# *De novo* PROTAC design using graph-based deep generative models

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

PROteolysis TArgeting Chimeras (PROTACs) are an emerging therapeutic modality for degrading a protein of interest (POI) by marking it for degradation by the proteasome. Recent developments in artificial intelligence (AI) suggest that deep generative models can assist with the *de novo* design of molecules with desired properties, and their application to PROTAC design remains largely unexplored. We show that a graph-based generative model can be used to propose novel PROTAC-like structures from empty graphs. Our model can be guided towards the generation of large molecules (30 – 140 heavy atoms) predicted to degrade a POI through policy-gradient reinforcement learning (RL). Rewards during RL are applied using a boosted tree surrogate model that predicts a molecule's degradation potential for each POI. Using this approach, we steer the generative model towards compounds with higher likelihoods of predicted degradation activity. Despite being trained on sparse public data, the generative model proposes molecules with substructures found in known degraders. After RL, predicted activity against a challenging POI increases from 50% to >80% with near-perfect chemical validity for sampled compounds, suggesting this is a promising approach for the optimization of large, PROTAC-like molecules for targeted protein degradation.

## 1 Introduction

Of the 1,200 new molecular entities approved by the FDA between 1985-2021, small molecule drugs approved under a "New Drug Application (NDA)" comprised ∼80% of them, with the other 20% being new biological products [CDER, 2022]. Generally speaking, small molecules are designed to impede the function of biologically-relevant target proteins. Small molecule inhibitors interfere with their targets by accessing specific parts of the protein and binding strongly enough to affect their behavior. However, it is estimated that ∼75% of the human proteome lacks deep binding sites and is thus "undruggable" by traditional small molecule inhibitors [Toure and Crews, 2016]. These so-called undruggable targets are nonetheless implicated in a wide range of diseases, including cancer, autoimmune diseases, and cardio-metabolomic diseases, motivating the development of therapeutic modalities beyond small molecule inhibitors.

An example of such an "undrugabble" target is the BCL-2 protein, which regulates apoptosis, making it a prime target for cancer drug discovery [Frenzel et al., 2009]. However, in 2021, Lv et al. reported a potent BCL-xL and BCL-2 dual degrader with significantly improved antitumor activity against BCL-xL/2-dependent leukemia cells. This degrader belongs to a class of emerging therapeutic modalities called PRoteolysis TArgeting Chimeras, or PROTACs. There are currently 15 heterobifunctional

Submitted to NeurIPS 2022 AI for Science Workshop.

PROTACs in clinical development [Békés et al., 2022]. Generally speaking, their function is enabled by a three-component structure consisting of two binding domains and an organic linker (Figure 1). The two binding domains include a *warhead* designed to bind a protein of interest (POI), and an *E3 ligand* designed to bind an E3 ligase. In the ideal scenario, the linker anchors the two proteins together briefly, leading to ubiquitination of the POI, which marks it for degradation by the proteasome.

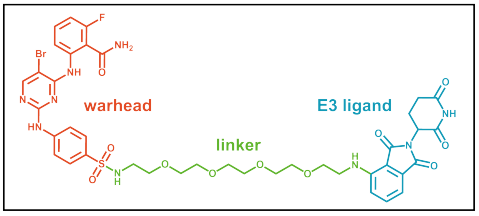


Figure 1: Example PROTAC structure from `protac-db` (PubChem CID: 155168919), highlighting the three general segments found in PROTACs: the warhead, linker, and E3 ligand.

The functionalities of each component are highly interdependent, such that rational design of PROTACs remains challenging. However, recent developments in artificial intelligence (AI) suggest that deep generative models (DGMs) can assist with the *de novo* design of molecules with desired pharmacological profiles [Elton et al., 2019]. While DGMs have been widely applied to the design of small molecule drugs, when it comes to PROTAC design DGMs are typically limited to optimization of the linker. Here, we introduce a graph-based DGM capable of designing PROTAC-like molecules atom-by-atom, starting from empty graphs. We show that the DGM learns to generate PROTAC-like structures containing many substructures found in known degraders, and we show quantitative improvement in a scalar model estimate of degradation activity against a POI. As the model was trained on sparse public data, we did not pursue experimental validation, but instead publish our workflow for PROTAC design open-source. We make the following contributions:

- a non-linear, boosted-tree-based model trained on public data for the prediction of protein degradation activity ($DC_{50}$) in PROTAC systems,
- application of an existing DGM to distribution-based learning tasks for PROTAC design (30–140 heavy atoms),
- application of policy-gradient reinforcement learning (RL) using a multi-objective scoring function to promote the design of structures with predicted protein degradation activity,
- a case-study where we apply the above three points to the *in silico* design of novel PROTAC-like structures for IRAK3 degradation.

## 2 Related work

Recent developments in deep learning have led to the emergence of DGMs for *de novo* molecular design [Chen et al., 2018, Jiménez-Luna et al., 2021, Meyers et al., 2021]. Some of the most successful are variations upon RL-based DGMs for the design of drug-like small molecules [Segler et al., 2017, Blaschke et al., 2020, Gao et al., 2022]. Policy-gradient RL has been shown to be successful in goal-directed drug design tasks which warrant the prioritization of certain properties, and has been successfully applied to fine-tune DGMs towards the design of molecules with a specific size, octanol-water partition coefficient (logP), or predicted pharmacological activity in a multi-objective fashion [Blaschke et al., 2020, Atance et al., 2021].

Previous work on DGMs for PROTAC design has focused on the conditional design of the linker starting from a desired PROTAC substructure. For example, a platform called LinkINVENT used RL to generate favorable connecting components between a pre-specified warhead and E3 ligand [Guo et al., 2022]. A graph-based DGM has also been trained to propose 3D linker structures conditioned on partial PROTAC structures [Imrie et al., 2020]. More recently, a method called PROTAC-RL was

developed which combines a transformer architecture and memory-assisted RL to linker design given an E3 ligand and warhead [Zheng et al., 2021].

# 3 Methods

We first discuss the design of a surrogate model for degradation performance, to serve as a PROTAC scoring function. We then describe the graph-based molecular DGM, followed by the integration of the surrogate model into an RL framework and its application towards IRAK3 degrader design.

### 3.0.1 Data pre-processing and feature preparation

Data was retrieved from the open-source PROTAC database (`protac-db`), which has compiled various experimental measurements from the literature.[Weng et al., 2020] A single data point includes the PROTAC's SMILES representation; the cell type, E3 ligase, and POI targeted in the experiment; and the $DC_{50}$ value. $DC_{50}$ value gives the concentration of PROTAC needed to degrade 50% of a POI in a given cell type with a specific E3 ligase. This is a measure of protein degradation activity, where a lower $DC_{50}$ value indicates a more potent PROTAC. The full dataset contains 3,994 datapoints, with 3,270 unique PROTACs represented. Datapoints containing duplicate PROTAC structures are from experiments conducted with the same molecule under different conditions.

To prepare the data, all rows with no explicit $DC_{50}$ value were dropped, resulting in 638 data points. The E3 ligase was represented using a one-hot representation of seven main classes (CRBN, VHL, IAP, MDM2, DCAF, AhR, or RNF), where the most common E3 ligase was cereblon (CRBN). Cell type was one-hot encoded into 148 classes. A comprehensive set of 88 unique sequences was used to define a vocabulary of bi-grams and tri-grams where each token is an amino acid. The size of the vocabulary was 7,841 words, and each word was used as a feature. PROTACs were represented using 1024-bit Morgan molecular fingerprints, where the fingerprint length was selected from hyperparameter optimization. Embeddings were concatenated into a final embedding with 9,077 features per data point. The continuous response variable was transformed into a categorical variable via the following cut-offs to achieve a balanced class split (Appendix Figure 7): $DC_{50} \geq 100$ nM $\rightarrow$ no to low activity (0); $DC_{50}$ 100 nM $\rightarrow$ high activity (1).

## 3.1 Surrogate model for protein degradation activity

To evaluate the quality of PROTAC structures, we developed a surrogate model to predict $DC_{50}$. The model takes as input the aforementioned embedding (subsection 3.0.1). The output is a binary label representing the activity level: 0 (low activity/high $DC_{50}$) or 1 (high activity/low $DC_{50}$). Data was divided into train/test splits using a semi-random 70/30 split, accounting for data-leakage by avoiding having the same PROTAC from in both splits. Data points where a PROTAC was in the training set were moved out of the test set, leading to 689 training points and 144 points in the hold-out test set. We trained a boosted tree-based model using Light Gradient Boosted Machine (LightGBM version 3.2.1) [LightGBM]. Hyperparameters were selected with Optuna version 2.10.0, using an F1-score objective function and five-fold cross validation [Optuna]. Varied parameters included bagging fraction, bagging frequency, learning rate, number of leaves, and feature fraction.

## 3.2 Graph-based generative model

We tackle the challenge of optimizing all three components simultaneously. We do this using GraphINVENT Mercado et al. [2021b], Atance et al. [2021], a graph-based autoregressive DGM which uses RL for molecular optimization, as it was previously demonstrated to be successful in the generation of large natural products of similar size to PROTACs Mercado et al. [2021a]. To build the DGM, we followed a three-step workflow: *preprocessing*, *training*, and *generation*. Preprocessing involves creating step-by-step graph reconstruction processes for each molecule in the training set. This step-wise information is then used to train the DGM. As `protac-db` contains <30 molecules with phosphorous and iodine, these were removed from the dataset for computational efficiency (less

padding). The final training set consisted of 4,120 molecules with atom types {C, N, O, F, S, Cl, and Br}, formal charges {-1, 0, and 1}, and a maximum heavy atom count of 139. This set was used to pre-train the DGM for 200 epochs with a batch size of 50.

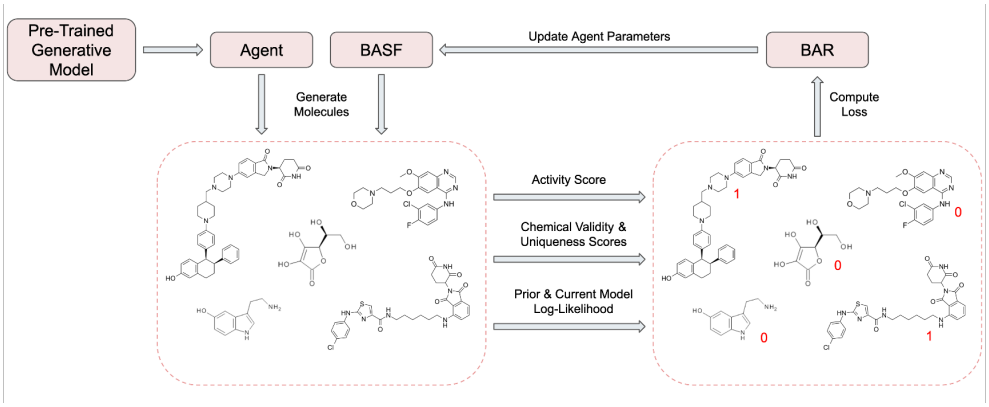


Figure 2: Reinforcement learning loop used to fine-tune the pre-trained deep generative model. BASF stands for "Best Agent So Far" and BAR represents the "Best Agent Reminder" loss.

### 3.3 RL framework for fine-tuning

We adopt the the memory-aware RL loss used in GraphINVENT [Atance et al., 2021] to steer our generative model towards PROTACs more likely to be active using a surrogate model for degradation activity. The framework, illustrated in Figure 2, begins with the pre-trained DGM, which is used to initialize the Agent as well as the Best Agent So Far (BASF). Both agents generate a set of molecules, saving the actions. Sampled molecules from each agent are scored by their predicted degradation activity, modified by a term that prevents rewarding duplicate, chemically invalid, heavy atom count <30[1], or improperly terminated molecules. The BASF, prior, and current agent log-likelihoods are used to compute the loss and update the model through gradient descent as described in [Atance et al., 2021]. This framework was used to fine-tune the model from the best pre-training epoch, $e^*$, for 200 additional steps. The pre-trained model was fine-tuned 10 times to collect statistics. The agent and BASF each use a generation batch size of six, where sampled molecules are evaluated at each RL step using the above scoring mechanism. For training the agent, the Adam optimizer was used (initial learning rate of $10^{-6}$) with the OneCycle learning rate scheduler, no weight decay, and $\alpha = 0.5$.

10K molecules were sampled from the converged agent at 200 fine-tuning steps. The molecules were evaluated for 1) degradation activity using the surrogate model (section 3.1), and 2) chemical diversity using Murcko scaffolds. These molecules were compared against the 2.5K molecules sampled from the model at $e^*$ before RL fine-tuning.

### 3.4 Case study: optimizing for IRAK3 degradation

We randomly selected interleukin-1 receptor-associated kinase 3 (IRAK3) from `protac-db` for fine-tuning the DGM. IRAK3 has been implicated in oncological signaling, and its inhibition induces T-cell proliferation for reduced tumor burden. IRAK3 contains an "undruggable" ATP binding site which has been the target of PROTAC development efforts [Degorce et al., 2020]. This is a challenging case study where the model has to learn to generate new PROTACs for IRAK3 degradation having seen few examples of IRAK3 degraders previously.

[1]Fewest heavy atoms per molecule in `protac-db` = 30.

# 4 Results

## 4.1 Activity scoring model metrics

We show here the results of the tree-based activity scoring model on the held-out test set. The model returns a score between 0 and 1 for each input molecule where higher scores suggest greater PROTAC activity. As shown in Figure 3, the final model's test AUC is 0.87, and for molecules with scores $\leq 0.3$ or $\geq 0.7$, the predicted activity is correct $87.6\%$ of the time. For molecules that score between $0.3$ and $0.7$, the probability of incorrect classification is $59.0\%$. Therefore, when using this model for DGM fine-tuning, molecules that score within this range are not rewarded. However, $90\%$ of molecules in the test set scored outside of this range, indicating the model is relatively confident in those predictions.

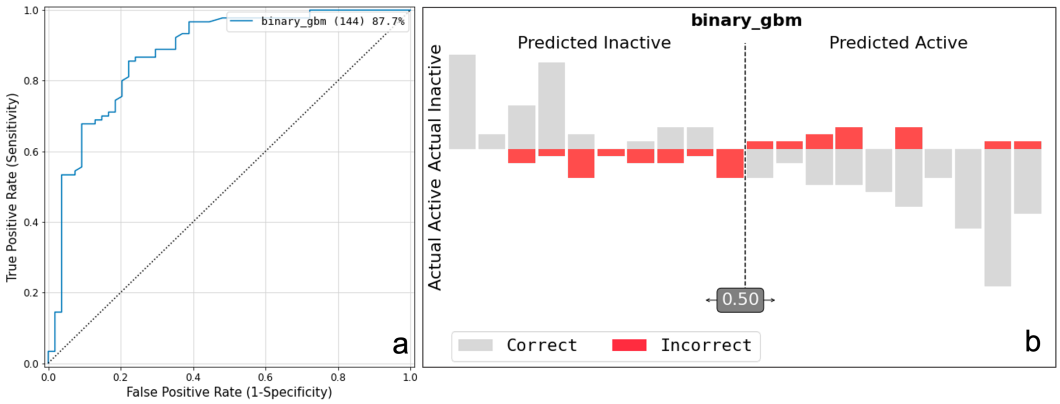


Figure 3: (a) AUC for surrogate model protein degradation activity on test set. (b) Graphical confusion matrix with respect to predicted activity score of test set molecules.

## 4.2 Fine-tuning DGM to generate active PROTACs

The DGM was fine-tuned using RL from the pre-trained model state at the epoch, $e^*$, that maximized the fraction of valid molecules in the range of epochs 150–200, as models in this range provide a good compromise between the validity of sampled structures and validation loss (Appendix Figure 8). Starting from the pre-trained model at $e^* = 192$, we observe that the score of the sampled molecules, interpretable as a binary activity measure, steadily increases for the first 100 RL steps before converging (Figure 4), suggesting that the model is learning to generate molecules that score highly for IRAK3 degradation.

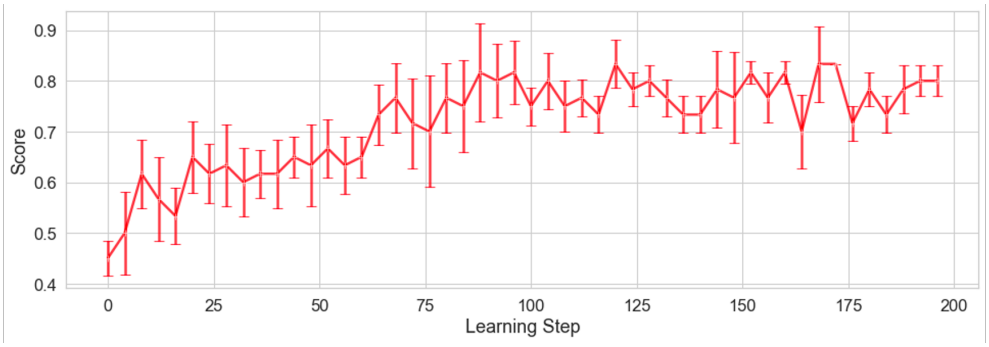


Figure 4: Activity score progression across learning steps during RL training. Error bars are the standard deviation from the results of 10 distinct runs.

To compare the quality of molecules being generated at these two different states, 2.5K were sampled from the pre-trained model at $e^*$, and 10K molecules were sampled from the fine-tuned model after 200 RL steps. Visualizations of randomly selected molecules from the two sampled sets are shown in Figure 5. What we observe is that at before applying RL, the DGM has already learned to generate barbell-like molecules. Many of the sampled molecules have well-developed E3 ligands and warheads, thought only about half ($50.8\%$) are predicted to be active degraders. However, after RL fine-tuning, we observe that $83.8\%$ of sampled molecules demonstrate predicted IRAK3 degradation activity, and nearly 100% chemical validity (Appendix Figure 8).

### 4.3 Evaluation of PROTACs generated by the best model

We evaluated the scaffold diversity of the 10K molecules generated from the DGM after RL. The average number of heavy atoms in this sampled set was 56.46, compared to 55.09 for `protac-db`. Additionally, both before and after fine-tuning, 0% of the sampled molecules were regenerated from the training set; in other words, they were all novel structures, likely due to the large number of nodes in the PROTAC graphs and the large action space during generation.

Diversity of generated molecules was evaluated using molecular scaffolds, a concept applied in medicinal chemistry to represent core substructures in bioactive compounds. Specifically, we identified the Murcko scaffolds present in the final 10K set of generated molecules and the $\sim$5K molecules in `protac-db` and computed the intersection of these two sets. In the 10K final generated molecules, we identified 537 unique Murcko scaffolds [RDKit], whereas in `protac-db` we identified 2,907 unique Murcko scaffolds. This is not surprising as during fine-tuning we are narrowing in on a smaller chemical space.

In Figure 6, we highlight the most common scaffolds shared by the top 100 generated and `protac-db` molecules. We see that the model has learned to reuse several substructures repeatedly. We see similar trends when analyzing the top 5 most common Murcko scaffolds to appear in the top 100 predicted IRAK3 degraders (out of the 10K generated set; Appendix Figure 9). Interestingly, the most common scaffold found in the top 100 predicted degraders is phthalimidinoglutarimide (PubChem CID: 91585), a known IRAK degrader and CRBN binder [Békés et al., 2022]. The second scaffold corresponds to a known degrader for the tau protein (PubChem CID: 137408522). Noticably, all top 5 most common scaffolds contain the phthalimidinoglutarimide scaffold (a substructure of thalidomide and lenalidomide), indicating that our model is satisfactorily learning the structure of CRBN ligands. This is not surprising as there is less variation amongst E3 ligands than warheads in `protac-db`.

## 5 Discussion

### 5.1 A look at the potential new degraders

Above we show that a graph-based DGM can learn the structure of potentially new degraders via reinforcement learning such that 82% of the final sampled molecules are predicted to be potent degraders with $DC_{50}$ < 100 nM when optimized against an example task from `protac-db`. Many of the sampled molecules contain scaffolds present in known degraders. Although the DGM receives no information during training on which component is the warhead, the linker, or the E3 ligand, it learns to generate promising new PROTAC-like structures based on the final reward for the entire output molecule. It also does this without any initial seed.

While previous work has primarily applied generative models to design small molecules containing $\leq 30$ heavy atoms, here we show that our DGM can be used to generate molecules containing up to 139 heavy atoms with nearly 100% chemical validity (Appendix Figure 8). This indicates that the DGM has learned chemical rules and can apply them to build large, complex molecules.

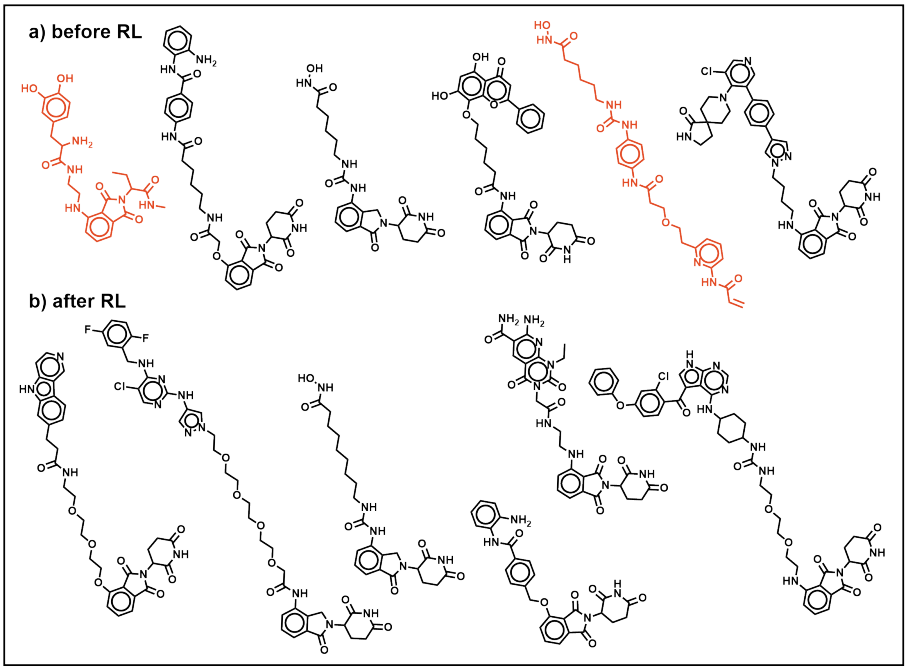


Figure 5: Generated molecules randomly sampled from the DGM (a) before and (b) after reinforcement learning (RL). The black molecules are predicted to be active for IRAK3 degradation by the surrogate model, whereas the red molecules are predicted to be inactive.

### 5.2 Limitations

The primary avenue for improvement in this work is in increasing the robustness of the $DC_{50}$ prediction model. In this work we were limited by the sparse nature of the public PROTAC data. Several data points in `protac-db`, which are mined from the literature, lacked $DC_{50}$ measurements, making those entries unusable. In addition, there is considerable class imbalance between E3 ligases, cell types, and POIs. Some ligases had hundreds of measured data points, while others just a few. Addressing these data limitations could enable the application of more complex architectures for the degradation prediction model, and also improve the DGM which builds on the accuracy of the surrogate model for PROTAC design. Another limitation is that synthesizability considerations have not been factored in to the DGM in this work, which will be relevant when it comes to experimental validation and are thus important to address, along with the aforementioned issues, in future work.

### 5.3 Future work

There remains much to improve for the creation of an automated PROTAC design pipeline; nonetheless, one main area of improvement is the accuracy of the $DC_{50}$ surrogate model. Better representations couldbe used for E3 ligase and cell type information. For instance, E3 ligase sequence information could be included, and cell type embeddings could be constructed to encode biological similarity between types. Additionally, besides the $DC_{50}$ values listed in `protac-db` for the response variable, docking studies could be used to estimate binding affinities between the PROTAC, E3 ligase, and POI. Other molecular fingerprints (e.g., atom-pair fingerprints) could also be experimented with.

While the molecules generated are predicted to be active by the surrogate model, they cannot be deemed true actives from computational predictions alone. Experimental validation *in vitro* is essential for moving forward in the drug development process of any compound designed *in silico*.

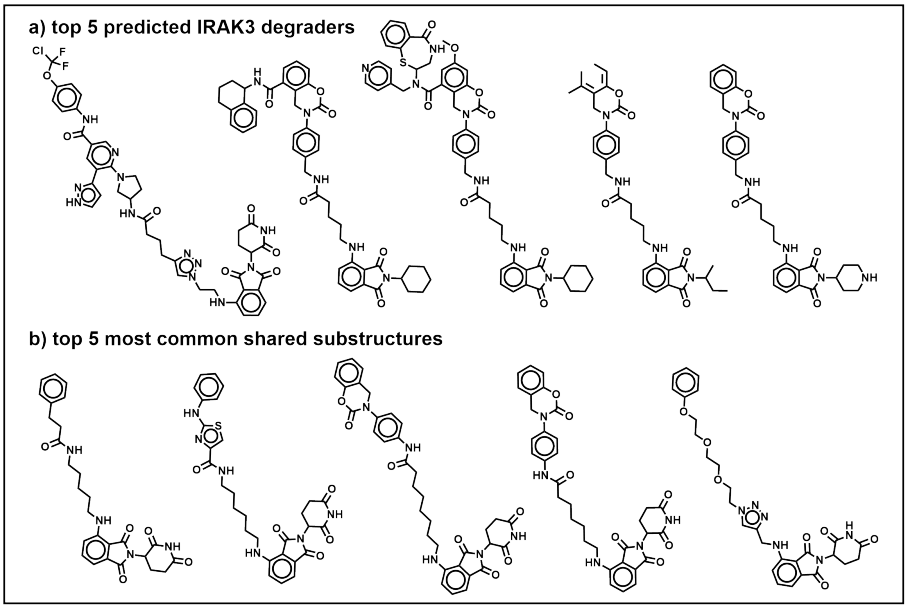


Figure 6: (a) Top 5 predicted IRAK3 degraders sampled from the DGM after reinforcement learning. (b) Top 5 most common Murcko scaffolds shared by the top 100 generated molecules and the known actives in `protac-db`.

# 6 Conclusions

Here, we have demonstrated how a graph-based DGM can be directed towards the generation of predicted active PROTACs via policy-gradient RL using a memory-aware loss function. It can be used to design PROTAC-like structures with up to 140 heavy atoms. The scoring model used in the RL framework is a boosted tree-based classification model for protein degradation activity; after hyperparameter tuning, the model displays a test AUC of $0.87$, though the generalizability of the model is limited due to the sparse nature of existing database. RL enhances the percentage of predicted active PROTACs generated by the DGM from $53\%$ to $82\%$ in 200 learning steps. Analysis of molecules sampled from the final fine-tuned model shows that while the compounds are 100% novel, they contain substructures present in known protein degraders. Generally, we have shown that our graph-based model can be used to optimize large therapeutic molecules such as PROTACs. With the availability of better public data, and the development of better physics-based models for ternary structure modeling, machine learning tools can be used to make the PROTAC design process less formidable. We hope this work inspires future research on *de novo* design tools for emerging therapeutic modalities.

# 7 Code availability

Code for the surrogate model, as well as pre-trained models, fine-tuned models, and generated structures, can be found at `https://zenodo.org/record/7134049#.YzkN_S-B3RY`.

# References

S. R. Atance, J. V. Diez, O. Engkvist, S. Olsson, and R. Mercado. De novo drug design using reinforcement learning with graph-based deep generative models. *ChemRxiv*, 2021. doi: 10.26434/chemrxiv-2021-9w3tc.

M. Békés, D. R. Langley, and C. M. Crews. Protac targeted protein degraders: the past is prologue. *Nature Reviews Drug Discovery*, 21(3):181–200, 2022.

T. Blaschke, J. Arús-Pous, H. Chen, C. Margreitter, C. Tyrchan, O. Engkvist, K. Papadopoulos, and A. Patronov. Reinvent 2.0: an ai tool for de novo drug design. *Journal of chemical information and modeling*, 60(12):5918–5922, 2020.

CDER. Compilation of cder new molecular entity (nme) drug and new biologic approvals, Mar 2022. URL `https://www.fda.gov/drugs/drug-approvals-and-databases/compilation-cder-new-molecular-entity-nme-drug-and-new-biologic-approvals`. Accessed Sep 22, 2022.

H. Chen, O. Engkvist, Y. Wang, M. Olivecrona, and T. Blaschke. The rise of deep learning in drug discovery. *Drug discovery today*, 23(6):1241–1250, 2018.

S. L. Degorce, O. Tavana, E. Banks, C. Crafter, L. Gingipalli, D. Kouvchinov, Y. Mao, F. Pachl, A. Solanki, V. Valge-Archer, B. Yang, and S. D. Edmondson. Discovery of Proteolysis-Targeting Chimera Molecules that Selectively Degrade the IRAK3 Pseudokinase. *Journal of Medicinal Chemistry*, 63(18), 2020. doi: https://doi.org/10.1021/acs.jmedchem.0c01125.

D. C. Elton, Z. Boukouvalas, M. D. Fugea, and P. W. Chunga. Deep learning for molecular design—a review of the state of the art. *Molecular Systems Design Engineering*, 4:828–849, 2019.

A. Frenzel, F. Grespi, W. Chmelewskij, and A. Villunger. Bcl2 family proteins in carcinogenesis and the treatment of cancer. *Molecular Systems Design Engineering*, 4:584–596, 2009. doi: 10.1007/s10495-008-0300-z.

W. Gao, T. Fu, J. Sun, and C. W. Coley. Sample efficiency matters: A benchmark for practical molecular optimization. *arXiv preprint arXiv:2206.12411*, 2022.

J. Guo, F. Knuth, C. Margreitter, J. P. Janet, K. Papadopoulos, and O. E. A. Patronov. Link-INVENT: Generative Linker Design with Reinforcement Learning. 2022. doi: 10.26434/chemrxiv-2022-qkx9f.

F. Imrie, A. R. Bradley, M. van der Schaar, and C. M. Deane. Deep Generative Models for 3D Linker Design. *Journal of Chemical Informatics*, 2020. doi: 10.1021/acs.jcim.9b01120.

J. Jiménez-Luna, F. Grisoni, N. Weskamp, and G. Schneider. Artificial intelligence in drug discovery: Recent advances and future perspectives. *Expert opinion on drug discovery*, 16(9):949–959, 2021.

LightGBM. LightGBM. `https://lightgbm.readthedocs.io/en/v3.3.2/`. Accessed 08/01/22.

D. Lv, P. Pal, X. Liu, Y. Jia, D. Thummuri, P. Zhang, W. Hu, J. Pei, Q. Zhang, S. Zhou, S. Khan, X. Zhang, N. Hua, Q. Yang, S. Arango, W. Zhang, D. Nayak, S. K. Olsen, S. T. Weintraub, R. Hromas, M. Konopleva, Y. Yuan, G. Zheng, and D. Zhou. Development of a bcl-xl and bcl-2 dual degrader with improved anti-leukemic activity. *Nature Communications*, 12, 2021.

R. Mercado, E. J. Bjerrum, and O. Engkvist. Exploring graph traversal algorithms in graph-based molecular generation. *Journal of Chemical Information and Modeling*, 62(9):2093–2100, 2021a.

R. Mercado, T. Rastemo, E. Lindelöf, G. Klambauer, O. Engkvist, H. Chen, and E. J. Bjerrum. Graph networks for molecular design. *Machine Learning Science and Technology*, 2(2), 2021b. doi: https://doi.org/10.1088/2632-2153/abcf91.

J. Meyers, B. Fabian, and N. Brown. De novo molecular design and generative models. *Drug Discovery Today*, 26(11):2707–2715, 2021.

Optuna. Optuna. `https://optuna.readthedocs.io/en/stable/`. Accessed 08/01/22.

RDKit. RDKit. Accessed 08/01/22.

M. Segler, T. Kogej, C. Tyrchan, and M. Waller. Generating Focused Molecule Libraries for Drug Discovery with Recurrent Neural Networks. *ACS Cent Sci*, 4(1), 2017. doi: https://doi.org/10.1021/acscentsci.7b00512.

M. Toure and C. M. Crews. Small-molecule protacs: new approaches to protein degradation. *Angewandte Chemie International Edition*, 55(6):1966–1973, 2016.

G. Weng, C. Shen, D. Cao, J. Gao, X. Dong, Q. He, B. Yang, D. Li, J. Wu, and T. Hou. PROTAC-DB. `http://cadd.zju.edu.cn/protacdb/downloads`. Accessed 08/01/22.

G. Weng, C. Shen, D. Cao, J. Gao, X. Dong, Q. He, B. Yang, D. Li, J. Wu, and T. Hou. Protac-db: an online database of protacs. *Nucleic Acids Research*, 49(D1):D1381–D1387, 2020.

S. Zheng, Y. Tan, Z. Wang, C. Li, Z. Zhang, X. Sang, H. Chen, and Y. Yang. Accelerated rational PROTAC design via deep learning and molecular simulations. *Nature Machine Intelligence*, 4(1), 2021. doi: https://doi.org/10.1038/s42256-022-00527-y.

# 8 Appendix

## 8.1 Feature importance ablation study

Feature importance was measured via an ablation study using input feature shuffling on the held-out test set. For a given embedding, the respective column values were shuffled in the input matrix to randomize the values for those variables. The scoring model was then used to predict the $DC_{50}$ class from the modified input. The results are shown in Table 1, where "Importance" is calculated by subtracting the AUC and F1 score sum from the original (unshuffled) model's AUC and F1 score sum. Input shuffling occurs with slight variation on each pass, so the experiment was conducted three times and used to compute the standard error of the mean (SEM) for each importance value.

| Shuffled Embedding | AUC | F1 Score | Importance | SEM |
|---|---|---|---|---|
| None (Original) | 0.860 | 0.893 | NA | NA |
| Full PROTAC | 0.611 | 0.832 | 0.310 | 0.012 |
| Receptor (POI) | 0.765 | 0.838 | 0.150 | 0.023 |
| E3 Ligase | 0.855 | 0.893 | 0.005 | 0.000 |
| Cell Type | 0.860 | 0.893 | 0.000 | 0.000 |

Table 1: Feature importance in the surrogate model.

From Table 1, we observe that the PROTAC molecular fingerprints (1024-bits) are the most important. The receptor n-grams (7,841 bi-gram & tri-gram features) follow, with the one-hot encoded E3 ligases (7 features) and cell types (148 features) being the least important. This shows that the model learns from both the PROTAC structure and receptor to reliably predict PROTAC degradation activity for the POI. In comparison, E3 ligase has relatively low feature importance, whereas the cell type is seemingly not contributing at all to the model's learning under the current featurization scheme.

## 8.2 Response variable preparation

Figure 7 shows the histogram which was used to determine the threshold for encoding the response variable, $DC_{50}$, into balanced binary classes. Based on this distribution, a threshold of 100 nM was used to split $DC_{50}$ values into two classes. All $DC_{50}$ values <100 were put into one class (1: "high activity"), and the remaining values were put into another class (0: "no to low activity").

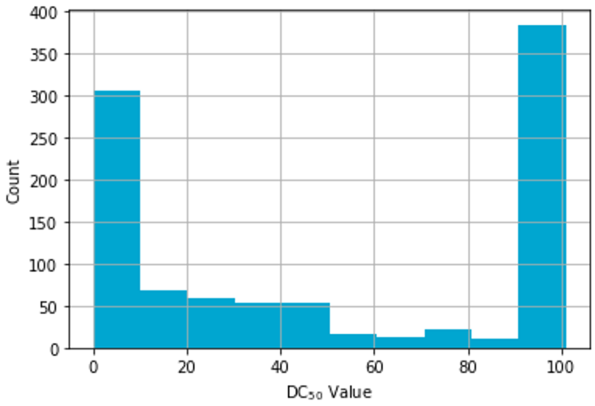


Figure 7: $DC_{50}$ histogram for the 638 `protac-db` datapoints used in this work.

| F1 Score | Bagging Fraction | Bagging Freq | Learning Rate | Number of Leaves | Feature Frac |
|---|---|---|---|---|---|
| 0.8316 | 0.8876 | 3 | 0.2299 | 19 | 0.5412 |
| 0.8288 | 0.8153 | 3 | 0.2936 | 21 | 0.4045 |
| 0.8287 | 0.8784 | 3 | 0.2957 | 20 | 0.3990 |
| 0.7126 | 0.6383 | 5 | 0.0346 | 17 | 0.4954 |
| 0.6981 | 0.4260 | 4 | 0.0123 | 24 | 0.4600 |
| 0.6953 | 0.5313 | 6 | 0.1922 | 11 | 0.5114 |

Table 2: Surrogate model parameters varied during hyperparameter optimization.

### 8.3 Hyperparameter optimization

Table 2 shows the hyperparameters tuned during surrogate model training. The top three rows show the best three hyperparamater combinations, ranked using an F1 objective function. The bottom three rows show the worst three combinations.

### 8.4 Additional training results

In Figure 8, we show the training/validation loss and fraction of valid molecules sampled from the DGM as a function of training epochs. Initially, as the number of epochs increases, the average validation loss increases. Subsequently, the validation loss increases sightly and then plateaus, indicating some amount of overfitting. The fraction of valid molecules reaches 100% by the final epoch.

### 8.5 Top Murcko scaffolds

In Figure 9 we illustrate the most common Murcko scaffolds present in the sampled molecules predicted to be most active after RL. The two left-most molecules in this figure are known protein degraders.

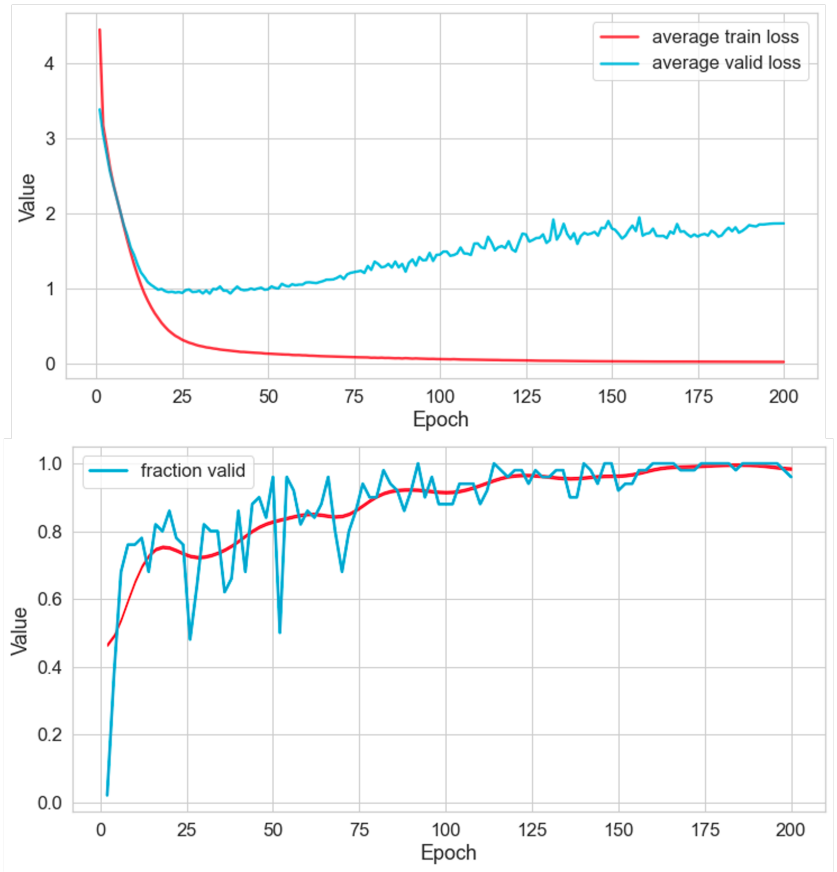


Figure 8: Training loss, validation loss, and fraction valid as a function of pre-training epochs.

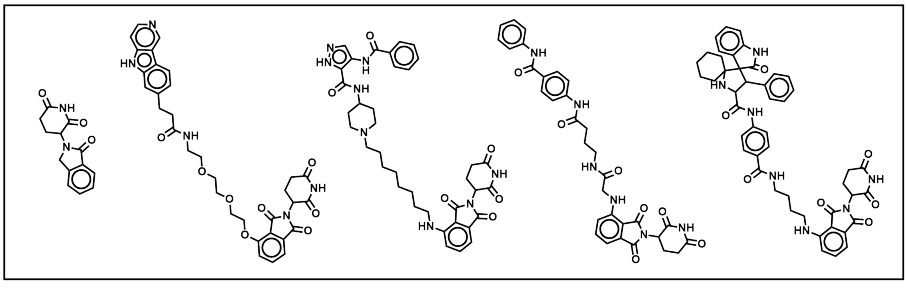

Figure 9: Most common Murcko scaffolds appearing in the top 100 predicted degraders for IRAK3, in order of most common (left) to less common (right).