computed over molecular positions, including padding positions. Conditioning

positions are excluded from this loss. When a sequence is shorter than the number of pooling segments, an empty segment contributes a zero vector before output layer normalization, following the denominator in Equation 2.

B.2 Pooling Development

Reduced-budget pilots compared learned-query pooling with $K = 1$, 16, and 64. Each run used at most 25 pairs per protein and a ten-epoch budget. No token count consistently led across the pilot metrics. The learned query tokens also became nearly identical in the inspected runs, motivating fixed, disjoint sequence segments that summarize different parts of the protein.

The fixed-segment pilots showed some improvement in validity under the tested settings. We selected $K = 16$ to provide several regional summaries while keeping the decoder's protein prefix short. These pilots informed the implementation choice but did not compare fully trained models across multiple seeds. The nominal test split was consulted during this development process.

B.3 Comparison with a Single-Token Predecessor

The current K16 system has slightly higher mean ipTM than an earlier single-token system on both comparison panels (Table 7). The paired intervals include zero, so the comparison does not establish a K16 advantage. The systems also differ in affinity fusion and training implementation. This comparison therefore evaluates the two systems as implemented, rather than isolating the pooling operation.

Panels L and K contain 42 and 82 targets selected from the systems' respective candidate pools. They share 12 targets, giving 112 distinct targets in total. Their results are consequently not independent replications.

Table 7: Single-token predecessor and K16 system comparison. Values are mean RF3 ipTM. Differences are paired K16-minus-predecessor means with 95% bootstrap intervals.

Panel	Targets	Predecessor	K16	Difference [95% CI]
L	42	0.9139	0.9259	0.0120 [−0.0053, 0.0303]
K	82	0.9073	0.9140	0.0066 [−0.0064, 0.0198]

C Affinity Oracles and Candidate Selection

C.1 Affinity Prediction under Protein-Cluster Separation

Affinity prediction was less accurate on the protein-cluster split than on the protein-random split in our development experiments. The DeepPurpose model with Morgan ligand fingerprints and a CNN protein encoder achieved test Spearman correlation of 0.671 under the protein-random split and 0.358 under the cluster split (Table 8). This difference motivated further oracle development for ranking candidates on proteins from separate clusters.

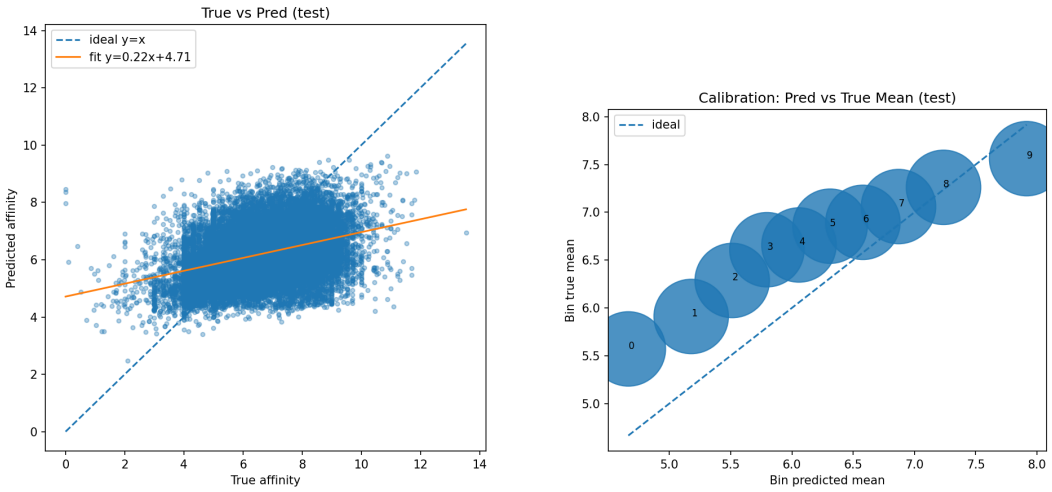
The earlier Morgan/CNN model retained a positive association with observed pAff on the cluster split, but its predictions covered a narrower range than the observed labels (Figure 3). Its mean prediction error, defined as predicted minus observed pAff, was -0.454. Mean observed pAff increased across predicted-score deciles, with underprediction in nine of the ten deciles. These diagnostics describe the Morgan/CNN development model. The final DeepPurpose oracle uses the AttentiveFP ligand encoder described below.

C.2 Final Affinity Oracles

To learn molecular features from atom and bond connectivity, we replace the fixed Morgan ligand representation with an AttentiveFP graph neural network while retaining the CNN protein encoder (Huang et al., 2020; Xiong et al., 2020; Wang et al., 2019). The resulting model achieves test Spearman correlation of 0.395 on the cluster split, compared with 0.358 for Morgan/CNN. Its test

Table 8: Oracle development results on test records. Random denotes the protein-random split, and Cluster denotes the 30% MMseqs2 protein-cluster split. n is the number of evaluated records. Final marks the two models used for candidate scoring. A dash indicates an unreported metric. These rows compare development runs and are not a controlled single-factor ablation.

Oracle	Split	n	Spearman ↑	Pearson ↑	RMSE ↓
DeepPurpose pretrained	Random	10,000	0.024	0.015	3.536
Morgan/CNN	Random	19,999	0.671	0.678	1.154
Morgan/CNN	Cluster	20,000	0.358	0.370	1.562
AttentiveFP/CNN (final)	Cluster	20,000	0.395	0.408	1.539
ConPLex pretrained	Cluster	3,000	0.038	—	—
ConPLex v2.0	Cluster	20,000	0.307	—	—
ConPLex v2.1	Cluster	20,000	0.324	—	—
ConPLex v2.2 (final)	Cluster	20,000	0.321	0.339	—



(a) Observed and predicted pAff

(b) Means within prediction deciles

Figure 3: Diagnostics for the earlier Morgan/CNN oracle on the protein-cluster test split. The affinity axes show pAff proxies. (a) Predictions have a narrower range than the observed labels. (b) Mean observed and predicted pAff for ten prediction deciles, each containing 2,000 records. The labels 0 to 9 identify the deciles. The final AttentiveFP/CNN oracle has separate results in Table 8.

Pearson correlation is 0.408, RMSE is 1.539, and MAE is 1.220. The selected checkpoint is from epoch 4 of a nine-epoch run and has the highest validation Pearson correlation, 0.3758.

We also adapt ConPLex to provide a second ranking signal from its drug–protein representations (Singh et al., 2023). An MLP predicts continuous pAff from the projected drug and protein features $\mathbf{d}$ and $\mathbf{p}$:

$$\widehat{a} = f_{\mathrm{MLP}}([\mathbf{d}, \mathbf{p}, |\mathbf{d} - \mathbf{p}|]).$$

The regression head has widths $3072 \to 512 \to 128 \to 1$. The training code combines mean squared error with an in-batch contrastive loss:

$$\mathcal{L}_{\mathrm{oracle}} = \lambda_r \mathcal{L}_{\mathrm{MSE}} + \lambda_c \mathcal{L}_{\mathrm{contrastive}}.$$

The archived v2.2 training script specifies $\lambda_r = 1.0$ and $\lambda_c = 0.2$. The original run log was unavailable for verification of these settings. The training code selects the checkpoint by validation Spearman correlation. The final v2.2 checkpoint achieves test Spearman 0.321 and Pearson 0.339. The v2.0 and v2.1 development runs achieve Spearman 0.307 and 0.324, respectively, placing the final model in the same performance range. Their diagnostic AUROCs are 0.683, 0.685, and 0.685 for v2.0, v2.1, and v2.2, using pAff ≥ 6.0 as the high-affinity label threshold. This threshold summarizes the regression scores and is not a verified binding classification.

AttentiveFP/CNN and ConPLex v2.2 are the two checkpoints used for oracle scoring throughout this paper. The arithmetic mean of their predictions defines the ensemble score in Equation 6. The protein-prefix experiment then tests the generator under the same candidate-scoring and selection procedure for matched and mismatched inputs (Section 3.2).

C.3 Candidate Selection and Diversity

For the original multi-target screen, matching the two oracle outputs and removing duplicate SMILES strings globally leaves $65{,}500$ candidates from $75{,}650$ generations. When a SMILES string occurs more than once, we retain the protein–ligand pair with the highest ensemble score. This count uses exact SMILES strings and differs from the count of unique valid canonical molecules in Section 3.1.

We initially select 100 candidates covering 85 proteins in descending ensemble-score order, allowing at most three candidates per protein and two per Murcko scaffold. Two candidates fail initial 3D preparation and are replaced by the next eligible candidates. The resulting 100 prepared complexes cover 84 proteins. The protein-prefix and external-generator comparisons instead select one candidate per target and condition.

The oracle outputs have Spearman correlation 0.618 and Pearson correlation 0.608 across the scored pool. Without the selection caps, the ensemble top-100 differs from the DGL-only top-100 by 79 molecules. The two predictors therefore produce materially different candidate rankings.

The original 100-candidate shortlist covers 100 distinct Murcko scaffolds and has mean nearest-neighbor ECFP4 Tanimoto similarity 0.375 (Table 9). At a similarity threshold of 0.7, single-linkage clustering leaves 94 of the 100 molecules as singletons, with no cluster larger than two molecules (Table 10). These statistics describe the chemical diversity of the selected set.

Table 9: Composition of the original 100 selected complexes. Nearest-neighbor similarity is calculated within the selected ligand set using ECFP4 Tanimoto similarity.

Property	Value
Distinct proteins	85
Distinct Murcko scaffolds	100
Mean ensemble score	8.454
Minimum ensemble score	8.313
Mean nearest-neighbor similarity	0.375
Molecules with nearest-neighbor similarity ≥ 0.5	23%
Molecules with nearest-neighbor similarity ≥ 0.7	6%

Table 10: Single-linkage clusters in the original selected set at three ECFP4 Tanimoto similarity thresholds. Largest gives the number of molecules in the largest cluster.

Similarity threshold	Clusters	Singletons	Largest
0.5	82	77	10
0.7	97	94	2
0.8	100	100	1

C.4 Molecular Properties and Selection Bias

We describe the generated molecules using molecular weight, QED (Bickerton et al., 2012), and synthetic accessibility score (Ertl & Schuffenhauer, 2009). A random sample of $30{,}000$ nonblank generated records, selected with seed 42, yields $26{,}657$ molecules with computable descriptors. We also calculate descriptors for distinct measured reference SMILES from the target table, giving $2{,}574$ computable molecules. Table 11 reports these two analysis subsets.

Table 11: Molecular properties in the descriptor-analysis subsets. MW, QED, and SA are medians. Higher QED indicates greater drug-likeness, and lower SA indicates greater predicted synthetic accessibility. $\rho_{\mathrm{MW,QED}}$ is the within-group Spearman correlation.

Group	n	MW (Da)	QED	SA	$\rho_{\mathrm{MW,QED}}$
Generated	26,657	410.7	0.483	3.01	-0.656
Measured references	2,574	438.4	0.437	3.47	-0.722

QED decreases with molecular weight in both subsets. A QED threshold of 0.5 would exclude 60.1% of the measured references, so we use these descriptors to characterize outputs without imposing that threshold as a candidate filter.

Effect of molecular weight on candidate selection. The selected top-100 has median molecular weight 531.2 Da, compared with 412.5 Da in the scored pool (Table 12). Ensemble score and molecular weight have Spearman correlation 0.447 across the pool, showing that the selection procedure favors larger molecules.

Table 12: Molecular weight before and after candidate selection. P90 is the 90th percentile.

Set	Median MW (Da)	P90 MW (Da)	MW ≥ 700 Da
Scored pool ($65{,}500$)	412.5	579.1	4.0%
Selected top-100	531.2	640.6	5.0%

To examine this size effect, we compare the selected set with uniformly sampled reference sets and with sets matched on molecular weight. For each selected molecule, an MW-matched replacement is drawn from pool molecules within ± 25 Da. The mean ensemble-score gap is 2.262 pAff units against uniform sets and 1.896 against MW-matched sets, a reduction of about 16%. This comparison describes how molecular weight affects the score distribution after selection. The three-arm RF3 experiment separately controls molecular weight when pairing generated ligands with off-target measured ligands (Appendix E).

D Conditioning Experiments

D.1 Protein-Prefix Control

The protein-prefix control compares ligands generated using the intended target's prefix with ligands generated using another protein's prefix. Both candidate sets are scored and structurally evaluated against the intended target. The panel contains the 42 proteins with the smallest unsigned sequence hashes among the 112 distinct targets in the predecessor comparison (Appendix B.3). Donor prefixes

are drawn from the 3,026 test proteins, excluding the intended target's cluster. Donors are assigned without reuse by nearest sequence length, with hash-based tie-breaking.

Each target and condition has 25 generations using seed 42, temperature 1.0, and top-$k = 50$ sampling. Invalid and blank outputs remain in the generation count. Among candidates with both oracle scores, we select the molecule with the highest mean score against the intended target. Ties are resolved by canonical isomeric SMILES and source-row order. This selection is completed before RF3 evaluation.

RF3 uses the intended target sequence and the same target MSA for both conditions. The MSA query sequence is checked against the target, and missing or mismatched alignments stop the run. Inference uses seed 0 and five samples in a batch of five. The primary result uses the sample selected by RF3 ranking score. We also average all five samples for each complex.

Paired analysis. The target is the analysis unit. RF3 95% intervals use 10,000 paired target-bootstrap replicates with base seed 20260904 and deterministic streams for each comparison and metric. Recorded Wilcoxon tests are unadjusted. The RF3 analysis specification was fixed before this execution, following earlier pilot and failure observations. Missing scores exclude a target only from the affected paired comparison and are reported with that comparison.

The oracle-score comparison uses the stored selected-candidate manifests. All 42 matched records have identical target sequences and requested affinity values across conditions. Targets are ordered by unsigned sequence hash, and the 95% percentile interval uses 10,000 target-bootstrap resamples with NumPy's default random generator and seed 20260920. This oracle analysis was added during revision using existing candidate scores.

Agreement across RF3 sample summaries. Averaging all five RF3 samples gives mean ipTM 0.9194 for matched pairs and 0.8275 for mismatched pairs. The paired difference is 0.0919, with a 95% interval of $[0.0620, 0.1232]$, and matched pairs score higher on 39 of 42 targets. This agrees with the direction and magnitude of the ranking-selected sample results in Table 1.

D.2 Within-Protein Response to Requested Affinity

Increasing the requested affinity raises the oracle-predicted pAff for most proteins in a separate K16-checkpoint experiment. We hold 300 proteins fixed and request pAff values of 5, 6, 7, 8, and 9, with 25 generations for each protein and value. This produces 37,500 generations. The training-label mean and standard deviation used for request normalization are 6.4474 and 1.5643.

For each protein, we fit an ordinary least-squares slope for predicted pAff against requested pAff. We also fit the molecular weight response and the predicted-pAff response adjusted for molecular weight. Table 13 averages these slopes across proteins, with 95% intervals from 4,000 protein-level bootstrap replicates.

Table 13: Response to requested affinity in the 300-protein sweep. Slopes are means across proteins. MW slopes are in Da per requested pAff unit. The final column counts proteins with positive slopes.

Response	Mean slope	95% CI	Positive slopes
Predicted pAff	0.0749	[0.0693, 0.0808]	289/300
MW	12.7410	[11.3200, 14.2866]	264/300
Predicted pAff, adjusted for MW	0.0506	[0.0459, 0.0556]	274/300

The predicted-pAff slope is positive for 289 of 300 proteins and remains positive for 274 proteins after linear adjustment for molecular weight. Validity ranges from 87.21% to 89.99% across request settings, and uniqueness among valid outputs ranges from 99.23% to 99.50%. These results show a response in oracle-predicted pAff as the request changes while the protein remains fixed.

In the original scored pool, requested affinity and ensemble score have Spearman correlation 0.468 across 65,500 candidates and 0.558 after averaging by protein. That analysis changes protein identity and affinity request together. The within-protein sweep separates these two inputs and is the more direct test of the affinity response.

D.3 RF3 Controls for the Affinity Input

The affinity controls test whether the original requested value improves RF3 confidence after candidate selection. Masking sets the normalized affinity scalar to zero before projection while retaining its token position and projection bias. This is the same numerical input as a request at the training-label mean, not removal of the affinity token. Permutation uses the smallest cyclic shift of panel requests that changes every target's value while keeping its protein prefix fixed. Both controls follow the candidate-selection and RF3 procedure in Appendix D.1.

RF3 means remain close under these controls. For masking, mean ipTM of the ranking-selected sample is 0.9197 with the original value and 0.9098 with the masked value. For permutation, the paired means are 0.9201 and 0.9190. One permuted-value ligand fails geometry preparation after 34 rooted-SMILES preparation attempts, leaving 41 complete pairs. The paired intervals include zero under both RF3 summaries (Table 14). These controls do not show a consistent RF3 benefit from the original affinity requests. The within-protein sweep above establishes a response in oracle scores, which is a different outcome from RF3 confidence or measured binding affinity.

Table 14: Affinity-value controls. Differences are original-input minus control ipTM. Ranking-selected uses the RF3 sample with the highest ranking score. Five-sample mean averages the same five samples. Wins count targets with higher ipTM for the original input. Differences are computed before rounding.

Control	RF3 summary	Paired n	Difference [95% CI]	Wins
Masked value	Ranking-selected	42	0.0099 [−0.0029, 0.0240]	21/42
Masked value	Five-sample mean	42	0.0100 [−0.0026, 0.0243]	22/42
Permuted value	Ranking-selected	41	0.0010 [−0.0075, 0.0105]	24/41
Permuted value	Five-sample mean	41	0.0011 [−0.0078, 0.0102]	24/41

E Three-Arm Structural Evaluation

E.1 Matched Controls and RF3 Evaluation

The three-arm experiment compares generated ligands with measured target binders and off-target measured ligands on the same protein. For proteins with several selected generated ligands, we use the first ligand in the oracle-based selection order. Off-target controls are drawn from $136{,}466$ distinct test ligands, using ligands measured against proteins in a different homology cluster and matching molecular weight to the generated ligand. Across the 84 assembled triplets, the median absolute molecular-weight difference between generated ligands and off-target controls is 0 Da, and the maximum is 0.052 Da.

Each triplet uses the same protein sequence and MSA across arms. RF3 produces five samples per complex with diffusion batch size five. The original runs use A100 GPUs for proteins of at most $1{,}200$ residues and A40 GPUs for longer proteins. Measured-target-binder RF3 outputs are available for 82 of the 84 proteins. We analyze these 82 complete triplets, with no reported clash flags.

Generated ligands have higher mean ipTM than off-target controls by 0.0521, with a paired 95% bootstrap interval of $[0.0341, 0.0693]$. Measured target binders have higher mean ipTM than generated ligands by 0.0317, with an interval of $[0.0194, 0.0451]$. Table 15 reports both ipTM and contact-restricted interface PAE for these same proteins.

E.2 Contact-Restricted Interface PAE

Contact-restricted PAE gives the same arm ordering as ipTM: measured target binders have the lowest mean error, followed by generated ligands and then off-target controls (Figure 4). Generated ligands have lower PAE than off-target controls on 64 of 82 proteins, with a mean difference of 0.925 Å.

We calculate this metric to describe confidence near the predicted ligand contact. A median of 8.2% of protein residues lie within 5 Å of the ligand in the analyzed complexes. Contact-aware confidence

Table 15: RF3 confidence for the three ligand arms on the same 82 proteins. Arm rows report mean $\pm$ sample standard deviation. Comparison rows give paired mean differences and the number of proteins favoring the first arm. Differences are first minus second for ipTM and second minus first for iPAE, so positive values favor the first arm. p denotes a two-sided Wilcoxon signed-rank test. Off-target measured controls are not verified non-binders.

Arm or comparison	ipTM ↑	Contact-5 Å iPAE ↓
Measured target binders	0.9456 ± 0.0323	2.289 ± 1.363
Generated ligands	0.9140 ± 0.0511	3.184 ± 1.790
Off-target measured controls	0.8619 ± 0.0694	4.110 ± 1.463
Target binders vs. generated	0.0317 (69/82)	0.895 (70/82)
Wilcoxon p	6.1×10^{-10}	2.0×10^{-9}
Generated vs. off-target controls	0.0521 (61/82)	0.925 (64/82)
Wilcoxon p	9.2×10^{-8}	3.8×10^{-7}

measures have also been studied for AlphaFold2 protein interfaces (Varga et al., 2025). For our RF3 analysis, we identify protein residues with an atom within 5 Å of a ligand atom and average the corresponding protein-row, ligand-column entries of the PAE matrix. This is our contact-restricted summary of RF3 PAE. The contact fraction describes the predicted geometry and does not specify how much those residues contribute to ipTM.

Mean contact-restricted PAE is 2.289 Å for measured target binders, 3.184 Å for generated ligands, and 4.110 Å for off-target controls. The measured-target-binder advantage over generated ligands is 0.895 Å, with lower values on 70 of 82 proteins. Both PAE and ipTM summarize RF3 confidence, so their agreement is consistency within the structural evaluation, not an independent measurement of binding affinity.

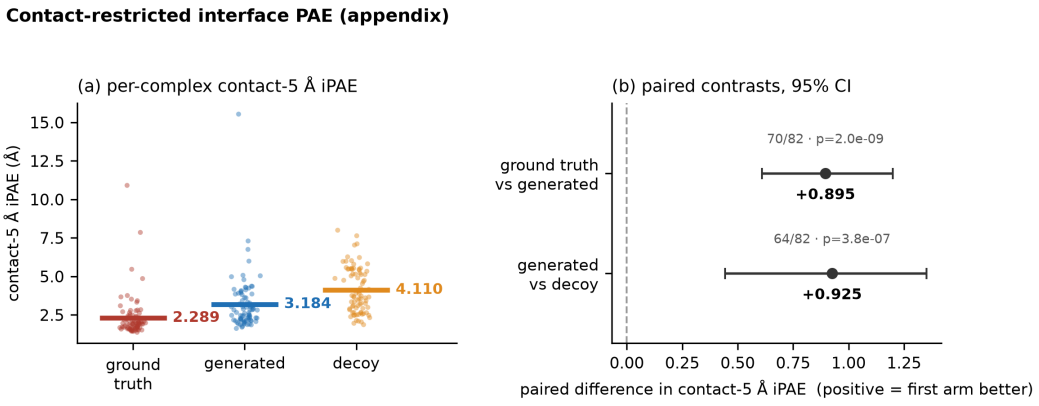


Figure 4: Contact-restricted interface PAE for the same 82 proteins as Figure 2. Lower values indicate greater local structural confidence. The existing figure labels "ground truth" and "decoy" denote measured target binders and off-target measured controls, respectively. (a) Per-complex values and arm means. (b) Paired differences with bootstrap 95% intervals.

F Public Pretrained Generator Comparisons

We compare our generator with the public PCMol (Bernatavicius et al., 2024) and ProtoBind-Diff (Mistryukova et al., 2025) implementations using their released checkpoints without retraining. Each comparison uses a separate 42-target panel and applies the same candidate selection and RF3 evaluation to both generators. We also compare candidate yield across the full panels and generation cost on six targets from each panel.

F.1 Models and Comparison Panels

PCMol conditions generation on structure-derived AlphaFold representations. Its panel contains 42 test targets with released accession-indexed representations. We retain two targets with existing RF3 evaluations and select 40 additional targets using the fixed sequence-hash rule. The panel therefore reflects the availability of public inputs. Absence of overlap with PCMol's training data could not be verified.

ProtoBind-Diff combines ESM2 protein-sequence features with a molecular diffusion language model. It uses the 42-target protein-prefix control panel. All 42 targets have homologs in its public training data under our overlap audit. These targets are held out from our full generator training, but training exposure is not symmetric across the compared models.

The PCMol code is pinned to commit

`6e2bd12bd7455e2a81c57032f2ef46fe2d30e8c5`

and uses checkpoint `XL/model_7.pkg`, with SHA-256 digest

`4646a3af2ad4f01f87f8433e95c40e2487a2b7110f952f51286086e53a59da02`

The ProtoBind-Diff code is pinned to commit

`6bebe8915d9e3d94de0da6a61bb52dba6d50e48b`

and uses `model.ckpt`, with SHA-256 digest

`790b6e8eee912b58a4aadab6fa83883fd1cb5bfafde952a95a31beb90625feb0`

The implementations run in pinned environments with compatibility adjustments. ProtoBind-Diff uses ESM2-t33-650M-UR50D features, 250 diffusion steps, nucleus parameter 0.9, noise parameter $\eta = 0.1$, and maximum ligand length 170.

F.2 Candidate Selection and RF3 Evaluation

Each model produces 25 generations per target, giving 1,050 outputs per arm. We score candidates against the intended target with the two affinity oracles and select one ligand using the highest ensemble score. All arms provide a scored candidate for every target. Selection and RF3 inference follow Appendix D.1.

Our model achieves comparable mean RF3 interface confidence to both baselines on their respective panels (Table 2). The primary comparison uses the RF3 sample with the highest ranking score. We compute paired 95% percentile intervals from 20,000 target-bootstrap replicates with seed 20260915. Averaging all five RF3 samples within each complex gives the same overall result. The mean ours-minus-baseline difference is -0.0118 for PCMol, with a 95% interval of $[-0.0451, 0.0199]$, and 0.0084 for ProtoBind-Diff, with an interval of $[-0.0068, 0.0227]$. These comparisons describe observed RF3 confidence under the shared evaluation procedure. No statistical equivalence margin was specified.

F.3 Candidate Yield under a Common Validity Procedure

Our model produces more distinct valid, connected molecules per generation on both comparison panels (Table 16). Their fraction is 4.86 percentage points higher than PCMol's and 23.62 points higher than ProtoBind-Diff's. We measure these yields from the original four generation pools, which contain 4,200 outputs in total. The separately timed calls below are not part of these pools.

RDKit 2026.03.4 (Landrum et al., 2026b) parses each raw string and applies default sanitization. A valid output must contain at least one atom. Blank strings and invalid outputs remain in the denominator. We canonicalize valid molecules as isomeric SMILES and count distinct strings within each target's 25 generations. A connected molecule has one RDKit fragment. Salts are not stripped. We apply this common procedure to all models because their upstream validity flags use different criteria.

We average target-level fractions and bootstrap paired targets with 20,000 replicates and seed 20260915. The intervals are exploratory and are not adjusted for multiple related metrics. An

Table 16: Candidate yield from the original generation pools. Each comparison contains 42 targets and 25 generations per target and model. Ours and Baseline are percentages of all generations, including blank and invalid outputs. Differences are ours minus baseline in percentage points (pp), with paired 95% target-bootstrap intervals. Unique counts distinct valid molecules within a target's 25 generations.

Metric ↑	Ours	Baseline	Difference (pp)	95% CI (pp)
PCMol comparison				
Valid	91.81	86.76	5.05	[1.14, 9.43]
Unique valid	91.81	86.10	5.71	[1.90, 10.10]
Unique valid, connected	90.95	86.10	4.86	[1.05, 9.24]
ProtoBind-Diff comparison				
Valid	89.24	65.33	23.90	[20.76, 27.14]
Unique valid	89.14	65.33	23.81	[20.67, 27.05]
Unique valid, connected	88.95	65.33	23.62	[20.48, 26.86]

independent analysis with RDKit 2025.09.6 (Landrum et al., 2026a) gives identical counts. Two canonical SMILES strings differ between versions, but each pair has the same Standard InChIKey. These metrics count usable molecular strings. Training-set novelty, scaffold diversity, and binding success are separate outcomes.

F.4 Cached-Feature Generation Benchmark

Our implementation uses less generation time and peak allocated GPU memory on every benchmark target (Figure 5). Table 3 summarizes the ratios. The benchmark measures native generation calls with cached protein features on the same physical NVIDIA A100-PCIE-40GB GPU.

Targets and repeated calls. Within each 42-target panel, we sort targets by protein length, divide them into three groups of 14, and select two targets per group using a fixed hash rule before timing. The two six-target subsets do not overlap. Protein lengths span 66–646 residues in the PCMol comparison and 125–297 residues in the ProtoBind-Diff comparison. For each model and target, we run one warm-up and three measured calls with deterministic paired seeds. Each call returns 25 generations in five batches of five, without replacing invalid outputs. The final allocation contains 24 warm-up calls and 72 measured calls, producing 600 and 1,800 outputs, respectively.

Timing and memory. Calls run sequentially. We synchronize CUDA before and after each timed call and reset peak memory statistics before the call. Timing ends after raw strings are decoded. Checkpoint loading, protein-feature extraction, explicit RDKit analysis, file writes, oracle scoring, and RF3 inference are excluded. Peak memory is `torch.cuda.max_memory_allocated`, including allocations already resident at the start of the call. It measures peak PyTorch allocation, not total GPU-process memory.

Implementation settings. Our generator uses FP32 with TF32 disabled in PyTorch 2.10.0 and CUDA 12.8. Sampling uses temperature 1.0, top-$k = 50$, and the original target-specific affinity requests. The checkpoint block size is 301, and decoding is capped at 300 steps. The sampler recomputes the projection, segment pooling, and cumulative molecular prefix at each step without a key–value cache. ESM3 residue features are cached, but the pooled prefix is not.

PCMol uses its native FP32 path in PyTorch 2.5.1 and CUDA 12.4, with `max_smiles_len`=102. Repeated AlphaFold-representation disk reads and post-generation RDKit analysis and CSV writing are outside the timer. Native DataParallel and string decoding remain inside. ProtoBind-Diff uses PyTorch 2.14.0, CUDA 13.0, and Lightning 16-mixed precision with the diffusion settings listed above. Cached ESM2 features use bfloat16. Each call creates a Lightning Trainer, so prediction setup and model device placement are included in its time. The ratios therefore compare these native implementations with their own precision and output-length settings, rather than GPU-resident decoders under identical runtime settings.

Parameter counts and aggregation. We count unique generator parameter objects in the evaluated checkpoints, including conditioning modules and excluding buffers, protein feature extractors, and optimizer state. The counts are 6,850,560 for our model, 101,340,995 for PCMol, and 159,530,279 for ProtoBind-Diff. The last count includes 1,280 non-trainable parameters. Protein-encoder size and feature-extraction cost are outside this generator comparison.

For each target and model, we take the median of the three measured calls separately for time and peak memory. We compute baseline-to-ours ratios for each target and summarize them with the geometric mean. Paired 95% percentile intervals use 20,000 target-bootstrap replicates with seed 20260915, averaging log ratios before exponentiation. All final-allocation measurements are retained. Targets are the comparison units, with three calls measuring repeatability within each target.

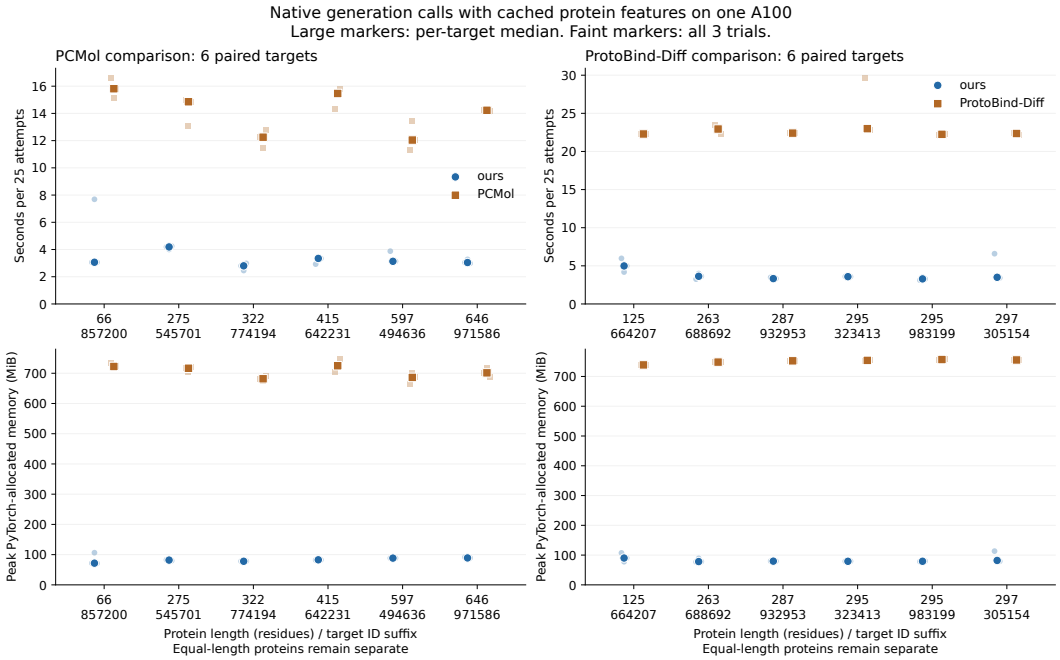


Figure 5: Generation time and peak PyTorch-allocated GPU memory with cached protein features on one A100. Columns show the separate six-target comparisons. Large markers are medians of three measured calls, and faint markers show individual calls. Each call produces 25 outputs. Targets are ordered by protein length and distinguished by sequence-hash suffixes. Horizontal positions are categorical. ProtoBind-Diff timing includes per-call setup and model device placement. Protein-feature extraction and downstream evaluation are excluded.

Benchmark records. The reported measurements come from one completed benchmark run. Technical failures interrupted an earlier run after 48 calls for our model, producing 1,200 outputs before the baseline measurements were completed. We retain those records but exclude them from the benchmark. At matching seeds, their output lists are identical to those from the completed run and are not additional independent samples.

The archived plotting script does not match its recorded checksum. The numerical summaries were independently recomputed from the archived measurements. The final figure groups records by target identity, keeping proteins of equal length separate.

G Evaluation of Released Pocket2Mol and TargetDiff Ligands

G.1 Released Molecules and Candidate Selection

We compare our generated ligands with Pocket2Mol (Peng et al., 2022) and TargetDiff (Guan et al., 2023) molecules released with ProtoBind-Diff (Mistryukova et al., 2025), using the repository commit in Appendix F. The comparison uses the released SMILES, with new structure preparation and common downstream evaluation. The original generator poses and published scores are not reused.

The 12 targets are CCR9, ESR1, FTO, GRIK1, HCRTR1, IDH1, JAK1, KDM1A, NR4A1, P2RX3, RIOK1, and SPIN1. Nine canonical sequences occur in our training data and three in validation data. None belongs to our 3,026-target test panel. The overlap audit distinguishes canonical-sequence matches from construct-level matches. This panel evaluates available molecules on targets already represented in our training or validation data.

Our model generates 1,000 outputs per target at requested pAff 7.0, temperature 1.0, and top-$k = 50$. The 12,000 outputs are generated in ten blocks of 100 per target, with block seeds $42 + 100000j$ for $j = 0, \ldots, 9$. For all three methods, RDKit 2026.03.4 retains valid, connected, unique canonical isomeric SMILES. We order candidates by a fixed SHA-256 hash of a shared salt, target name, and canonical SMILES. Selection uses no oracle score, novelty criterion, molecular-weight cutoff, or QED cutoff. The first 100 molecules per target and method are selected for docking, and the first five of those for RF3. The downstream selection budgets are equal, although the upstream generation budgets differ.

G.2 Common RF3 Evaluation

Our ligands achieve mean RF3 ipTM of 0.8368, compared with 0.8428 for Pocket2Mol and 0.8195 for TargetDiff (Table 17). Thus, generation without binding-pocket coordinates gives similar mean RF3 confidence to the released pocket-conditioned outputs on this panel.

Each target uses the same protein construct and query-verified MSA across methods, with MSA depths from 511 to 4,360. All 180 selected complexes complete five-sample RF3 inference, giving 900 samples. We select the sample with the highest RF3 ranking score for each complex, average ipTM over the five ligands for each target, and then average over targets. The 12 targets are the comparison units. Paired intervals use 10,000 target-bootstrap replicates with seed 220906.

Table 17: RF3 confidence after common rescoring of released Pocket2Mol and TargetDiff molecules and our generated molecules. Each entry averages five ligands for one target, using the RF3 sample with the highest ranking score for each ligand. Higher ipTM indicates greater interface confidence. Target mean weights the 12 targets equally. The panel contains nine training targets and three validation targets for our model.

Target	Ours	Pocket2Mol	TargetDiff
CCR9	0.7465	0.7629	0.7535
ESR1	0.9152	0.9348	0.8442
FTO	0.8692	0.9129	0.8810
GRIK1	0.8148	0.8761	0.8524
HCRTR1	0.9082	0.9121	0.8848
IDH1	0.7503	0.7564	0.7112
JAK1	0.9393	0.9203	0.9095
KDM1A	0.8331	0.8461	0.8642
NR4A1	0.7224	0.7530	0.7422
P2RX3	0.9033	0.8125	0.8018
RIOK1	0.8529	0.8778	0.8262
SPIN1	0.7864	0.7485	0.7628
Target mean	0.8368	0.8428	0.8195

The mean ours-minus-Pocket2Mol difference is -0.0060, with a 95% interval of $[-0.0258, 0.0178]$ and higher scores for our model on 3 of 12 targets. Against TargetDiff, the difference is 0.0173, with

an interval of $[-0.0042, 0.0404]$ and higher scores on 7 targets. This is an exploratory comparison of RF3 confidence on a training-exposed panel, not a held-out test of generalization or measured binding performance.

G.3 Docking Results and Preparation Checks

Docking provides a separate comparison, but incomplete coverage limits it to five paired targets against Pocket2Mol. No target has complete paired coverage against TargetDiff under the 100-of-100 rule (Table 18).

We use AutoDock Vina 1.2.5 (Trott & Olson, 2010; Eberhardt et al., 2021) with a rigid receptor, one CPU, seed 42, exhaustiveness 16, five modes, and energy range 3. The search box is a 20 Å cube centered on the reference ligand's heavy-atom center of mass. The first-mode score is the docking outcome, with lower values preferred.

Table 18: Docking coverage among 1,200 selected molecules per method. Complete targets have finite scores for all 100 selected molecules. Failed and not-run cases remain missing and are not assigned substitute scores.

Method	Finite scores	Failed	Not run	Complete targets
Ours	1,156	28	16	5/12
Pocket2Mol	1,196	4	0	11/12
TargetDiff	1,033	132	35	1/12

On the five complete paired targets, the mean Pocket2Mol-minus-ours score difference is -0.0761 kcal/mol, with a 95% interval of $[-0.5209, 0.5477]$. Our model has a lower mean docking score on one of these five targets. This result describes the complete subset only. The TargetDiff docking contrast remains unassessed because there are no complete paired targets.

Native-ligand redocking achieves symmetry-aware RMSD within 2 Å on 3 of 12 targets, with median RMSD 2.485 Å. RMSD is calculated without an additional alignment or minimization. Receptor preparation removes heterogens, including the magnesium ion in RIOK1. These preparation checks limit confidence in the docking scores as evidence of ligand quality. PoseBusters checks were not performed.

Preparation and runtime records. During execution, ligand preparation was changed to run in a separate process that could be terminated after 120 seconds. The cluster execution audit reports byte-identical accepted PDBQT inputs under the original and revised implementations for 3,448 preparations, including the 12 native ligands. The two implementations were not independently rerun during the subsequent local analysis.

The frozen record of runtime settings was rewritten after 93 RF3 complexes had completed, and its original version was not retained. The execution audit reports unchanged RF3 settings, but the missing original record prevents independent verification.