Testing the Limits of SMILES-Based *De Novo* Molecular Generation with Curriculum and Deep Reinforcement Learning

Maranga Mokaya[1], Fergus Imrie[2], Willem P. van Hoorn[3], Aleksandra Kalisz[3], Anthony R. Bradley[3], Charlotte M. Deane[1,3] *

[1] Department of Statistics, University of Oxford
[2] University of California, Los Angeles
[3] Exscientia, Oxford, UK

*Corresponding author: deane@stats.ox.ac.uk

1 Abstract

Deep reinforcement learning methods have been shown to be potentially powerful tools for *de novo* design. Recurrent neural network (RNN)-based techniques are the most widely used methods in this space. In this work, we examine the behaviour of RNN-based methods when there are few (or no) examples of molecules with the desired properties in the training data. We find that targeted molecular generation is often possible, but the diversity of generated molecules is often reduced, and it is not possible to control the composition of generated molecular sets. To help overcome these issues, we propose a new curriculum learning-inspired, recurrent Iterative Optimisation Procedure that enables the optimisation of generated molecules for seen and unseen molecular profiles and allows the user to control whether a molecular profile is explored or exploited. Using our method, we generate specific and diverse sets of molecules with up to 18 times more scaffolds than standard methods for the same sample size. However, our results also point to significant limitations of one-dimensional molecular representations, as used in this space. We find that the success or failure of a given molecular optimisation problem depends on the choice of SMILES.

2 Introduction

Developing a novel drug is a complex and difficult problem plagued with failure at many stages.[1–4] The efficiency of drug development could be improved by producing better early hits and generating novel molecules with specific properties which would improve cost, speed, and effectiveness.[5]

Ideally, given a target and a required molecular profile, we would search for suitable molecules in all drug-like chemical space. However, given that there are an estimated 10^{60} synthetically accessible drug-like molecules[6], of which approximately 10^8 have been synthesized[7], experimental methods are not sufficient for comprehensive sampling of chemical space.

Computational methods offer the promise of searching larger areas of chemical space and virtual screening is commonly used to search curated chemical libraries for potential hits.[8–11] However, the chemical space available for assessment is only a tiny proportion of the possible space.[12–14]

Instead of searching existing molecular datasets, computational *de novo* design models aim to create new sets of novel molecules[11,15–18] Recently, deep learning molecular generation tools have become more prevalent, which are also often paired with optimisation pipelines to produce focused sets of molecules with improved performance.[19] Autoencoders[20] are frequently used for generation and optimisation.[21] Here the discrete representation of a molecule is converted to a continuous representation (encoded) from which its properties can be predicted and optimised. The resulting continuous representation is then converted back to a discrete molecular representation (decoded).[22,23] Bayesian optimisation has been paired with variational autoencoders (VAE) to explore chemical space for specific molecular profiles. However, the transition from latent space to discrete molecular representations is oftentimes non-trivial.[21] Generative Adversarial networks (GAN) have been applied to de novo drug design. Together with reinforcement learning (RL), these models have been shown to generate diverse libraries of realistic molecules with specific properties.[24–26] Recent developments in state-of-the-art natural language processing (NLP) tools have been implemented in *de novo* design with the introduction of transformer models[27] for SMILES generation.[28–31] These tools have been reported with stable, and sometimes better, performance than older architectures.

Like transformers, recurrent neural networks (RNNs) based molecular generation models were inspired by NLP tools and have proved popular due to their ability to model long term dependences in strings.[32] Using the

same training regime implemented in NLP, RNNs have successfully been applied to the generation of novel molecules while still utilising a simple network architecture.[33,34] Unbiased RNNs have been shown to generate SMILES that cover large areas of chemical space (especially relative to training data). These models have also been shown to benefit from randomised SMILES with further performance improvements.[11,35,36] An early example of this approach used RNNs to generate a molecular library through a SMILES language model[15], before fine-tuning the model on a smaller subset of molecules with desired properties.

Another popular method for molecular generation is RL. In theory, RL methods allow users to generate a set of molecules with specific properties without explicit examples of molecules that match the reward profile. Successful hit generation requires new molecules with novel combinations of properties, which often do not exist in currently available datasets; therefore, for RL methods to be truly useful, they must be able to extrapolate. It is currently unclear how well these models can do so. It is also important that RL methods generate sets of molecules that explore chemical space for a complex molecular profile and produce a diverse library and exploit chemical space to generate focused molecular libraries. Here, based on previous on-policy RL models,[37] we explore scenarios where optimisation is attempted with little or no representation in the training data and investigate the extent of extrapolation. We manipulate the prevalence of specific properties, measured as a percentage of the entire training dataset, and test the limits of optimisation of individual properties. We find that the RL models tested can extrapolate beyond the training data, but often produce molecule sets with little diversity. We show that these models are frequently unable to generate molecules that satisfy complex molecular profiles. We go on to demonstrate a curriculum-learning-inspired optimisation procedure that enables the generation of specific and diverse sets of molecules that satisfy complex and unseen molecular profiles. We also highlight the limitations of SMILES-based molecular generation tools.

3 Results & Discussion

Deep RL molecular generation models are powerful tools for optimising molecular properties. However, their usefulness is dependent on their input training data. We show how these tools can optimise for specific property values; however, only within a property specific value range. We also show how these methods can generate molecular profiles that are not present in training data and how the representation of training data affects the composition of generated sets of molecules.

By evaluating the performance of deep RL molecular generation methods with increasing proportions of training data that match a desired property profile, we show the effects of representation on generated molecules. To overcome the generated library restrictions caused by training data representation, we propose a curriculum learning inspired approach (rIOP) that allows for the optimisation of under-represented properties. rIOP also allows users to optimise generated molecules toward complex molecular profiles that are not possible with REINVENT while controlling the diversity of generated libraries. Our work highlights the strengths and limitations of using these tools on single parameter optimisation tasks. These strengths and limitations are important to understand before the discussed methods can be used for multiparameter optimisation in a drug discovery pipeline.

3.1 Control of Generated Libraries

A standard goal for *de novo* design deep RL tools is to produce novel molecules with controlled distribution of a single property or many properties. Previous studies using REINVENT and ReLeaSE have shown that it is possible to bias generated molecules towards specific properties such as hydrophobicity, melting point, or predicted activity against the DRD2 receptor.[38,39]

We tested the optimisation performance of REINVENT by shifting the reward for cLogP and HBA count. Figure 1a shows the property distribution of generated molecules for the reward ranges of cLogP between -15 and 20. It shows that it is possible to control the position of agent distributions with the reward range. However, in extreme cases (cLogP reward between -15 and -10), optimisation was unsuccessful, and we observe no change in the agent distribution; the training data distribution is reproduced.

The same behaviour was observed for HBA counts (Figure 1b). Optimisation anywhere in the range of 0-20 was possible; however, optimising the model to generate molecules with 20 HBA's or more was unsuccessful (red distribution). As with cLogP, a distribution similar to the training data was reproduced. The training data included several molecules with more than 30 HBA's. A complete breakdown of the generated molecular sets, reward ranges, and molecule examples can be found in Supplementary Information 2.

We postulate that this failure occurs because in the extreme case, fewer molecules generated by the prior model return any reward during RL. If trained for an infinite time, the model will eventually randomly generate SMILES that will return a positive reward; nevertheless, poor representation can prevent effective optimisation. This ineffective optimisation then leads to the model repeatedly producing the same SMILES seen in training; thus, the training data is reproduced.

3.2 Effect of percentage representation

We have shown that using deep RL molecular generation tools, optimisation of under-represented properties is

sometimes not possible. To investigate how widespread this issue is, we tested the ability of the models to generate molecules with properties within and outside the training data. We created several datasets in which the proportion of molecules that corresponded to the desired reward profile varied.

Extended Data Table 1 shows that using REINVENT optimisation was successful for all properties across all percentage representations. In all cases, the generated set distributions were shifted toward our desired property profile relative to the training data. These results show that, for the properties tested, the model was able to learn the chemical-structural relationship from the surrounding molecules in the distribution; it is possible to learn without representation in the training data.

Extended Data Table 1 also shows that the composition of each generated library depends on the representation of the desired property in the training data. For example, we can generate sets where most molecules have a HBA count > 8 (the reward threshold), but a higher percentage representation leads to molecules with higher HBA counts with the same reward function. The mean of the 0% representation is 9 HBAs compared to 20 for the 10% experiment. Directional changes across each experiment can be seen for all properties tested (supplementary information 3). We postulate that the trends in the generated molecules mirror the training data. For example, for a specific reward property value (e.g. HBA = 5) if, as the percentage representation of that property profile increases, the diversity of the training data increases, you will see an increase in the diversity of the generated library. Conversely, if all the training data examples were very similar, you would observe a reduction in training data diversity and generated library diversity. Therefore, for most properties, we expect that a lower percentage representation would lead to a less diverse generated library.

These results show that it is possible to generate molecular profiles that are not seen during training and that the composition of the generated molecular sets depends on the degree to which the desired molecular profile is prevalent in the training data. Therefore, depending on the use case for these models, different percentage representations for training may be suitable. However, when the aim is to generate an unseen molecular profile (0% representation), standard methods leave the user without control over the composition of the generated library.

3.3 Curriculum Learning for Generated Library Control

We have shown that it is possible to generate molecules with unseen molecular profiles in training data (Extended Data Table 1). However, the model's ability to do this is limited at the extremes of the property distribution (Figure 1). The composition of each generated set depends on the prevalence of molecules in the training data with the desired molecular profile. Building on previous work[40], we postulate that higher training data representations would often lead to greater diversity for generated molecules, as the model would often have a more diverse set of examples to learn chemical-structural relationships.

To improve the efficacy of deep RL generation methods, we propose a new curriculum learning-inspired approach, called recurrent iterative optimization procedure (rIOP). Our method allows deep RL generation methods to maximise the diversity of generated molecules for seen and unseen molecular profiles during optimisation. It also enables the model to generate molecules that perform more complex optimisation tasks where standard methods fail.

3.3.1 Iterative Optimization Procedure to Improve Diversity

To demonstrate how rIOP can increase the diversity of simple optimisation tasks, we undertook a TPSA optimisation task with SrIOP and REINVENT's standard implementation. Standard methods do generate molecules in this range; however, we expect that with SrIOP we will see an improvement in the diversity of generated molecules. TPSA shift is a simple optimisation task; therefore, we only sampled the agent of the previous step when training the current prior (see Methods 5.3).

Figure 2 shows each step of the SrIOP procedure and the change in property distribution at each stage to match the reward function. To measure the diversity of each generated library, we calculated the number of unique Murcko scaffolds generated. With each SrIOP step, we see a reduction in the number of scaffolds generated. This is expected as we are moving toward the limit of the full TPSA property distribution, where there are fewer ways to achieve these property values. Extended Data Table 2 shows that in our last step (SrIOP 4) we produce 18 times more scaffolds using SrIOP (55 scaffolds) compared to REINVENT (3 scaffolds). Furthermore, SrIOP generates more molecules from the 500 sampled that match the reward profile (494 compared to 476 for SrIOP and REINVENT, respectively). Extended Data Table 2 also shows the internal diversity of generated sets and the proportion of generated molecules that were present in the training dataset. Examples of generated molecules can be found in the data repository section 6.

3.3.2 Iterative optimisation Procedure for Diversity Control

For *de novo* design tools to be effective, it should be possible to control the specificity of the generated molecules. We have shown how the representation of a desired profile during training can affect the composition, and hence the specificity, of generated sets of molecules. Our results also show how the use of our new method,

SrIOP, can improve the diversity of molecules generated during simple optimisation tasks.

To test this, we aimed to generate druglike molecules (QED > 0.8) from training with a dataset containing no high QED molecules (QED < 0.8). Figure 3 shows how with and without diversity filters it is possible to generate molecules with high QED values. Figure 3b also shows that a wider distribution of molecules is produced with a diversity filter enabled. Of the 500 molecules sampled, SrIOP generates 490 molecules that match the reward function compared to only 317 using REINVENT. Of the 490 molecules, SrIOP generated 22 scaffolds. Standard methods produced more with 130 scaffolds across those 317 molecules. However, with DF enabled, we observed a significant increase in SrIOP performance, with scaffold diversity increasing from 22 to 297 in the 301 generated molecules. We also see a change using REINVENT; the model generates 234 scaffolds across 236 molecules. These results show how, using our SrIOP, we can generate a specific library of molecules (without DF), and a diverse library with DF enabled. In contrast, REINVENT is only able to generate diverse molecules, eliminating the user's ability to control the composition of the generated sets. Each generated dataset with example molecules is available in the data repository (Data Availability 6).

SrIOP gives the user more control over the specificity of the generated library. If they wish to exploit a property, SrIOP used without DF filters will return a very narrow selection of molecules that match your reward profile. In contrast, if a diverse library is required, enabling diversity filters with SrIOP will produce one. In this example, we chose a simple optimisation task as there are many ways to increase QED for a molecule. Therefore, we expected that the standard method would perform well. SrIOP still outperforms REINVENT in both specific (no diversity filter) and diverse (diversity filter used) set generation; however, we expect the difference in performance to increase for more complex optimisation tasks.

3.4 Comparison to other curriculum learning methods

Curriculum learning has long been used as a tool to overcome complex machine learning problems in various applications.[41] However, its use in deep RL molecular generation tools is limited. There is one implementation of a similar method by the original authors of REINVENT[40], that applies a curriculum learning approach to solve complex optimisation tasks, which we call ReCL.

3.4.1 Iterative Optimisation Procedure for Complex Tasks

One common use case of deep learning RL models is to optimise for molecules similar to a target structure. In such scenarios there may be few examples of the target structure in training data. To determine how useful SrIOP is in this practical situation, we have used it to generate molecules similar to target structures (Extended Data Figure 1) with increasing difficulty.

Table 1a shows how for a simple molecule (Extended Data Figure 1a) it is possible to generate molecules identical to the target. Both SrIOP and ReCL perform well, with SrIOP generating more molecules with a tanimoto similarity between 0.9 and 1.0 (486 and 325 for SrIOP and ReCL, respectively).

For a more complex molecule (Extended Data Figure 1b), ReCL is less successful (Extended Data Figure 2b and Extended Data Figure 2c) because it cannot generate any molecules with a high similarity to the target structure (tanimoto similarity greater than 0.7). For SrIOP almost all (497 of 500 sampled) generated molecules have a high similarity to the target structure.

Another benefit of our method is the ability to control the diversity of the generated library.

Table 1 shows how, without a diversity filter, all 497 molecules sampled have the same scaffold. However, with diversity filters enabled, SrIOP generates 142 scaffolds across 447 molecules. In this example, we highlight the ability of SrIOP to fulfil more complex reward functions where similar CL methods fail. We show how it can also be used to control the diversity of the generated library through the inclusion of diversity filters.

3.5 Limitations of SMILES-Based Molecular Generators

We have investigated the performance of, and proposed novel, SMILES-based, deep learning, molecular generation tools. These tools learn to generate novel one-dimensional SMILES representations of three-dimensional molecular structures (see Methods 5). SMILES-based tools are popular because they only require simple architectures and can be trained quickly. However, SMILES do present challenges; namely, they do not detail the three-dimensional structure of a molecule beyond atomic connections, and there are several ways to represent the same molecule. Canonical SMILES provide a standard method to generate SMILES; however, it has been shown that SMILES-based models trained on random SMILES show improved model coverage and a reduction in overfitting.[36]

The lack of structural information and inherent redundancy in SMILES can cause SMILES-based models to struggle to fully understand the chemical and structural relationships between molecules. This is because the similarity between two SMILES is not well correlated with the similarity between the chemical structures they represent. This limitation of SMILES-

based methods and its effects can be seen in our study. For example, when we tried to generate molecules with increasingly complex structures, the performance of the model was heavily dependent on the strings used to represent each substructure. We found that for the best performance, the difference between each string representing a new target structure should be minimised.

We found that the choice of SMILES directly affects the performance of the models. We aimed to generate molecules with a specific substructure, however, we used several alternate SMILES to represent each identical substructure during RL. Approximately a quarter of all substructure generation attempts failed (no molecules were generated with the final target substructure) across all molecules (Extended Data Table 3). But, for each failed attempt, an alternative series of SMILES representing the same molecules was successful. We observed these successful and unsuccessful attempts for near identical molecules. This highlights how the likelihood of success is more dependent on the choice of SMILES over molecule complexity. Figure 4 shows the total number of SMILES sampled from the final agent that included the desired substructure and the total number of distinct scaffolds present in successful attempts for two structurally similar molecules (A and B). For molecule A, both ReCL and SrIOP fail to generate the final target substructure at least once, and both methods fail in the final step (supplementary information 7). For molecule B, we were able to generate substructures regardless of the choice of SMILES even though the intermediate and final substructures were almost identical compared to those of molecule A. This further highlights the issues caused by SMILES as you would expect similar performance across both models given the targets' structural similarity. Instead, the SMILES used has the largest effect on model performance.

The choice of SMILES also has a large effect on the diversity of the generated molecules. For molecule B using SrIOP series 3 (Figure 4c) we generated more than 250 distinct scaffolds across the 500 molecules sampled, while all other SMILES series generated between 50 and 150 scaffolds with ReCL and SrIOP. Similar fluctuations in performance were observed for both ReCL and SrIOP across all molecules tested (supplementary information Figure 13).

Molecular representations such as SELFIES[42] and DeepSMILES[43] attempt to overcome some of the issues of SMILES in machine learning. However, higher dimensional representations that include structural information are likely to be a more powerful way to represent molecules.

4 Conclusions

We have investigated how well deep RL molecular design methods can search beyond the chemical space represented in training data and the effects of the composition of the training data on generated molecular sets. The results show that it is possible to control the distribution of molecules generated by altering the reward function. However, we demonstrate how standard methods (REINVENT) can fail towards the edge of the training data distribution. We found that it is possible to generate molecules with properties that are not present in the training data; nevertheless, we showed that the representation of the desired molecular profile affects the distribution of the generated molecular library. We highlight the lack of control standard methods provide in terms of composition, particularly diversity, of generated molecules and the limitations of SMILES-based molecular generation methods. To overcome some of these issues, we propose a new curriculum learning approach, recurrent iterative optimisation procedure (rIOP), to help boost the diversity of generated molecules when few or no examples of the desired molecular profile are present in the training data. Using this method, we generate structures similar to a series of unseen target structures and outperform other curriculum learning approaches (ReCL). We describe SrIOP and DrIOP, which enable a user to control the diversity of generated molecules for simple and complex optimisation tasks. Using several SMILES representations of the same molecule when generating target structures, we show how the choice of SMILES directly affects the success and performance of SMILES-based tools. Therefore, our method, like any method based on SMILES or other one-dimensional representations, will be hampered by the lack of direct structural information.

5 Methods

We assessed the performance of a popular on-policy SMILES generation model, REINVENT[38], to determine the limits of deep RL tools in molecular design. Like earlier RL molecular generation tools[39], REINVENT involves a two-step process. The first is to train a prior RNN to generate SMILES through supervised learning. This model is trained to correctly predict the next character of a SMILES string given a starting token or an incomplete string. The second is to fine-tune the prior model, producing an agent model able to generate a focused library through a reward-feedback loop. During this second step, the model learns a policy that maximises the likelihood of generating a molecule with a favourable reward (Figure 5). Full details of the models can be found in the work of Olivecrona *et al.*[38]

For all experiments described in the following, the model was trained on subsets of 1.5 million drug-like molecules from ChEMBL[44]. After the model was fully trained, we sampled 500 molecules unless otherwise stated. We chose to generate 500 molecules as this provided a large enough sample from which we could draw clear conclusions about the distribution of generated molecules.

5.1 Property Characterisation

Hydrogen bond acceptors (HBA), donors (HBD), molecular weight (MW), topological polar surface (TPSA) and cLogP were calculated using the chemical descriptor module from RDKit[45] Synthetic accessibility (SA)[46], QED[35], and Tanimoto similarities were also calculated using RDKit.

For the analysis of generated molecular sets, only unique molecules (all valid SMILES generated once repeats are removed) were considered. To compare the diversity of training and generated datasets, Murcko scaffolds[47] were generated using RDKit. The generated library internal diversity scores were calculated using MOSES.[48]

5.2 Reinforcement Learning

To successfully optimise for a property, a suitable reward function must be provided. For simplicity, throughout our study, we used the same step reward function (examples available in the Supplementary Information 1). Any invalid SMILES did not return a reward, and all valid SMILES that met the reward criteria returned a reward of one.

5.2.1 Optimisation Success

We initially determined the success of optimisation using the proportion of molecules in the generated library that fell within the reward range. However, after several properties were tested at increasing representations, it became clear that the difference between the proportion of optimised molecules in successful and unsuccessful optimisation attempts was large enough so that a success threshold was not appropriate. Instead, optimisation attempts that showed an increase in the proportion of optimised molecules were deemed successful. This was possible because all unsuccessful attempts resulted in zero optimised molecules.

5.2.2 Controlling Generated Libraries

To determine the degree to which the property distribution of the molecules generated by the agent (agent distribution) can be controlled, we set REINVENT the task of shifting the distribution of cLogP or HBA counts across their respective ranges. These properties or the property ranges tested may not be the most important in a drug discovery context; however, these experiments allow us to assess optimisation performance. If it is possible to control the distribution of generated molecular sets, we expect to observe changes in the composition of these sets as the reward range changes. During RL, all valid SMILES that satisfy the reward function are given a reward of one. All other molecules, valid or not, receive zero reward.

5.2.3 Effects of Percentage Representation

In our study, our aim was to determine the effect that training data representation on the generated molecules. To do this, we prepared several training datasets in which the proportion of molecules that matched the desired reward profile varied. We chose reward profiles (supplementary information 1) at the upper end of the full ChEMBL training data distribution such that at least 10% of the training data matched the reward function. Once the reward range was calculated, all molecules that matched the reward profile were removed from the full training data set. Then smaller random samples equal in size to 0%, 2%, 5%, 7% and 10% of the entire training dataset were put back and used to train the model from scratch.

5.3 Recurrent iterative optimization procedure

We propose a novel, curriculum learning inspired recurrent Iterative Optimisation Procedure (rIOP). Curriculum learning is a method used to teach models how to complete difficult tasks through the gradual introduction of more complex examples during training.[49] For single-step optimisation attempts, it is common for RL methods to exploit molecular motifs found to return positive rewards, leading to generated sets with low diversity (specialisation). We expect that the greater the difference between the reward profile and training data, the more prevalent this behaviour is. By splitting the optimisation task into a series of smaller tasks, we reduced the difference between the molecules generated by the prior and the desired reward profile at each step. Thus, reducing the likelihood of early specialisation. Repeating a prior/agent training loop with a series of small changes in the reward profile, we encouraged each agent model to shift its property distribution toward the final, desired, distribution. The resulting agent was then used as the prior in the next step. Splitting the final optimisation task into a series of increasingly complex subtasks allowed the model to satisfy increasingly difficult molecular profiles that directed it toward the final goal.

We demonstrate the use of two implementations of rIOP, that can be used in on-policy RL training regimes. The first, single model rIOP (SrIOP), only samples from the previous agent when training the current model. The second, double model (DrIOP), samples from the previous two models. Unless otherwise stated, for DrIOP we sampled the current agent once every five times the previous agent was sampled.

5.3.1 Diversity Filters

To control diversity, where appropriate, we incorporated the diversity filters described by Blaschke *et al.*[50] With diversity filters (DF) enabled, the model will only give a positive reward for the first n molecules that satisfy the

reward function for a given scaffold. Once *n* molecules that match the reward profile have been generated, molecules with this scaffold are no longer rewarded. This prevents the model from entering a local optimisation minimum by producing many molecules with the same scaffold and small structural differences to satisfy the reward function.

5.3.2 RIOP for Diversity Control

To demonstrate how rIOP can be used to control the diversity of generated molecules, we conducted two experiments. Firstly, we generated molecules with a reward for TPSA between 250 and 300 using SrIOP and REINVENT's standard implementation. Secondly, we created a set of molecules in which our aim was to maximise QED. We use no molecules with QED greater than 0.8 during training, then iteratively increased the QED reward profile at each step. Diversity filters were also used to further improve the diversity of generated molecules.

5.3.3 rIOP for Complex Optimisation Tasks

To showcase rIOP's ability to complete complex optimisation tasks, we generated molecules similar to target structures with no relevant examples in the training data. We removed all molecules with tanimoto similarity greater than 0.4 to each target from the training data and then increased the tanimoto similarity reward threshold by 0.1 at each step.

5.3.4 Limitations of SMILES

To examine the limitations of SMILES-based molecular generators, we attempted to generate molecules that included a target substructure using multiple different SMILES strings to represent the intermediate substructures. We used ReCL and SrIOP to generate molecules with a series of increasingly complex substructures. For each molecule, we enumerated five alternate SMILES for each target intermediate substructure. The intermediate SMILES across each series were kept constant.

6 Data Availability

The trained generative model used in some of our work is already published by Patronov *et al*[40] and is available at: https://github.com/m-mokaya/RIOP/blob/main/models/random.prior.new.

All other raw data needed to reproduce the experiments in this work are provided at: https://github.com/m-mokaya/RIOP/blob/main/data.

7 Code Availability

The code used in this study is available at https://github.com/m-mokaya/RIOP. Example notebooks for each experiment are available at https://github.com/m-mokaya/RIOP/tree/main/notebooks.

DOI: https://doi.org/10.5281/zenodo.7406695

8 Author Contributions

M.M. developed the code. M.M., F.I., A.R.B., and C.M.D. designed the experiments. M.M. performed the experiments and analyses. M.M. wrote the manuscript, and all other authors revised it. A.R.B. and C.M.D. supervised the work. All authors read and approved the final manuscript.

9 Acknowledgements

This work was supported by the Engineering and Physical Sciences Research council [EP/S024093/1] and Exscientia.

10 Competing Interests

The authors declare no competing interests.

11 Tables

Table 1: Breakdown of generated sets using SrIOP and ReCL in target similarity optimisation for (a) simple molecules and (b) complex molecules. For simple molecules, (a), a high similarity was all molecules with tanimoto similarity greater than 0.9. For complex molecules, (b), the threshold was a tanimoto similarity greater than 0.7. Diversity filters (DF) were used on the complex molecule.

(a)

Method	# High similarity	# Scaffolds
SrIOP	486	1
ReCL	448	2

(b)

Method	# High similarity	# Scaffolds
SrIOP	497	1
SrIOP (DF)	447	142
ReCL	0	0
ReCL (DF)	3	3

12 Figures Legends/Captions

Figure 1: Distribution of generated libraries for (a) cLogP optimisation and (b) number of HBA's optimisation. Each line corresponds to the property distribution of the molecules sampled by an agent trained with a reward range detailed in the legend. Both figures show that optimisation is possible within a specific range (e.g., 0-20 HBA's), outside this range optimisation fails. The model is unable to generate appropriate molecules, so the training data distribution is recreated.

Figure 2: TPSA distribution of molecules sampled from each intermediate (1-3) and final (4) agent trained during rIOP. We show that we can shift the distribution iteratively towards target property values. TPSA reward range for each step were (1) 100 − 150, (2) 150-200, (3) 200-250, (4) 250-300.

Figure 3: Comparison of SrIOP to REINVENT (STD) for generation of druglike molecules (a) without diversity filters, (b) with diversity filters. SrIOP (blue), standard (orange), prior (green) and training data (dotted) distributions for the generation of high QED molecules (QED > 0.90). Only low QED molecules (< 0.8) were used in training, then the QED reward range was increased each SrIOP step. Both methods can generate the desired molecules, however, SrIOP generates more molecules in the desired range in both cases.

Figure 4: Effects of SMILES choice on substructure generation performance using ReCL and SrIOP. For each molecule 5 sets of different intermediate SMILES were generated, then used during optimisation. Each SMILES variation at each step represented the same molecule. (a) and (c) are the total number of SMILES sampled from the final agent that included the target substructure (500 molecules were sampled). SrIOP (pink) and ReCL (blue). (b) and (d) are the total number of distinct scaffolds present in the successful samples. SrIOP (orange) and ReCL (green). (a) and (b) correspond to molecule A, while (c) and (d) correspond to molecule B. The figure shows that the SMILES choice directly affects optimisation performance and diversity of generated molecules. For example, optimisation of molecule A failed (no molecules matching the desired substructure) at least once with both methods, despite intermediate SMILES at each step across all sets representing the same structure.

Figure 5: High-level diagram displaying the architecture of the deep reinforcement learning model used by Popova et al and Olivecrona et al.[38,39] (a) Supervised learning language model. The prior model learns to generate novel SMILES from a large dataset of SMILES from ChEMBL.31 (b) Reinforcement Learning. The agent model (based on the prior model) is trained to generate SMILES that return a favourable reward.

13 References

1. Schneider, P. & Schneider, G. De Novo Design at the Edge of Chaos. *J. Med. Chem.* **59**, 4077–4086 (2016).
2. Waring, M. J. *et al.* An analysis of the attrition of drug candidates from four major pharmaceutical companies. *Nat. Rev. Drug Discov.* **14**, 475–486 (2015).
3. Hay, M., Thomas, D. W., Craighead, J. L., Economides, C. & Rosenthal, J. Clinical development success rates for investigational drugs. *Nat. Biotechnol.* **32**, 40–51 (2014).
4. Bunnage, M. E. Getting pharmaceutical R&D back on target. *Nat. Chem. Biol.* **7**, 335–339 (2011).
5. Hughes, J., Rees, S., Kalindjian, S. & Philpott, K. Principles of early drug discovery: Principles of early drug discovery. *Br. J. Pharmacol.* **162**, 1239–1249 (2011).
6. Bohacek, R. S., McMartin, C. & Guida, W. C. The art and practice of structure-based drug design: A molecular modeling perspective. *Med. Res. Rev.* **16**, 3–50 (1996).
7. Kim, S. *et al.* PubChem Substance and Compound databases. *Nucleic Acids Res.* **44**, D1202–D1213 (2016).
8. Romano, J. D. & Tatonetti, N. P. Informatics and Computational Methods in Natural Product Drug Discovery: A Review and Perspectives. *Front. Genet.* **10**, 368 (2019).
9. Lin, X., Li, X. & Lin, X. A Review on Applications of Computational Methods in Drug Screening and Design. *Molecules* **25**, (2020).
10. Besnard, J. *et al.* Automated design of ligands to polypharmacological profiles. *Nature* **492**, 215–220 (2012).
11. Gómez-Bombarelli, R. *et al.* Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Cent. Sci.* **4**, 268–276 (2018).
12. Stumpfe, D. & Bajorath, J. Similarity searching. *WIREs Comput. Mol. Sci.* **1**, 260–282 (2011).
13. Horvath, D. A Virtual Screening Approach Applied to the Search for Trypanothione Reductase Inhibitors. *J. Med. Chem.* **40**, 2412–2423 (1997).
14. Surabhi, S. & Singh, B. K. COMPUTER AIDED DRUG DESIGN: AN OVERVIEW. *J. Drug Deliv. Ther.* **8**, 504–509 (2018).
15. Segler, M. H. S., Kogej, T., Tyrchan, C. & Waller, M. P. Generating Focused Molecule Libraries for Drug Discovery with Recurrent Neural Networks. *ACS Cent. Sci.* **4**, 120–131 (2018).
16. Mauser, H. & Stahl, M. Chemical Fragment Spaces for de novo Design. *J. Chem. Inf. Model.* **47**, 318–324 (2007).
17. Hartenfeller, M., Proschak, E., Schüller, A. & Schneider, G. Concept of Combinatorial De Novo Design of Drug-like Molecules by Particle Swarm Optimization. *Chem. Biol. Drug Des.* **72**, 16–26 (2008).
18. Dey, F. & Caflisch, A. Fragment-Based de Novo Ligand Design by Multiobjective Evolutionary Optimization. *J. Chem. Inf. Model.* **48**, 679–690 (2008).
19. Elton, D. C., Boukouvalas, Z., Fuge, M. D. & Chung, P. W. Deep learning for molecular design - a review of the state of the art. (2019) doi:10.1039/C9ME00039A.
20. Baldi, P. Autoencoders, Unsupervised Learning, and Deep Architectures. in *Proceedings of ICML Workshop on Unsupervised and Transfer Learning* (eds. Guyon, I., Dror, G., Lemaire, V., Taylor, G. & Silver, D.) vol. 27 37–49 (PMLR, 2012).
21. Jin, W., Barzilay, R. & Jaakkola, T. Junction Tree Variational Autoencoder for Molecular Graph Generation. (2018) doi:10.48550/ARXIV.1802.04364.

22. Weininger, D. SMILES, a Chemical Language and Information System: 1: Introduction to Methodology and Encoding Rules. *J. Chem. Inf. Comput. Sci.* **28**, 31–36 (1988).
23. Lim, J., Ryu, S., Kim, J. W. & Kim, W. Y. Molecular generative model based on conditional variational autoencoder for de novo molecular design. *J. Cheminformatics* **10**, 31 (2018).
24. Goodfellow, I. J. *et al.* Generative Adversarial Networks. *ArXiv14062661 Cs Stat* (2014).
25. Putin, E. *et al.* Reinforced Adversarial Neural Computer for de Novo Molecular Design. *J. Chem. Inf. Model.* **58**, 1194–1204 (2018).
26. Guimaraes, G. L., Sanchez-Lengeling, B., Outeiral, C., Farias, P. L. C. & Aspuru-Guzik, A. Objective-Reinforced Generative Adversarial Networks (ORGAN) for Sequence Generation Models. (2017) doi:10.48550/ARXIV.1705.10843.
27. Vaswani, A. *et al.* Attention Is All You Need. (2017) doi:10.48550/ARXIV.1706.03762.
28. Grechishnikova, D. Transformer neural network for protein-specific de novo drug generation as a machine translation problem. *Sci. Rep.* **11**, 321 (2021).
29. Bagal, V., Aggarwal, R., Vinod, P. K. & Priyakumar, U. D. MolGPT: Molecular Generation Using a Transformer-Decoder Model. *J. Chem. Inf. Model.* **62**, 2064–2076 (2022).
30. Zheng, S. *et al.* Deep scaffold hopping with multimodal transformer neural networks. *J. Cheminformatics* **13**, 87 (2021).
31. He, J. *et al. Transformer Neural Network for Structure Constrained Molecular Optimization*. https://chemrxiv.org/engage/chemrxiv/article-details/60c7578c702a9b118b18cafe (2021) doi:10.26434/chemrxiv.14416133.v1.
32. Goldberg, Y. A Primer on Neural Network Models for Natural Language Processing. (2015) doi:10.48550/ARXIV.1510.00726.
33. Kotsias, P.-C. *et al.* Direct steering of de novo molecular generation with descriptor conditional recurrent neural networks. *Nat. Mach. Intell.* **2**, 254–265 (2020).
34. Bjerrum, E. J. & Threlfall, R. Molecular Generation with Recurrent Neural Networks (RNNs). (2017) doi:10.48550/ARXIV.1705.04612.
35. Arús-Pous, J. *et al.* Exploring the GDB-13 chemical space using deep generative models. *J. Cheminformatics* **11**, 20 (2019).
36. Arús-Pous, J. *et al.* Randomized SMILES strings improve the quality of molecular generative models. *J. Cheminformatics* **11**, 71 (2019).
37. Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.* **8**, 229–256 (1992).
38. Olivecrona, M., Blaschke, T., Engkvist, O. & Chen, H. Molecular de-novo design through deep reinforcement learning. *J. Cheminformatics* **9**, 48 (2017).
39. Popova, M., Isayev, O. & Tropsha, A. Deep reinforcement learning for de novo drug design. *Sci. Adv.* **4**, eaap7885 (2018).
40. Guo, J. *et al.* Improving de novo molecular design with curriculum learning. *Nat. Mach. Intell.* **4**, 555–563 (2022).
41. Soviany, P., Ionescu, R. T., Rota, P. & Sebe, N. Curriculum Learning: A Survey. (2021) doi:10.48550/ARXIV.2101.10382.
42. Krenn, M., Häse, F., Nigam, A., Friederich, P. & Aspuru-Guzik, A. Self-referencing embedded strings (SELFIES): A 100% robust molecular string representation. *Mach. Learn. Sci. Technol.* **1**, 045024 (2020).
43. O'Boyle, N. & Dalke, A. *DeepSMILES: An Adaptation of SMILES for Use in Machine-Learning of Chemical Structures*. https://chemrxiv.org/engage/chemrxiv/article-details/60c73ed6567dfe7e5fec388d (2018) doi:10.26434/chemrxiv.7097960.v1.
44. Gaulton, A. *et al.* ChEMBL: a large-scale bioactivity database for drug discovery. *Nucleic Acids Res.* **40**, D1100–D1107 (2012).
45. Landrum, G. RDKit: Open-source cheminformatics. (2006).
46. Ertl, P. & Schuffenhauer, A. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *J. Cheminformatics* **1**, 8 (2009).
47. Bemis, G. W. & Murcko, M. A. The Properties of Known Drugs. 1. Molecular Frameworks. *J. Med. Chem.* **39**, 2887–2893 (1996).
48. Polykovskiy, D. *et al.* Molecular Sets (MOSES): A Benchmarking Platform for Molecular Generation Models. *ArXiv181112823 Cs Stat* (2020).
49. Elman, J. L. Learning and development in neural networks: the importance of starting small. *Cognition* **48**, 71–99 (1993).
50. Blaschke, T. *et al.* REINVENT 2.0: An AI Tool for De Novo Drug Design. *J. Chem. Inf. Model.* **60**, 5918–5922 (2020).