DrugEx v2: *De Novo* Design of Drug Molecule by Pareto-based Multi-Objective Reinforcement Learning in Polypharmacology

Xuhan Liu[1], Kai Ye[2], Herman W. T. van Vlijmen[1,3], Michael T. M. Emmerich[4], Adriaan P. IJzerman[1], Gerard J. P. van Westen[1, *]

[1]Drug Discovery and Safety, Leiden Academic Centre for Drug Research, Einsteinweg 55, Leiden, The Netherlands

[2]School of electronics and information engineering, Xi'an Jiaotong University, 28 Xianning W Rd, Xi'an, China

[3]Janssen Pharmaceutica NV, Turnhoutseweg 30, B-2340, Beerse, Belgium

[4]Leiden Institute of Advanced Computer Science, Einsteinweg55, Leiden, The Netherlands

[*]To whom correspondence should be addressed: Gerard J. P. van Westen, Drug Discovery and Safety, Leiden Academic Centre for Drug Research, Einsteinweg 55, Leiden, The Netherlands. Tel: +31-71-527-3511. Email: gerard@lacdr.leidenuniv.nl.

Email Address of other authors: (1) Xuhan Liu: x.liu@lacdr.leidenuniv.nl; (2) Kai Ye: kaiye@xjtu.edu.cn; (3) Herman W. T. van Vlijmen: hvvlijme@its.jnj.com; (4) Michael T. M. Emmerich: m.t.m.emmerich@liacs.leidenuniv.nl; (5) Adriaan P. IJzerman: ijzerman@lacdr.leidenuniv.nl.

Abbreviations

ARs	Adenosine Receptors
DL	Deep Learning
MT-DNN	Multi-Task Deep Neural Network
ECFP	Extended Connectivity Fingerprint
EA	Evolutionary Algorithm
EDA	Estimation of Distribution Algorithm
GPCRs	G Protein-coupled Receptors
GRU	Gated Recurrent Unit
LSTM	Long Shot-Term Memory
QSAR	Quantitative Structure-Activity Relationship
RBF	Radial Basis Function
RMSE	Root Mean Square Error
ReLU	Rectified Linear Unit
RF	Random Forest
RL	Reinforcement Learning
RNNs	Recurrent Neural Networks
SVM	Support Vector Machine
t-SNE:	t-distributed Stochastic Neighbor Embedding

Abstract

In polypharmacology, ideal drugs are required to bind to multiple specific targets to enhance efficacy or to reduce resistance formation. Although deep learning has achieved breakthrough in drug discovery, most of its applications only focus on a single drug target to generate drug-like active molecules in spite of the reality that drug molecules often interact with more than one target which can have desired (polypharmacology) or undesired (toxicity) effects. In a previous study we proposed a new method named *DrugEx* that integrates an exploration strategy into RNN-based reinforcement learning to improve the diversity of the generated molecules. Here, we extended our *DrugEx* algorithm with multi-objective optimization to generate drug molecules towards more than one specific target (two adenosine receptors, A_1AR and A_{2A}AR, and the potassium ion channel hERG in this study). In our model, we applied an RNN as the *agent* and machine learning predictors as the *environment*, both of which were pre-trained in advance and then interplayed under the reinforcement learning framework. The concept of evolutionary algorithms was merged into our method such that *crossover* and *mutation* operations were implemented by the same deep learning model as the *agent*. During the training loop, the agent generates a batch of SMILES-based molecules. Subsequently scores for all objectives provided by the *environment* are used for constructing Pareto ranks of the generated molecules with non-dominated sorting and Tanimoto-based crowding distance algorithms. Here, we adopted GPU acceleration to speed up the process of Pareto optimization. The final reward of each molecule is calculated based on the Pareto ranking with the ranking selection algorithm. The agent is trained under the guidance of the reward to make sure it can generate more desired molecules after convergence of the training process. All in all we demonstrate generation of compounds with a diverse predicted selectivity profile toward multiple targets, offering the potential of high efficacy and lower toxicity.

Keywords: deep learning; adenosine receptors; cheminformatics; reinforcement learning; multi-objective optimization; exploration strategy.

Introduction

The ‘one drug, one target, one disease’ paradigm, which has dominated the field of drug discovery for many years, has made great contributions to drug development and the understanding of their molecular mechanisms of action [1]. However, this strategy is encountering problems due to the intrinsic promiscuity of drug molecules, *i.e.* recent studies showed that one drug molecule could interact with six protein targets on average [2]. Side effects of drugs caused by binding to unexpected off-targets are one of the main reasons of clinical failure of drug candidates and even withdrawal of FDA-approved novel drugs [3,4]. Up to now, more than 500 drugs have been withdrawn from the market due to fatal toxicity [5]. Yet, disease often results from the perturbation of biological systems by multiple genetic and/or environmental factors, thus complex diseases are more likely to require treatment through modulating multiple targets simultaneously. Therefore, it is crucial to shift the drug discovery paradigm to “polypharmacology” for many complex diseases [6,7].

In polypharmacology, ideal drugs are required to bind to multiple specific targets to enhance efficacy or to reduce resistance formation (in which case multiple targets can be multiple mutants of a single target) [8]. It has been shown that partial inhibition of a small number of targets can be more efficient than the complete inhibition of a single target, especially for complex and multifactorial diseases [6,9]. In parallel, common structural and functional similarity of proteins results in drugs binding to off-targets; therefore we also demand drugs to have a high target selectivity to avoid binding to unwanted target proteins. For example, the adenosine receptors (ARs) are a class of rhodopsin-like G protein-coupled receptors (GPCRs) having adenosine as the endogenous ligand. Adenosine and ARs are ubiquitously distributed throughout the human tissues, and their interactions trigger a wide spectrum of physiological and pathological functions. There are four subtypes of ARs, including A_1, A_{2A}, A_{2B} and A_3, each of which has a unique pharmacological profile, tissue distribution, and effector coupling [10,11]. The complexity of adenosine signaling and the widespread distribution of ARs have always given rise to challenges in developing target-specific drugs [12]. In addition to the similarity of the pharmacophores of some generic proteins (*e.g.* human Ether-à-go-go-Related Gene, hERG) should also be taken into consideration as they can be sensitive to binding exogenous ligands and cause side effects. hERG is the alpha subunit of a potassium ion channel [13] and has an inclination to interact with drug molecules because of its larger inner vestibule as the ligand binding pocket [14]. When hERG is inhibited this may cause long QT syndrome [15].

In addition to visual recognition, natural language processing and gaming, deep learning has been increasingly applied in drug discovery [16]. It does not only perform well in prediction models for virtual screening, but is also used to construct generative models for drug *de novo* design and/or drug optimization [17] . For example, our group implemented a fully-connected deep neural network (DNN) to construct a proteochemometric model (PCM) with all high quality ChEMBL data [18] for prediction of ligand bioactivity [19]. Its performance was shown to be better than other shallow machine learning methods. Moreover, we also developed a generative model with recurrent neural networks (RNNs), named *DrugEx* for SMILES-based *de novo* drug design [20]. It was shown that the generated molecules had large diversity and were similar to known ligands to some extent to make sure that reliable and diverse drug candidates can be designed.

Since the first version of *DrugEx* (*v1*) demonstrated effectiveness for designing novel A_{2A}AR ligands, we began to extend this method for drug design toward multiple targets. In this study, we updated *DrugEx* to the second version (*v2*) through merging crossover and mutation operations, which were derived from evolutionary algorithms, into the reinforcement learning (RL) framework. In order to evaluate the performance of our additions we tested our method into both multi-target and target-specific cases. For the multi-target case, desired molecules should have a high affinity towards both A_1AR and A_{2A}AR. In the target-specific case, on the other hand, we required molecules to have only high affinity towards the A_{2A}AR but a low affinity to the A_1AR for. In order to decrease toxicity and adverse events, molecules were additionally obliged to have a low affinity for hERG in both cases. It is worth noting that generated molecules should also be chemically diverse and have similar physico-chemical properties to known ligands. All python code for this study is freely available at http://github.com/XuhanLiu/DrugEx.

Materials and Methods

Data Source

Drug like molecules represented as SMILES format were downloaded from the ChEMBL database (version 26). After data preprocessing, including recombining charges, removing metals and small fragments, we collected 1.7 million molecules and named it the *ChEMBL* set, used for SMILES syntax learning. This data preprocessing step was implemented in RDKit [21]. Furthermore, 25,731 ligands were extracted from the ChEMBL database to construct the *LIGAND* set, which had bioactivity measurements towards the human A_1AR, A_{2A}AR, and hERG. The *LIGAND* set was used for constructing prediction models for each target and fine-tuning the generative models. The number of ligands and bioactivities for these three targets in the *LIGAND* set is represented in Table 1. Duplicate items were removed and if multiple measurements for the same ligands existed, the average pChEMBL value (pX, including pKi, pKd, pIC50, or pEC50) was calculated. To judge if a molecule is active or not, we defined the threshold of bioactivity as pX = 6.5. If the pX < 6.5, the compound was predicted as undesired (low affinity to the given target); otherwise, it was regarded as desired (having high affinity) [19].

Prediction Model

In order to predict the pX for each generated molecule for a given target, regression QSAR models were constructed with different machine learning algorithms. To increase the chemical diversity available for the QSAR model we included lower quality data without pChEMBL value, *i.e.* molecules that were labeled as “Not Active” or without a defined pX value. For these data points we defined a pX value of 3.99 (slightly smaller than 4.0) to eliminate the imbalance of the dataset and guarantee the model being able to predict the negative samples. During the training process, sample weights for low quality data were set as 0.1, while the data with exact pX were set as 1.0. This allowed us to particularly incorporate the chemical diversity, while avoiding degradation of model quality. Descriptors used as input were ECFP6 fingerprints [22] with 2048 bits (2048 dimensions, or 2048D) calculated by the RDKit Morgan Fingerprint algorithm (using a three-bond radius). Moreover, the following 19D physico-chemical descriptors were used: molecular weight, logP, number of H bond acceptors and donors, number of rotatable bonds, number of amide bonds, number of bridge head atoms, number of hetero atoms, number of spiro atoms, number of heavy atoms, the fraction of SP3 hybridized carbon atoms, number of aliphatic rings, number of saturated rings, number of total rings, number of aromatic rings, number of heterocycles, number of valence electrons, polar surface area and Wildman-Crippen MR value. Hence, each molecule in the dataset was transformed into a 2067D vector. Before being input into the model, the value of input vectors were normalized to the range of [0, 1] by the MinMax method. Model output value is the probability whether a given chemical compound was active based on this vector.

Table 1: The number of ligands and bioactivities for each of the human protein targets A_1AR, A_{2A}AR and hERG in the *LIGAND* set.

	A_1AR	A_{2A}AR	hERG
Total Ligands	7700	8406	16733
Bioactivities	13100	12129	22156
Active Ligands (pX >= 6.5)	1990	2511	924
Inactive Ligands (pX < 6.5)	1859	1709	6438
Inactive Ligands (No pX)	1764	1993	1275
Other Ligands	2087	4704	8906

Four algorithms were benchmarked for QSAR model construction, Random Forest (RF), Support Vector Machine (SVM), Partial Least Squares regression (PLS), and Multi-task Deep Neural Network (MT-DNN). RF, SVM and PLS models were implemented through Scikit-Learn [23], and the MT-DNN model through PyTorch [24]. In the RF, the number of trees was set as 1000 and split criterion was "gini". In the SVM, a radial basis function (RBF) kernel was used and the parameter space of C and γ were set as [2^{-5}, 2^{15}] and [2^{-15}, 2^{5}], respectively. In the MT-DNN, the architecture contained three hidden layers activated by a rectified linear unit (ReLU) between input and output layers, and the number of neurons were 2048, 4000, 2000, 1000 and 3 in these subsequent layers. The training process consisted of 100 epochs with 20% of hidden neurons randomly dropped out between each layer. The mean squared error was used to construct the loss function and was optimized by the Adam algorithm [25] with a learning rate of 10^{-3}.

Generative Model

As in *DrugEx v1*, we organized the vocabulary for the SMILES construction. Each SMILES-format molecule in the *ChEMBL* and *LIGAND* sets was split into a series of tokens. Then all tokens existing in this dataset were collected to construct the SMILES vocabulary. The final vocabulary contained 85 tokens (Table S1) which were selected and arranged sequentially into valid SMILES sequences through correct grammar.

The RNN model constructed for sequence generation contained six layers: one input layer, one embedding layer, three recurrent layers and one output layer. After being represented by a sequence of tokens, molecules can be received as categorical features by the input layer. In the embedding layer, vocabulary size, and embedding dimension were set to 85 and 128, meaning each token could be transformed into a 128 dimensional vector. For a recurrent layer, the long-short term memory (LSTM) was used as recurrent cell with 512 hidden neurons instead of the gated recurrent unit (GRU) [26] which was employed only in *DrugEx v1*. The output at each position was the probability that determined which token in the vocabulary would be chosen to grow the SMILES string.

During the training process we put a start token (GO) at the beginning of a batch of data as input and an end token (END) at the end of the same batch of data as output. This ensures that our generative network could choose correct tokens each time based on the sequence it had generated previously. A negative log likelihood function was used to construct the loss function to guarantee that the token in the output sequence had the largest probability to be chosen after being trained. In order to optimize the parameters of the model, the Adam algorithm [25] was used for the optimization of the loss function. Here, the learning rate was set at 10^{-3}, the batch size was 512, and training steps were set to 1000 epochs.

Reinforcement Learning

SMILES sequence construction under the RL framework can be viewed as a series of decision-making steps (Fig. 1). The generator (*G*) and the predictors (*Q*) are regarded as the policy and reward function, respectively. In this study we use multi-objective optimization (MOO), and each objective is a requirement to be achieved maximally for each scenario, albeit with differences in desirability. Our aim was defined by the following problem statement:

$$maximize\ R_1, \qquad maximize\ R_2, \qquad \dots , \qquad maximize\ R_n$$

Here, n equals the number of objectives (n = 3 in this study), and R_i, the score for each objective i, was calculated as follows:

$$R_i = \begin{cases} minmax(pX_i), & if\ high\ affinity\ required \\ 1 - minmax(pX_i), & if\ low\ affinity\ required \\ 0, & if\ SMILES\ invalid \end{cases}$$

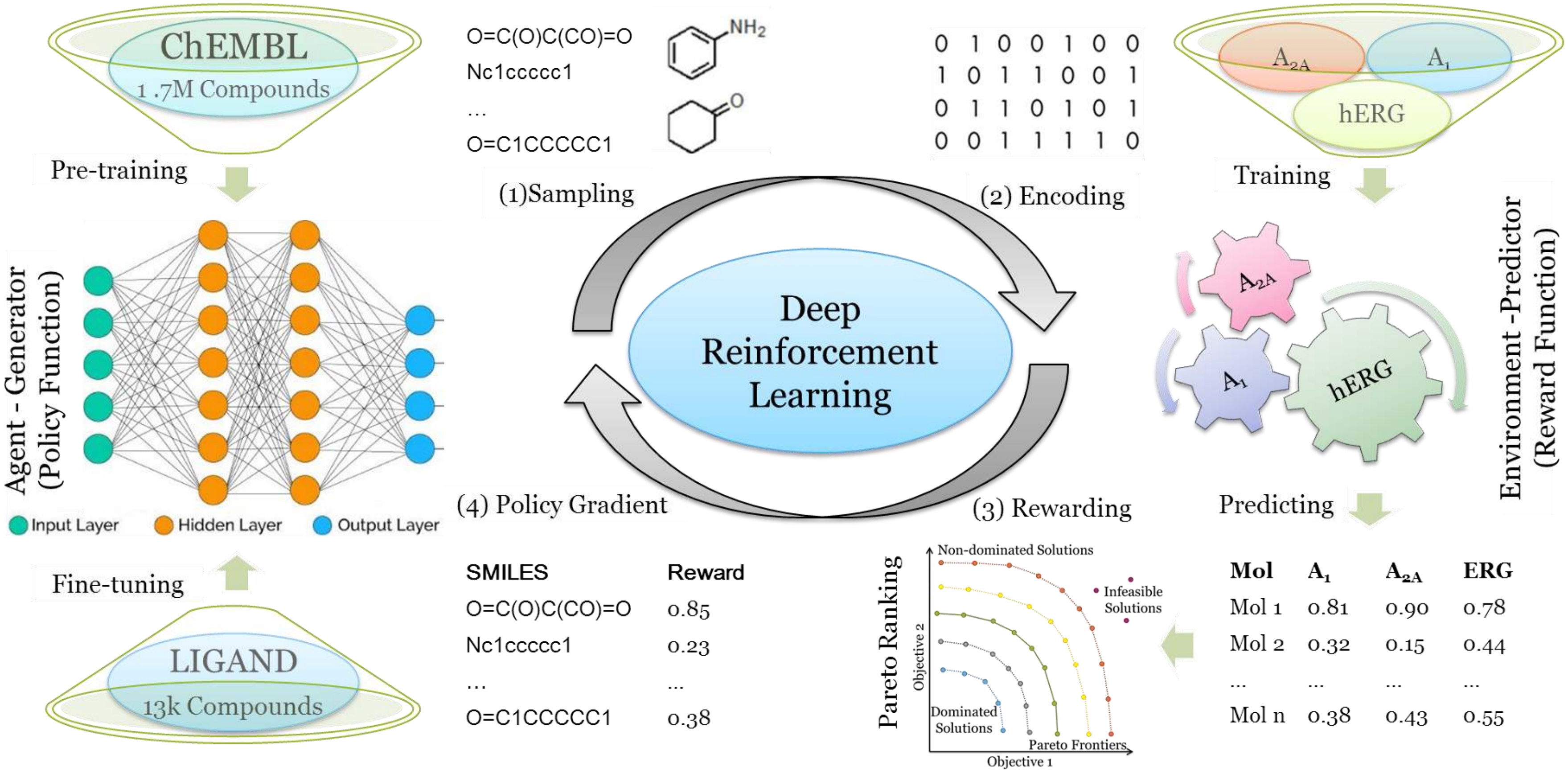


Fig. 1: **The workflow of the training process of our deep learning-based molecule generator *DrugEx2* utilizing reinforcement learning.** After the generator has been pre-trained/fine-tuned, (1) a batch of SMILES are generated by sampling tokens step by step based on the probability calculated by the generator; (2) These valid SMILES are parsed to be molecules and encoded into descriptors to get the predicted pXs with well-trained predictors; (3) The predicted pXs are transformed into a single value as the reward for each molecule based on Pareto optimization; (4) These SMILES sequences and their rewards are sent back to the generator for training with policy gradient methods. These four steps constitute the training loop of reinforcement learning.

here the pX_i (the range from 3.0 to 10.0) was the prediction score given by each predictor for the i^{th} target, which was normalized to the interval [0, 1] as the reward score. If having no or low affinity for a target was required (off-target) this score would be subtracted from 1 (inverting it).

In order to evaluate the performance of the generators, three coefficients are calculated with the generated molecules, including validity, desirability, and uniqueness which are defined as:

$$\text{Validity} = \frac{N_{valid}}{N_{total}}$$

$$\text{Desirability} = \frac{N_{desired}}{N_{total}}$$

$$\text{Uniqueness} = \frac{N_{unique}}{N_{total}}$$

where N_{total} is the total number of molecules, N_{valid} is the number of the molecules parsed by the valid SMILES sequences, N_{unique} is the number of molecules which are different from others in the dataset, and $N_{desired}$ is the number of desired molecules. Here, we determine if generated molecules are desired based on the reward R_i if all of them are larger than the threshold (0.5 by default when pX = 6.5). In addition, we calculated SA score (from 1 to 10) for each molecule to measure the synthesizability of which larger value means more difficult to be synthesized. And we also computed QED (from 0 to 1) score to evaluate the drug-likeness of which larger value means more drug-like for each molecule. The calculation of both SA and QED scores were implemented by RDKit.

To orchestrate and combine these different objectives, we compared two different reward schemes: the Pareto front (PF) scheme and the weighted sum (WS) scheme. These were defined as follows:

(a) Weighted sum (WS) scheme: the weight for each function is not fixed but dynamic, and depends on the desired ratio for each objective, which is defined as:

$$\mathrm{r}_i = \frac{N_i^s}{N_i^l}$$

here for objective *i* the N_i^s and N_i^l are the number of generated molecules which have a score smaller or larger than the threshold. Moreover, the weight is normalized ratio defined as:

$$w_i = \frac{r_i}{\sum_{k=1}^{M} r_k}$$

and the final reward R^* was calculated by

$$R^* = \sum_{i=1}^{n} w_i R_i ,$$

(b) Pareto front (PF) scheme: operates on the desirability score, which is defined as

$$\mathrm{D}_i = \begin{cases} 1, & if\ R_i > t_i \\ R_i/t_i, & if\ R_i \le t_i \end{cases}$$

where t_i is the threshold of the i^{th} objective, and we set all of objectives had the same threshold as 0.5 as stated in the methods. Given two solutions m_1 and m_2 with their scores $(x_1, x_2, ..., x_n)$ and $(y_1, y_2, ..., y_n)$, then m_1 is said to Pareto dominate m_2 if and only if:

$$\forall\ \mathrm{j} \in \{1, \dots, \mathrm{n}\}: x_j \ \ge y_j\ and\ \exists\ \mathrm{j} \in \{1, \dots, \mathrm{n}\}: x_j > y_j$$

otherwise, m_1 and m_2 are non-dominated with each other. After the dominance between all pair of solutions being determined, the non-dominated scoring algorithm [27] is exploited to obtain a rank of Pareto frontiers which consist of a set of solutions. The solutions in the top frontier are dominated by the other solutions in the bottom frontier, but the solutions in the same frontier are non-dominated with each other [28]. In order to speed up the non-dominated sorting algorithm, we employed *PyTorch* to implement this procedure with GPU acceleration. After obtaining the frontiers ranking from dominated solutions to dominant solutions, the molecules were ranked based on the average of Tanimoto-distance instead of crowding distance with other molecules in the same frontier, and molecules with smaller distances were ranked on the top. The final reward R^* is defined as:

$$\mathrm{R}_i^* = \begin{cases} 0.5 + \dfrac{k - N_{undesired}}{2N_{desired}}, & if\ desired \\ \dfrac{k}{2N_{undesired}}, & if\ undesired \end{cases}$$

here the parameter *k* is the index of the solution in the Pareto rank, and rewards of undesired and desired solutions will be evenly distributed in (0, 0.5] and (0.5, 0.1], respectively.

During the generation process, for each step, *G* determines the probability of each token from the vocabulary to be chosen based on the generated sequence in previous steps. Its parameters are updated by employing a policy gradient based on the expected end reward received from the predictor. The objective function is designated as follows:

$$J(\theta) = \mathbb{E}[R^*(y_{1:T})|\theta] = \sum_{t=1}^{T} logG(y_t|y_{1:t-1}) \cdot R^*(y_{1:T})$$

By maximizing this function, the parameters θ in *G* can be optimized to ensure that *G* can construct desired SMILES sequences which can obtain the highest reward scores judged by all the *Qs*.

Algorithm extrapolation

Evolutionary algorithms (EAs) are common methods used in drug discovery [29]. For example, *Molecule Evoluator* is one of EAs, with mutation and crossover operations based on SMILES representation [30] for drug *de novo* design. In addition, some groups also proposed other variations of EAs [31], e.g., estimation of distribution algorithm (EDA) which is a model-based method and replaces the *mutation* and *crossover* operations with probability distribution estimation and sampling of new individuals (Fig. 2) [32]. Similar to EDA, *DrugEx* is a model-based method too, in which the deep learning model was employed to estimate the probability distribution of sequential decision making. However, we use a DL method to define model-based *mutation* and *crossover* operations. Moreover, we employed an RL method to replace the sample selection step for the update of model or population in EDA or EA, respectively.

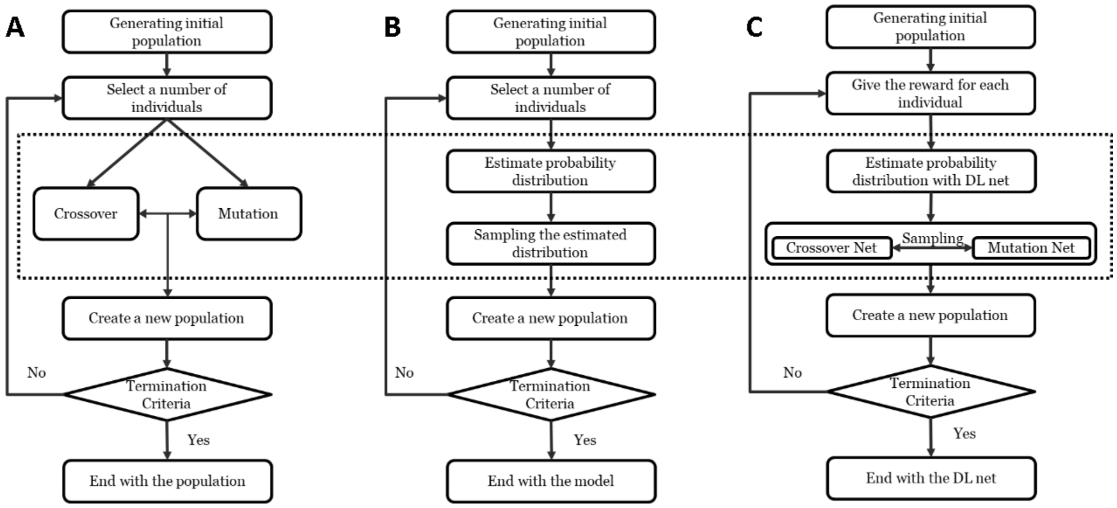

A
Generating initial population
Select a number of individuals
Crossover
Mutation
Create a new population
No
Termination Criteria
Yes
End with the population
B
Generating initial population
Select a number of individuals
Estimate probability distribution
Sampling the estimated distribution
Create a new population
No
Termination Criteria
Yes
End with the model
C
Generating initial population
Give the reward for each individual
Estimate probability distribution with DL net
Crossover Net
Sampling
Mutation Net
Create a new population
No
Termination Criteria
Yes
End with the DL net

Fig. 2: Flowchart comparison of evolutionary algorithm (A), estimation of distribution algorithm (B) and our proposed method (C).

Exploration Strategy

In our previous study, we had implemented the exploration strategy through importing a fixed exploration net to enlarge the diversity of the generated molecules during the training loops. In this study, we continued to extend the methods of this exploration strategy, which resemble the *crossover* and *mutation* operations from evolutionary algorithms (EAs). Here, besides the *agent* net (G_A), we also defined exploration strategy with two other DL models: *crossover* net (G_C) and *mutation* net (G_M), which have the same RNN architecture (Fig. 3). Before the training process, they were initialized by a pre-trained or fine-tuned model. The G_M was the basic strategy employed in the previous version and its parameters were fixed and not updated during the whole training process. The G_C implemented in this work was an extended strategy whose parameters were updated iteratively based on the G_A. During the training process, each SMILES sequence was generated through combining these three RNNs: for each step, a random number from 0 to 1 is generated. If it is larger than the mutation rate (ε), the probability for token sampling is controlled by the combination of G_A and G_C, otherwise, it is determined by G_M. For each training loop, only the parameters in G_A were updated instantly based on the gradient of the RL objective function. An iteration was defined as the period of epochs after the desirability score of molecules generated by G_A did not increase. Subsequently the parameters of G_C were updated with G_A directly and the training process continued for the next iteration. The training process would continue till the percentage of desired molecules in the current iteration was not better than in the previous iterations.

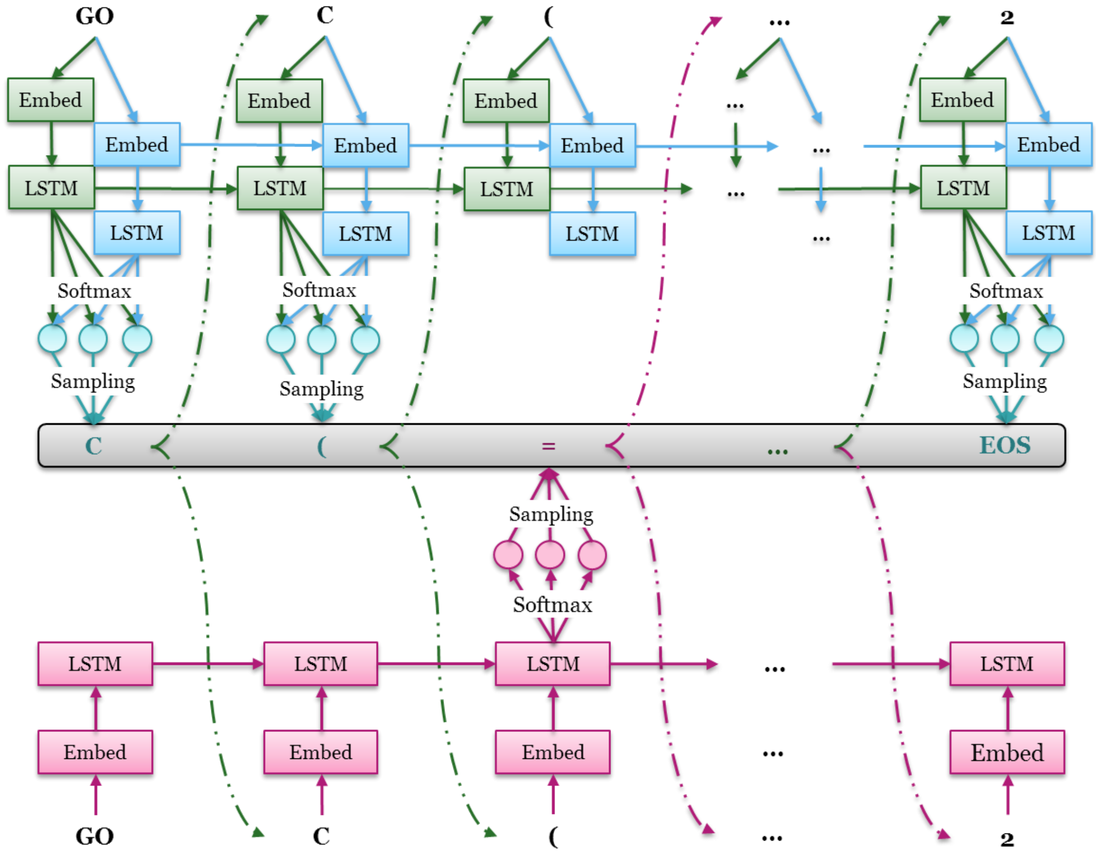


Fig. 3: The mechanism of updated exploration strategy, including agent net G_A, mutation net G_M (red) and crossover net G_C (blue). In the training loop, G_M is fixed, G_C is updated iteratively and G_A is trained at each epoch. For each position, a random number from 0 to 1 is generated. If it is larger than the mutation rate (ε), the probability for token sampling is controlled by the combination of G_A and G_C, otherwise, it is determined by G_M.

Molecular Diversity

To measure molecular diversity, we adopted the metric proposed by Solow and Polasky in 1994 to estimate the diversity of a biological population in an eco-system [33]. It has been shown to be an effective method to measure the diversity of drug molecules [34]. The formula to calculate diversity was redefined to normalize the range of values from [1, m] to (0, m] as follows:

$$I(A) = \frac{1}{|A|} \boldsymbol{e}^{\intercal} F(\boldsymbol{s})^{-1} \boldsymbol{e}$$

where A is a set of drug molecules with a size of $|A|$ equal to m, $\boldsymbol{e}$ is an m-vector of 1's and $F(\boldsymbol{s}) = [f(d_{ij}))]$ is a non-singular $m \times m$ distance matrix, in which $f(d_{ij})$ stands for the distance function of each pair of molecule provided as follows:

$$f(d) = e^{-\theta d_{ij}}$$

here we defined the distance d_{ij} of molecules s_i and s_j by using the Tanimoto-distance with ECFP6 fingerprints as follows:

$$d_{ij} = d(s_i, s_j) = 1 - \frac{|s_i \cap s_j|}{|s_i \cup s_j|},$$

where $|\, s_i \cap s_j \,|$ represents the number of common fingerprint bits, and $|\, s_i \cup s_j \,|$ is the number of union fingerprint bits.

Results and Discussion

Performance of Predictors

All molecules in the *LIGAND* set were used to train the QSAR models, after being transformed into predefined descriptors, including 2048D ECFP6 fingerprints and 19D physicochemical properties. We then tested the performance of these different algorithms with five-fold cross validation and an independent test of which the performances are shown in Fig. 4AB. Here, the dataset was randomly split into five folds in the cross validation, while a temporal split with a cut-off at the year of 2015 was used for the independent test. In the cross validation test, the MT-DNN model achieved the highest value for R^2 and the lowest RMSE value for A_1AR and $A_{2A}AR$, but the RF model had the best performance for hERG based on R^2 and RMSE. However, for the independent test the RF model reached the highest R^2 and lowest RMSE across the board, although it was worse than the performance in the cross-validation test. A detailed performance overview of the RF model is shown in Fig. 4C-E. Because the generative model might create a large number of novel molecules, which would not be similar to the molecules in the training set, we took the robustness of the predictor into consideration. In this situation the temporal split has been shown to be more robust [19,35]. Hence the RF algorithm was chosen for constructing our environment which provides the final reward to guide the training of the generator in RL.

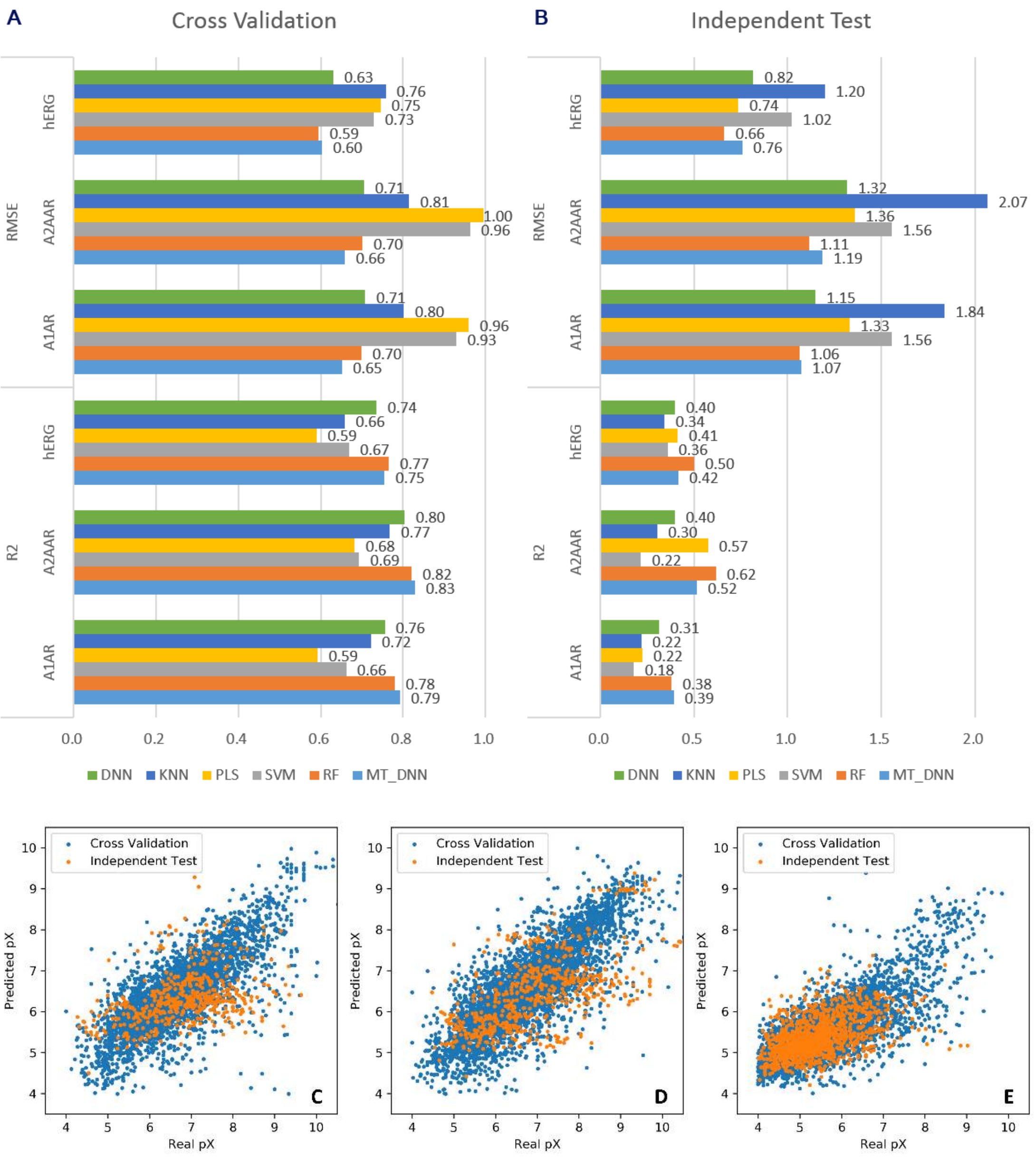


Fig. 4: Performance comparison of different machine learning regression models. In these two histograms (A-B), the results were obtained based on five-fold cross validation (A) and independent test (B) for the three targets. The R^2 and RMSE scores were used to evaluate the performance of different machine learning models including DNN, KNN, PLS, SVM RF and MT-DNN. In the scatter plots (C-E), each point stands for one molecule with its real pX (*x*-axis) and the predicted pX (*y*-axis) by the RF model which was chosen as the final predictors for A_1AR (C), A_{2A}AR (D) and hERG (E) based on five-fold cross validation (blue) and independent test (orange).

Model optimization

As in our previous work in *DrugEx v1*, we firstly pre-trained and fine-tuned the generator with the *ChEMBL* and *LIGAND* set, respectively. When testing the different types of RNNs, we analyzed the performance of the pre-trained model with 10,000 SMILES generated, and found that LSTM generated more valid SMILES (97.5%) than GRU (93.1%) which had been adopted in our previous work. Moreover, for the fine-tuning process, we split the *LIGAND* set into two subsets: training set and validation set; the validation set was not involved in parameters updating but it was essential to avoid model overfitting and to improve uniqueness of generated molecules. Subsequently 10,000 SMILES were sampled for performance evaluation. We found that the percentage valid SMILES was 97.9% for LSTM, larger than GRU with 95.7% valid SMILES, a slight improvement compared to the pre-trained model. In the end, we employed the LSTM-based pre-trained/fine-tuned models for the following investigation.

We employed the models for two cases (multi-target and target-specific) of multi-objective drug design towards three protein targets. During the training loop of *DrugEx v2*, the parameter of $\boldsymbol{\varepsilon}$ was set to different values: 10^{-2}, 10^{-3}, 10^{-4} and we also tested it without mutation net, *i.e.* the value of $\boldsymbol{\varepsilon}$ was set to 0. Generators were trained by using a policy gradient with two different rewarding schemes. After the training process converged, 10,000 SMILES were generated for each model for performance evaluation. The percentage of valid, desired, unique desired SMILES and the diversity were calculated (Table 2). Furthermore, we also compared the chemical space of these generated molecules with known ligands in the *LIGAND* set. Here, we employed first two components of t-SNE on the ECFP6 descriptors of these molecules to represent the chemical space.

Performance comparisons

We compared the performance of *DrugEx v2* with *DrugEx v1* and two other DL-based *de novo* drug design methods: *REINVENT* [36] and *ORGANIC* [37]. In order to make a fair benchmark, we trained these four methods with the same environments to provide the unified predicted bioactivity scores for each of the generated molecules. It should be mentioned that these methods are all SMILES-based RNNs generators but trained under different RL frameworks. Therefore, these generators were constructed with the same RNN structures of and initialized with the same pre-trained/fine-tuned models.

In the WS scheme we did not choose fixed weights for objectives but dynamic values which can be adjusted automatically during the training process. The reason for this is that if the fixed weights should be optimized as the hyperparameters, which would be more time consuming. Moreover, the distribution of scores for each objective was not comparable. If the affinity score was required to be higher, few of the molecules generated by the model with initial state were satisfactory, but if a lower affinity score was required, most of the generated molecules by the pre-trained/fine-tuned model met this need without further training of RL. Therefore, weights were set as dynamic parameters and determined by the ratio between desired and undesired molecules generated by the model at the current training step. This approach ensures that the objectives with lower scores would get more importance than others during the training loop to balance the different objectives and generate more desired molecules.

The performance of the model with different ε is shown in Table S2. A higher ε generates molecules with larger diversity but low desirability compared to a lower ε in both multi-target and target-specific cases. In addition, an appropriate ε guarantees the model generates molecules which have a more similar distribution of important substructures with the desired ligands in the *LIGAND* set. With the WS scheme, the model generates molecules with a high desirability, but the diversity is lower than the desired ligands in the training set. On the contrary, the PF scheme helped the model generate molecules with a larger diversity than the ligands in the training set, but the desirability was not as high as in the WS rewarding scheme. Moreover, the generated molecules in the PF scheme have more similar distribution of substructures to the *LIGAND* set than in the WS scheme.

Table 2: Comparison of validity, desirability, uniqueness and substructure distributions of SMILES generated by four different methods in the multi-target case with PF and WS rewarding schemes, respectively. For the validity, desirability and uniqueness, the largest data is bold, while for the distribution of substructures, the bold data are labeled as the most closed to the values in the *LIGAND* set.

Rewarding Scheme	Dataset	Validity	Desirability	Uniqueness	Diversity	Purine Ring	Furan Ring	Benzene Ring
	LIGAND	100.00%	12.40%	100.00%	0.66	21.30%	35.44%	79.24%
PF	*DrugEx v1*	98.28%	43.27%	88.96%	0.71	**17.37%**	41.05%	80.95%
	DrugEx v2	99.57%	**80.81%**	87.29%	0.7	13.97%	**32.01%**	**80.26%**
	ORGANIC	**98.84%**	66.01%	82.67%	0.65	17.27%	56.38%	68.87%
	REINVENT	99.54%	57.43%	**98.84%**	**0.77**	0.64%	40.38%	92.05%
WS	*DrugEx v1*	97.76%	38.44%	93.44%	0.71	**10.76%**	**36.42%**	86.99%
	DrugEx v2	**99.80%**	**97.45%**	89.08%	0.49	3.63%	21.06%	96.18%
	ORGANIC	99.08%	61.10%	77.65%	0.68	9.08%	70.99%	**83.91%**
	REINVENT	99.54%	70.98%	**99.11%**	**0.71**	0.04%	23.23%	96.28%

In the multi-target case, these four methods with different rewarding schemes show similar performance, *i.e.* the WS scheme can help models improve the desirability while the PF scheme assists models to achieve better diversity and distribution of substructures (Table 2). Here, *REINVENT* with the PF scheme achieved the largest diversity, whereas *DrugEx v1* had the most similar substructure distribution to the molecules in the *LIGAND* set, and *DrugEx v2* achieved the best desirability with both PR and WS schemes compared to the three other algorithms. The diversity and distribution of substructures were also most similar to the best results. In addition, in the target-specific case results were similar to the multi-target case, (Table 3), and for the distribution of purine and furan rings, *DrugEx v2* surpassed v1 to be most similar to the *LIGAND* set. When investigating the SA and QED scores, we observed that PF scheme helped all of generated molecules being more drug-like because of higher QED scores than WS scheme in both multi-target case (Fig. 6A-D) and target-specific case (Fig. 6E-H). In comparison of these methods, the molecules generated by *REINVENT* were supposedly easier to be synthesized and more drug-like than others, but the molecules of *DrugEx v1* had more similar distributions with the molecules in the *LIGAND* set.

Table 3: Comparison of validity, desirability, uniqueness and substructure distributions of SMILES generated by four different methods in the target-specific case with PF and WS rewarding schemes, respectively. For the validity, desirability and uniqueness, the largest data is bold, while for the distribution of substructures, the bold data are labeled as the most closed to the values in the *LIGAND* set.

Rewarding Scheme	Dataset	Validity	Desirability	Uniqueness	Diversity	Purine Ring	Furan Ring	Benzene Ring
	LIGAND	100.00%	14.63%	100.00%	0.67	28.27%	50.61%	71.84%
PF	*DrugEx v1*	98.07%	48.42%	87.32%	0.73	**29.65%**	61.61%	**70.99%**
	DrugEx v2	99.53%	**89.49%**	90.55%	0.73	23.73%	**56.23%**	67.40%
	ORGANIC	98.29%	86.98%	80.30%	0.64	10.60%	89.27%	65.28%
	REINVENT	**99.59%**	70.66%	**99.33%**	**0.79**	3.85%	33.82%	92.53%
WS	*DrugEx v1*	97.61%	44.96%	95.89%	**0.68**	78.92%	**80.21%**	68.02%
	DrugEx v2	**99.62%**	**97.86%**	90.54%	0.31	19.58%	98.56%	51.87%
	ORGANIC	98.97%	88.14%	84.13%	0.49	9.68%%	96.66%	**71.48%**
	REINVENT	99.55%	81.27%	98.87%	0.34	**25.13%**	97.52%	74.61%

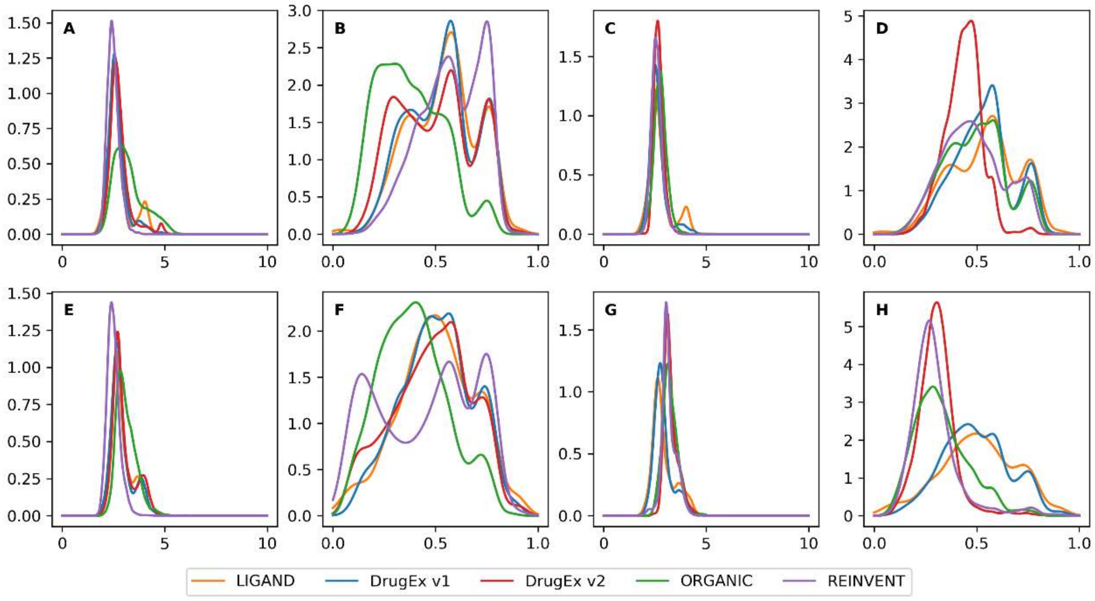


Fig. 5: the distribution of SA score and QED score of desired ligands in the *LIGAND* set and of molecules generated by four different methods with PR (A, B, E and F) and WS (C, D, G and H) rewarding schemes in the multi-target case (A-D) and target-specific case (E-H). The

molecules from the *LIGAND* set were shown as color of orange, and the molecules generated by *DrugEx v1, v2, ORGANIC* and *REINVENT* were represented with colors of blue, green, red, and purple, respectively. Overall DrugEx v1 and v2 are better able to emulate the observed distributions in the training set compared to *ORGANIC* and *REINVENT*.

With respect to chemical space, we employed t-SNE with the ECFP6 descriptors of all molecules for both multi-target (Fig. 6A-H) and target-specific cases (Fig. 6I-P). In the multi-target case, most of desired ligands in the *LIGAND* set were distributed in the margin and PR scheme could guide all of the generators to search more regions than WS scheme. In the target-specific case, the desired ligands in the *LIGAND* set were distributed more dispersed in both of the margin and the center regions. However, PF scheme was not shown the similar results as in the target-specific case to improve the coverage compared with WS scheme except for *DrugEx v2*. For both of these two cases, only part of the region occupied by desired ligands in the *LIGAND* set were overlapped with *REINVENT* and *ORGANIC*, but almost all of it is covered by *DrugEx v1* and *v2*. Especially, in contrast to WS scheme *DrugEx v2* had a significant improvement of chemical space coverage with PF scheme. A possible reason is that the molecules generated by *DrugEx v1* and *v2* offer a more similar distribution of substructures to desired ligands in the *LIGAND* set than *REINVENT* and *ORGANIC*.

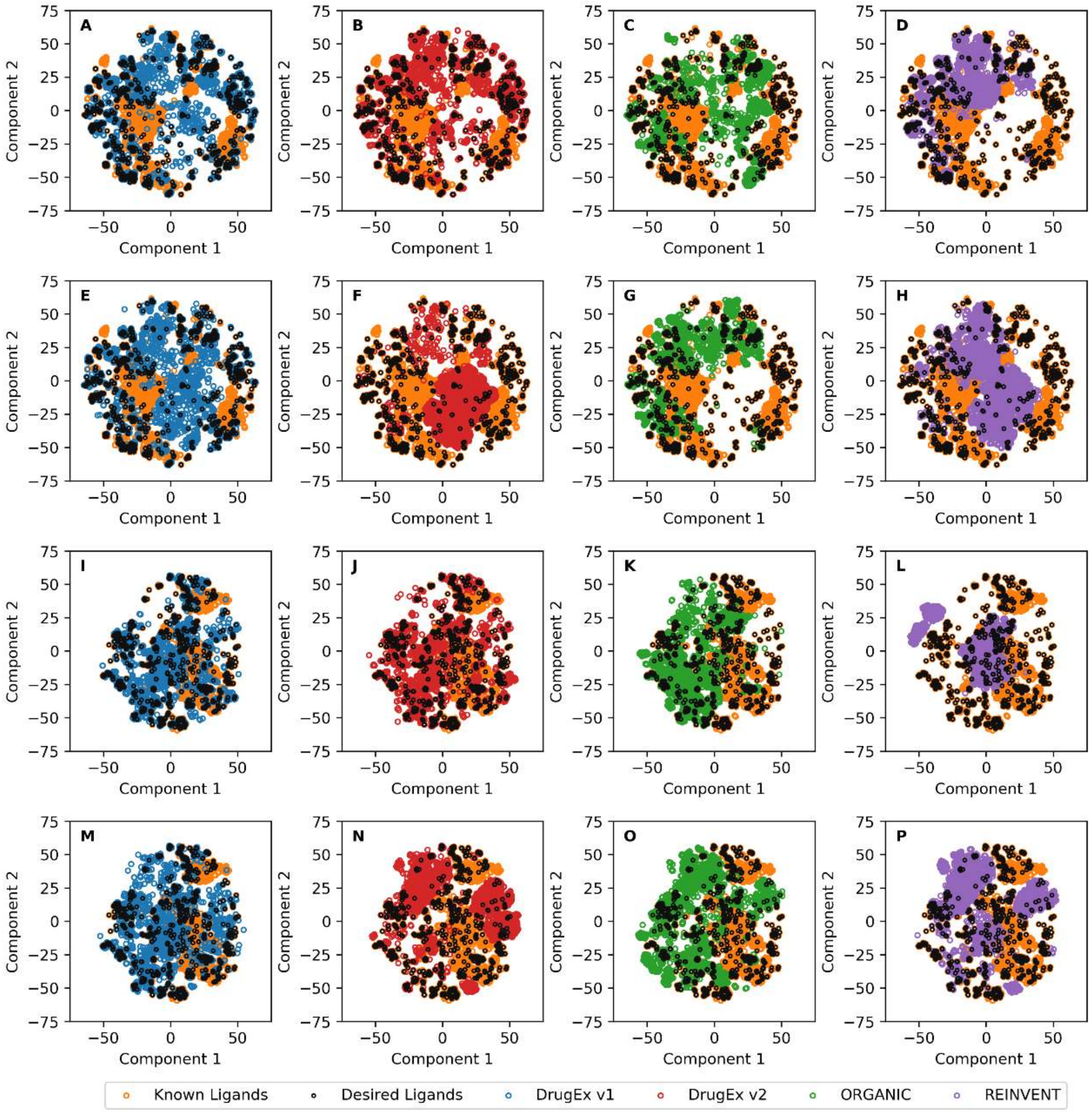
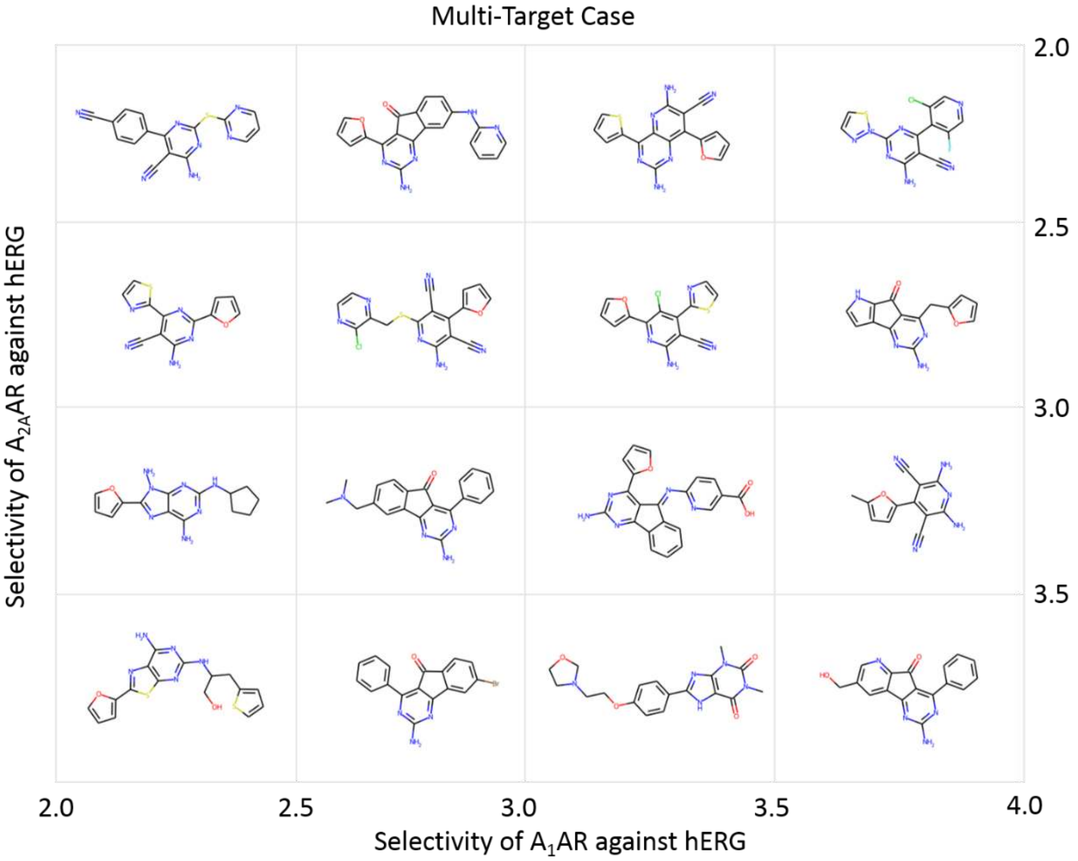


Fig. 6: Comparison of the chemical space of ligands in the *LIGAND* set (orange for all molecules and black for desired molecules) and of generated molecules by *DrugEx v1* (A, E, I, M, blue), *v2* (B, F, J, N, red)*, ORGANIC* (C, G, K, O, green) and *REINVENT* (D, H, L, P, purple), for the multi-target case (A-H) and target specific case (I-P). Chemical space is represented by the first two components in t-SNE with ECFP6 descriptors of molecules. The first and third rows were obtained with PF rewarding scheme, and the second and fourth rows were obtained with WS rewarding scheme. Similar to our previous work it can be seen that DrugEx better covers the whole chemical space of the input data. In particular in the multi-target case with a pareto optimization based scoring function (E-H) the improved coverage in all sections, including isolated active ligands, becomes clear.

As an example, 16 possible antagonists (without ribose moiety and molecular weight < 500) generated by *DrugEx v2* with PR scheme were selected as candidates for both multi-target cases and target specific case, respectively. These molecules were ordered by the selectivity which was calculated as the difference of pXs between two different protein targets. In the multi-target cases (Fig. 7A), because the desired ligands prefer A_1AR and $A_{2A}AR$ to hERG, the row and column is the selectivity of $A_{2A}AR$ and A_1AR against hERG, respectively, while the generated molecules are required to bind only $A_{2A}AR$ rather than A_1AR and hERG in the target-specific case (Fig. 7B), selectivity of $A_{2A}AR$ against A_1AR and hERG were represented as the row and column, respectively.


Multi-Target Case
2.0
2.5
3.0
3.5
Selectivity of A2AAR against hERG
2.0
2.5
3.0
3.5
4.0
Selectivity of A1AR against hERG

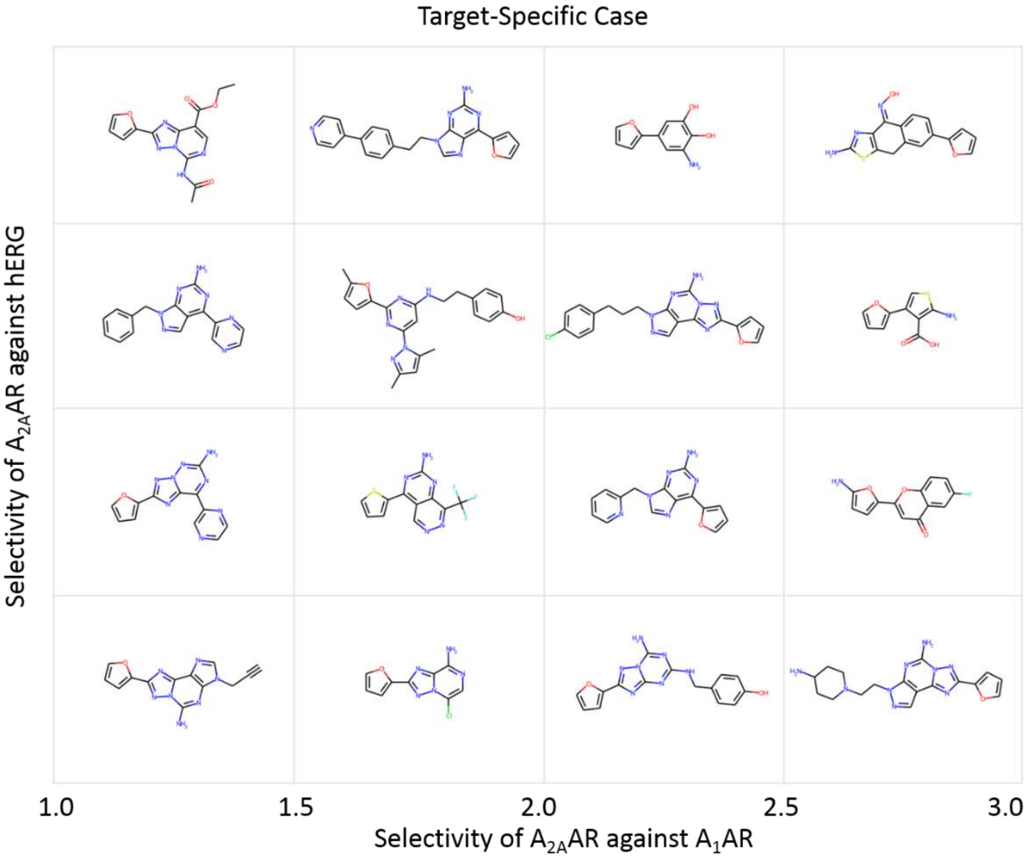

Target-Specific Case
2.0
2.5
3.0
3.5
4.0
1.0
1.5
2.0
2.5
3.0
Selectivity of A_{2A}AR against hERG
Selectivity of A_{2A}AR against A_1AR

Fig. 7: Some candidate molecules were selected from molecules generated by *DrugEx v2* with PR scheme for both multi-target case and target-specific case. In multi-target case (A), these molecules were ordered by the selectivity of A_1AR and A_{2A}AR against hERG as *x*-axis and *y*-axis, respectively. In target-specific case (B), these molecules were ordered by the selectivity of A_{2A}AR against A_1AR and hERG as *x* and *y*-axis, respectively.

In order to prove the effectiveness of our proposed method, we tested it with 20 goal-directed molecule generation tasks on the GuacaMol benchmark platform [38]. These tasks contain different requirements, including similarity, physicochemical properties, isomerism, scaffold matching, *etc.* The detailed description of these tasks is provided in ref [38] and our results are shown in Table S3. We pre-trained our model with the dataset provided by the GuacaMol platform, in which all molecules from the ChEMBL database are included and similar molecules to the target ligands in the tasks were removed. Then we choose the top 1024 molecules in the training set to fine-tune our model for each task, before reinforcement learning was started. Our method scores the best in 12 out of 20 tasks compared with the baseline models provided by the GuacaMol platform, leading to an overall second place. Moreover, the performance between the LSTM benchmark method and our methods were similar in these tasks, possibly because they have similar architectures of neural networks. All in all, this benchmark demonstrated that our proposed method has improved generality for drug *de novo* design tasks. It is worth being mentioned that our method is not effective enough yet for some tasks of contradictory objectives in the narrow chemical space. The main reason is that our method emphasizes to obtain a large number of feasible molecules to occupy the diverse chemical space rather than small number of optimal molecules to achieve the highest score. For example, in the *Sitagliptin MPO task*, the aim is finding molecules which are dissimilar to sitagliptin but have a similar molecular formula to sitagliptin, and our method was not as good as Graph GA, which is a graph-based genetic algorithm.

Conclusion and Future Prospects

In this work, we proposed a Pareto-based multi-objective learning algorithm for drug *de novo* design towards multiple targets based on different requirements of affinity scores for multiple targets. We transferred the concept of an evolutionary algorithm (including mutation and crossover operations) into RL to update *DrugEx* for multi-objective optimization. In addition, Pareto ranking algorithms were also integrated into our model to handle the contradictory objectives common in drug discovery and enlarge the chemical diversity. In order to prove effectiveness, we tested the performance of *DrugEx v2* in both multi-target and target-specific cases. We found that a large percentage of generated SMILES were valid and desired molecules without many duplications. Moreover, the generated molecules were also similar to known ligands and covered almost every corner of the chemical space that known ligands occupy, which could not be repeated by tested competing methods. In future work, we will try the generality of our proposed methods with different molecular representations, such as graphs or fragments [29]. We will also integrate more objectives (e.g. stability, synthesizability), especially when these objectives are contradictory, such that the model allows user-defined weights for each objective to generate more reliable candidate ligands and better steer the generative process.

Authors' Contributions

XL and GJPvW conceived the study and performed the experimental work and analysis. KY, APIJ, ME and HWTvV provided feedback and critical input. All authors read, commented on and approved the final manuscript.

Acknowledgements

XL thanks Chinese Scholarship Council (CSC) for funding, GJPvW thanks the Dutch Research Council and Stichting Technologie Wetenschappen (STW) for financial support (STW-Veni #14410).

Competing Interests

The authors declare that they have no competing interests

Reference

1. Chaudhari R, Tan Z, Huang B, Zhang S (2017) Computational polypharmacology: a new paradigm for drug discovery. Expert Opin Drug Discov 12 (3):279-291. doi:10.1080/17460441.2017.1280024
2. Giacomini KM, Krauss RM, Roden DM, Eichelbaum M, Hayden MR, Nakamura Y (2007) When good drugs go bad. Nature 446 (7139):975-977. doi:10.1038/446975a
3. Lounkine E, Keiser MJ, Whitebread S, Mikhailov D, Hamon J, Jenkins JL, Lavan P, Weber E, Doak AK, Cote S, Shoichet BK, Urban L (2012) Large-scale prediction and testing of drug activity on side-effect targets. Nature 486 (7403):361-367. doi:10.1038/nature11159
4. Cook D, Brown D, Alexander R, March R, Morgan P, Satterthwaite G, Pangalos MN (2014) Lessons learned from the fate of AstraZeneca's drug pipeline: a five-dimensional framework. Nat Rev Drug Discov 13 (6):419-431. doi:10.1038/nrd4309
5. Siramshetty VB, Nickel J, Omieczynski C, Gohlke BO, Drwal MN, Preissner R (2016) WITHDRAWN--a resource for withdrawn and discontinued drugs. Nucleic Acids Res 44 (D1):D1080-1086. doi:10.1093/nar/gkv1192
6. Hopkins AL (2008) Network pharmacology: the next paradigm in drug discovery. Nat Chem Biol 4 (11):682-690. doi:10.1038/nchembio.118
7. Anighoro A, Bajorath J, Rastelli G (2014) Polypharmacology: challenges and opportunities in drug discovery. J Med Chem 57 (19):7874-7887. doi:10.1021/jm5006463
8. van Westen GJ, Wegner JK, Geluykens P, Kwanten L, Vereycken I, Peeters A, IJzerman AP, van Vlijmen HW, Bender A (2011) Which compound to select in lead optimization? Prospectively validated proteochemometric models guide preclinical development. PLoS One 6 (11):e27518. doi:10.1371/journal.pone.0027518
9. Csermely P, Agoston V, Pongor S (2005) The efficiency of multi-target drugs: the network approach might help drug design. Trends Pharmacol Sci 26 (4):178-182. doi:10.1016/j.tips.2005.02.007
10. Fredholm BB (2010) Adenosine receptors as drug targets. Exp Cell Res 316 (8):1284-1288. doi:10.1016/j.yexcr.2010.02.004
11. Fredholm BB, IJzerman AP, Jacobson KA, Linden J, Muller CE (2011) International Union of Basic and Clinical Pharmacology. LXXXI. Nomenclature and classification of adenosine receptors--an update. Pharmacol Rev 63 (1):1-34. doi:10.1124/pr.110.003285
12. Chen JF, Eltzschig HK, Fredholm BB (2013) Adenosine receptors as drug targets--what are the challenges? Nat Rev Drug Discov 12 (4):265-286. doi:10.1038/nrd3955
13. Trudeau MC, Warmke JW, Ganetzky B, Robertson GA (1995) HERG, a human inward rectifier in the voltage-gated potassium channel family. Science 269 (5220):92-95. doi:10.1126/science.7604285
14. Milnes JT, Crociani O, Arcangeli A, Hancox JC, Witchel HJ (2003) Blockade of HERG potassium currents by fluvoxamine: incomplete attenuation by S6 mutations at F656 or Y652. Br J Pharmacol 139 (5):887-898. doi:10.1038/sj.bjp.0705335
15. Sanguinetti MC, Tristani-Firouzi M (2006) hERG potassium channels and cardiac arrhythmia. Nature 440 (7083):463-469. doi:10.1038/nature04710
16. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521 (7553):436-444. doi:10.1038/nature14539

17. Chen H, Engkvist O, Wang Y, Olivecrona M, Blaschke T (2018) The rise of deep learning in drug discovery. Drug discovery today. doi:10.1016/j.drudis.2018.01.039

18. Gaulton A, Bellis LJ, Bento AP, Chambers J, Davies M, Hersey A, Light Y, McGlinchey S, Michalovich D, Al-Lazikani B, Overington JP (2012) ChEMBL: a large-scale bioactivity database for drug discovery. Nucleic Acids Res 40 (Database issue):D1100-1107. doi:10.1093/nar/gkr777

19. Lenselink EB, Ten Dijke N, Bongers B, Papadatos G, van Vlijmen HWT, Kowalczyk W, IJzerman AP, van Westen GJP (2017) Beyond the hype: deep neural networks outperform established methods using a ChEMBL bioactivity benchmark set. Journal of cheminformatics 9 (1):45. doi:10.1186/s13321-017-0232-0

20. Liu X, Ye K, van Vlijmen HWT, IJzerman AP, van Westen GJP (2019) An exploration strategy improves the diversity of de novo ligands using deep reinforcement learning: a case for the adenosine A2A receptor. Journal of cheminformatics 11 (1):35. doi:10.1186/s13321-019-0355-6

21. RDKit: Open-Source Cheminformatics Software. http://www.rdkit.org.

22. Rogers D, Hahn M (2010) Extended-connectivity fingerprints. Journal of chemical information and modeling 50 (5):742-754. doi:10.1021/ci100050t

23. Scikit-Learn: machine learning in Python. http://www.scikit-learn.org/.

24. PyTorch. https://pytorch.org/.

25. Kingma DP, Ba J (2014) Adam: A Method for Stochastic Optimization. arXiv:1412.6980

26. Chung J, Gulcehre C, Cho K, Bengio Y (2014) Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. ArXiv:1412.3555

27. Deb K, Agrawal S, Pratap A, Meyarivan T A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II. In: Schoenauer M, Deb K, Rudolph G et al. (eds) Parallel Problem Solving from Nature PPSN VI, Berlin, Heidelberg, 2000// 2000. Springer Berlin Heidelberg, pp 849-858

28. Emmerich MTM, Deutz AH (2018) A tutorial on multiobjective optimization: fundamentals and evolutionary methods. Nat Comput 17 (3):585-609. doi:10.1007/s11047-018-9685-y

29. Liu X, IJzerman AP, van Westen GJP (2021) Computational Approaches for De Novo Drug Design: Past, Present, and Future. Methods Mol Biol 2190:139-165. doi:10.1007/978-1-0716-0826-5_6

30. Lameijer EW, Kok JN, Back T, IJzerman AP (2006) The molecule evoluator. An interactive evolutionary algorithm for the design of drug-like molecules. Journal of chemical information and modeling 46 (2):545-552. doi:10.1021/ci050369d

31. van der Horst E, Marques-Gallego P, Mulder-Krieger T, van Veldhoven J, Kruisselbrink J, Aleman A, Emmerich MT, Brussee J, Bender A, IJzerman AP (2012) Multi-objective evolutionary design of adenosine receptor ligands. Journal of chemical information and modeling 52 (7):1713-1721. doi:10.1021/ci2005115

32. Nicolaou CA, Brown N (2013) Multi-objective optimization methods in drug design. Drug Discov Today Technol 10 (3):e427-435. doi:10.1016/j.ddtec.2013.02.001

33. Solow AR, Polasky S (1994) Measuring biological diversity. Environmental and Ecological Statistics 1 (2):95-103. doi:10.1007/BF02426650

34. Yevseyeva I, Lenselink EB, de Vries A, IJzerman AP, Deutz AH, Emmerich MTM (2019) Application of portfolio optimization to drug discovery. Information Sciences 475:29-43.

doi:https://doi.org/10.1016/j.ins.2018.09.049

35. Sheridan RP (2013) Time-split cross-validation as a method for estimating the goodness of prospective prediction. Journal of chemical information and modeling 53 (4):783-790. doi:10.1021/ci400084k

36. Olivecrona M, Blaschke T, Engkvist O, Chen H (2017) Molecular de-novo design through deep reinforcement learning. Journal of cheminformatics 9 (1):48. doi:10.1186/s13321-017-0235-x

37. Benjamin S-L, Carlos O, Gabriel L. G, Alan A-G (2017) Optimizing distributions over molecular space. An Objective-Reinforced Generative Adversarial Network for Inverse-design Chemistry (ORGANIC). doi:10.26434/chemrxiv.5309668.v3

38. Brown N, Fiscato M, Segler MHS, Vaucher AC (2019) GuacaMol: Benchmarking Models for de Novo Molecular Design. Journal of chemical information and modeling 59 (3):1096-1108. doi:10.1021/acs.jcim.8b00839

Table S1: All tokens in vocabulary for SMILES sequence construction with RNN model.

Atoms							Bonds	Controls		
Common Atoms					Aromatic Atoms		--	Rings	Branchs	On-Off
B	[Ag-3]	[CH-]	[N]	[SH2]	[b-]	[se+]	-	1	(	GO
C	[As+]	[CH2]	[O+]	[SH]	[c+]	[se]	=	2	)	EOS
F	[As]	[CH]	[O-]	[Se+]	[c-]	[te+]	#	3		
I	[B-]	[I+]	[OH+]	[SeH]	[cH-]	[te]		4		
L	[BH-]	[IH2]	[O]	[Se]	[n+]	b		5		
N	[BH2-]	[N+]	[P+]	[SiH2]	[n-]	c		6		
O	[BH3-]	[N-]	[PH]	[SiH]	[nH+]	n		7		
P	[B]	[NH+]	[S+]	[Si]	[nH]	o		8		
R	[C+]	[NH-]	[S-]	[Te]	[o+]	p		9		
S	[C-]	[NH2+]	[SH+]		[s+]	s				

Considering that the sterochemical information of molecules and ionic bonds were ignored, we removed the “@”, “\”, “/”, “.”.

Table S2: Comparison of validity, desirability, uniqueness and substructures distributions of SMILES generated by *DrugEx v2* with different ε in the multi-target and target-specific cases by using PF and WS rewarding schemes, respectively. For the validity, desirability and uniqueness, the largest data is bold, while for the distribution of substructures, the bold data are labeled as the most closed to the values in the *LIGAND* set.

Case	**Reward**	**Dataset**	**Validity**	**Desirability**	**Uniqueness**	**Diversity**	**Purine**	**Furan**	**Benzene**
	Scheme	**/ ε**					**Ring**	**Ring**	**Ring**
Multi-Target Case		*LIGAND*	100.00%	14.63%	100.00%	0.67	21.30%	35.44%	79.24%
	PF	10^{-2}	99.39%	71.37%	**90.47%**	**0.72**	12.39%	34.69%	82.05%
		10^{-3}	99.57%	80.81%	88.96%	0.71	**13.97%**	32.01%	**80.26%**
		10^{-4}	**99.72%**	**83.86%**	87.19%	0.71	12.45%	30.58%	84.04%
		0	99.47%	73.76%	84.41%	0.70	13.35%	**35.71%**	81.89%
	WS	10^{-2}	99.54%	87.56%	93.08%	**0.60**	**9.66%**	**28.83%**	92.19%
		10^{-3}	**99.80%**	97.45%	93.44%	0.49	3.63%	21.06%	96.18%
		10^{-4}	99.79%	**98.15%**	**93.56%**	0.53	2.89%	24.95%	**91.46%**
		0	99.78%	98.00%	90.19%	0.49	5.02%	16.45%	96.77%
Target-Specific Case		*LIGAND*	100.00%	12.40%	100.00%	0.66	28.27%	50.61%	71.84%
	PF	10^{-2}	99.48%	88.76%	**91.98%**	**0.77**	18.31%	**47.50%**	68.95%
		10^{-3}	99.53%	89.49%	87.32%	0.72	23.73%	56.23%	67.40%
		10^{-4}	**99.55%**	**91.84%**	88.31%	0.74	**26.86%**	39.68%	74.36%
		0	99.54%	91.47%	88.94%	0.75	22.95%	43.08%	**71.50%**
	WS	10^{-2}	99.16%	86.45%	93.97%	**0.42**	**42.84%**	97.26%	**72.45%**
		10^{-3}	99.62%	**97.86%**	**95.89%**	0.31	60.81%	98.56%	51.87%
		10^{-4}	**99.67%**	96.82%	94.56%	0.34	55.14%	**93.69%**	45.40%
		0	99.33%	96.28%	92.60%	0.35	42.86%	98.34%	63.47%

Table S3: Results of the Goal-Directed tasks for our proposed method *DrugEx v2* and other baseline models on GuacaMol Benchmark. GucacaMol platform contains 20 tasks with different requirements, including smilarity, physicochemical properties, isomerism, scaffold matching, *etc.*. The results for baseline models were cited from ref [38]. The bold data are shown as the best result for each task achieved by different methods.

Benchmark	Best of Dataset	SMILES GA	Graph MCTS	Graph GA	SMILES LSTM	DrugEx v2
Celecoxib rediscovery	0.505	0.732	0.355	**1**	**1**	**1**
Troglitazone rediscovery	0.419	0.515	0.311	**1**	**1**	**1**
Thiothixene rediscovery	0.456	0.598	0.311	**1**	**1**	**1**
Aripiprazole similarity	0.595	0.834	0.38	**1**	**1**	**1**
Albuterol similarity	0.719	0.907	0.749	**1**	**1**	**1**
Mestranol similarity	0.629	0.79	0.402	**1**	**1**	**1**
C11H24	0.684	0.829	0.41	0.971	**0.993**	**0.993**
C9H10N2O2PF2Cl	0.747	0.889	0.631	0.982	0.879	**1**
Median molecules 1	0.334	0.334	0.225	0.406	**0.438**	0.418
Median molecules 2	0.351	0.38	0.17	0.432	0.422	**0.435**
Osimertinib MPO	0.839	0.886	0.784	0.953	0.907	**0.967**
Fexofenadine MPO	0.817	0.931	0.695	**0.998**	0.959	0.942
Ranolazine MPO	0.792	0.881	0.616	**0.92**	0.855	0.909
Perindopril MPO	0.575	0.661	0.385	0.792	0.808	**0.812**
Amlodipine MPO	0.696	0.722	0.533	0.894	0.894	**0.898**
Sitagliptin MPO	0.509	0.689	0.458	**0.891**	0.545	0.517
Zaleplon MPO	0.547	0.413	0.488	**0.754**	0.669	0.693
Valsartan SMARTS	0.259	0.552	0.04	**0.99**	0.978	0.978
Scaffold Hop	0.933	0.97	0.59	**1**	0.996	0.989
Deco Hop	0.738	0.885	0.478	**1**	0.998	0.986
Total	12.144	14.398	9.011	**17.983**	17.341	17.537

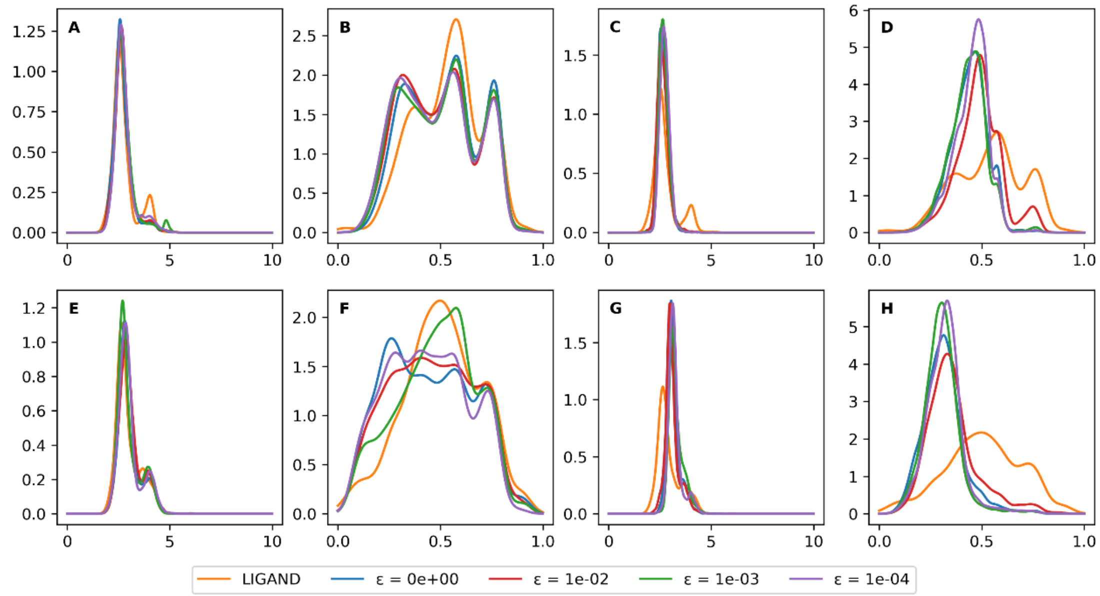


Fig. S1: the distribution of SA score and QED score of desired ligand in the *LIGAND* set and molecules generated by *DrugEx v2* with different *ε* in the multi-target case (A-D) and target-specific case (E-H) by using PR (A, B, E and F) and WS (C, D, G and H) rewarding schemes.