Diversity Oriented Deep Reinforcement Learning for Targeted Molecule Generation

Tiago Pereira
University of Coimbra Centre for Informatics and Systems: Universidade de Coimbra Centro de Informatica e Sistemas https://orcid.org/0000-0003-2487-0097

Maryam Abbasi (maryam@dei.uc.pt)
University of Coimbra Centre for Informatics and Systems: Universidade de Coimbra Centro de Informatica e Sistemas https://orcid.org/0000-0002-9011-0734

Bernardete Ribeiro
University of Coimbra Centre for Informatics and Systems: Universidade de Coimbra Centro de Informatica e Sistemas https://orcid.org/0000-0002-9770-7672

Joel P. Arrais
University of Coimbra Centre for Informatics and Systems: Universidade de Coimbra Centro de Informatica e Sistemas https://orcid.org/0000-0003-4937-2334

Research article

Keywords: Drug Design, SMILES, Reinforcement Learning, RNN

Posted Date: November 24th, 2020

DOI: https://doi.org/10.21203/rs.3.rs-110570/v1

Version of Record: A version of this preprint was published on March 9th, 2021. See the published version at https://doi.org/10.1186/s13321-021-00498-z.

RESEARCH

Diversity Oriented Deep Reinforcement Learning for Targeted Molecule Generation

Tiago Pereira, Maryam Abbasi*, Bernardete Ribeiro and Joel P. Arrais

*Correspondence:
maryam@dei.uc.pt
University of Coimbra, Centre for Informatics and Systems of the University of Coimbra, Department of Informatics Engineering, Pinhal de Marrocos, Portugal, PT
Full list of author information is available at the end of the article

Abstract

In this work, we explore the potential of deep learning to streamline the process of identifying new potential drugs through the computational generation of molecules with interesting biological properties. Two deep neural networks compose our targeted generation framework: *the Generator*, which is trained to learn the building rules of valid molecules employing SMILES strings notation, and *the Predictor* which evaluates the newly generated compounds by predicting their affinity for the desired target. Then, the Generator is optimized through Reinforcement Learning to produce molecules with bespoken properties.

The innovation of this approach is the exploratory strategy applied during the reinforcement training process that seeks to add novelty to the generated compounds. This training strategy employs two Generators interchangeably to sample new SMILES: the initially trained model that will remain fixed and a copy of the previous one that will be updated during the training to uncover the most promising molecules. The evolution of the reward assigned by the Predictor determines how often each one is employed to select the next token of the molecule. This strategy establishes a compromise between the need to acquire more information about the chemical space and the need to sample new molecules, with the experience gained so far.

To demonstrate the effectiveness of the method, the *Generator* is trained to design molecules with high inhibitory power for the adenosine A_{2A} and κ opioid receptors. The results reveal that the model can effectively modify the biological affinity of the newly generated molecules towards the craved direction. More importantly, it was possible to find auspicious sets of unique and diverse molecules, which was the main purpose of the newly implemented strategy.

Keywords: Drug Design; SMILES; Reinforcement Learning; RNN

Introduction

Drug development is a process that aims to bring new drugs into the market. This task is extremely complicated from both technical and financial perspectives since it comprises several challenging stages until reaching the final objective [1]. Also, the success rate is quite low, which means that some long-term research projects may end up in inglorious efforts. Nonetheless, as new diseases arise or new ways of treatment for the existing conditions are explored, it's evident that there is a need for an efficient and reliable drug development pipeline.

Several computational strategies have been used to make the process more efficient and less likely to fail, taking into account the challenges of the task. At the one hand, the search space of chemical compounds can be reduced using the virtual screening. This technique reduces the search space of chemical libraries by filtering

a set of molecules, successively, according to the desired properties [2]. Nonetheless, this approach is largely dependent on the size and diversity of the initial set of molecules [3]. On the other hand, there are computational techniques for the *de novo* drug design which involve exploring the chemical space for the generation of new compounds from scratch, in an automated way [4]. Initially, the most successful algorithms included atom-based elongation or fragment-based combination and were often coupled with Evolutionary Algorithms (EA) or other global optimization techniques [5, 6]. However, recent developments in deep learning (DL) have broadened the area of de novo molecule generation. As a result, it became a problem of inverse design in which the desirable properties are previously defined and then, via Reinforcement Learning (RL) or other optimization methods, the chemical space that satisfies those properties is explored. In this regard, these techniques have been successfully applied to hit discovery [7].

In 2009, Nicolau et al. have designed new molecules combining evolutionary techniques with graph-theory to perform a global search for promising molecules [8]. The findings obtained here have demonstrated the applicability of these methods, and it's usefulness for in-vitro molecular design.

More recently, some RL-based methods have also been widely employed in the drug discovery process. On that account, Benjamin et al. have explored the combination of a Generative Adversarial Network (GAN) with RL to perform a biased molecular generation in a work named ORGANIC [9]. Other variants of RL methods, such as the REINFORCE, have also been recently applied in de novo drug design with encouraging results, showing that deep generative models are very effective in modelling the Simplified Molecular Input Line Entry Specification (SMILES) representation of molecules using Recurrent Neural Networks (RNNs). Olivecrona et al. have combined RNNs and RL in a work named REINVENT to generate molecules containing specific biological or chemical properties in the form of SMILES through learning an augmented episodic likelihood composed by a prior likelihood and a user-defined scoring function [10]. Also, Popova et al. have implemented a model consisting of an RNN with stack-augmented memory as a computational generator and a Quantitative Structure-Activity Relationship (QSAR) model to estimate the properties to be optimized by RL, both based on SMILES notation [11].

Other RL methods such as Deep Q-Learning has also proven to be a successful way of research. In 2019, Zhou et al. designed new molecules with specific desired properties, formalizing the problem as a Markov Decision Process (MDP) and using a value function to solve it [12]. This method has achieved comparable performance against several other recently published algorithms for de novo molecular design.

Nevertheless, the computational generation of the lead compounds must always include specific key properties. On the one hand, these molecular generative models must produce candidate molecules that are biologically active against the desired target and safe for the organism [13]. On the other hand, it is no less important to guarantee the chemical validity of the generated molecules and also their chemical and physical diversity [14, 7]. In general, the works mentioned above are successful both in optimizing specific single molecular properties and also in generating chemically valid molecules. However, diversity is often neglected in lead design methods, even though it is an essential feature to ensure the novelty and the interest of the

new compounds [7]. In this regard, Liu et al. have implemented a generative model for the discovery of new drug-like molecules active against the adenosine A_{2A} receptors ($A_{2A}R$) combining RNNs, RL and an exploration strategy to ensure greater chemical diversity in the obtained compounds [15]. The latter procedure aimed to increase the diversity of molecules through the alternated use of two computational generators: one initially trained, which remains fixed and the other, updated at each iteration through RL.

Therefore, the problem faced by Liu et al. and, which we will address in this work, is the inefficient coverage of the chemical space when searching for new putative potential drugs. Often these computational methods show an inability to generate a set of molecules that have drug-like properties and, at the same time, substantial novelty compared to the already existing molecules [7, 16]. As a consequence, the goal is to obtain a set of biologically interesting molecules, that contains as much diversity as possible - both internal diversity and, ideally, diversity comparing with prevailing solutions. This novelty is essential for drug candidate molecules since it's only by fulfilling this prerequisite that it's possible to discover alternative therapeutic solutions better than the existing ones [17, 15, 13]. Additionally, another issue to answer is the inability of these generative models to maintain the percentage of valid molecules after changing the distribution of the biological property of interest [11].

Our solution is an end-to-end deep RL model for targeted molecular generation, implemented with RNNs and a policy gradient REINFORCE algorithm [18]. As a practical example, we are implementing a framework for generating lead compounds of interest, in the form of SMILES notation, that can bind to interesting receptors such as $A_{2A}R$ [19] or KOR [20]. Therefore, we propose a new strategy to balance the exploration/exploitation dilemma, based on the two Generators methodology founded in the work of Liu et al. but with a valuable distinctiveness. In this scenario, there are two possible generators: one of the models is more involved with exploration, and the other is more focused on exploitation. Then, the decision of which one will be employed to predict the next token is based on the previous evolution of the numerical reward.

In addition, to prevent the repetitive generation of molecules, we create a memory cell and update it with the last generated molecules so that there will be a penalty for the reward whenever the diversity of this set of molecules decreased during the process of exploring chemical space.

Notwithstanding, besides the computational generator, this framework is composed of a QSAR model for predicting the affinity of the newly generated molecules against the desired target. During the development of the QSAR, different architectures and molecular descriptors have been tested to obtain a robust model. Even though this work is directed at a specific targets, it can be easily adapted to different goals with biological interest.

Methods

The proposed framework can be divided into two main parts. First, two deep neural networks are implemented using supervised learning: the generative and the QSAR models. The former will be trained to learn the building rules of molecules using

SMILES notation, and, the latter will be trained to predict specific biological activities of compounds. Both are built with recurrent models, LSTM and GRU cells, respectively, and using SMILES strings as input data. In the second step, the Generator will be re-trained through RL, and the Predictor will guide the training process to guarantee the generation of molecules with the desired property optimized. In this work, we are introducing a new strategy for the selection of tokens in the generation of SMILES strings and an approach to enhance novelty in the compounds. The goal is, therefore, to generate valid molecules with promising properties and, at the same time, to ensure a satisfactory diversity after application of RL to bias the Generator towards the desired purpose. Figure 1 describes the general workflow of the framework to perform a targeted lead generation.

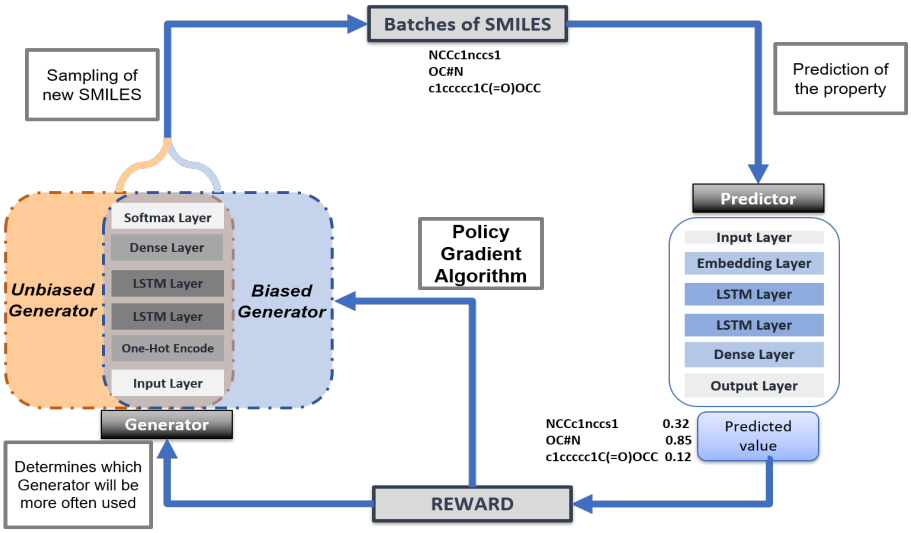


Figure 1 General overview of the framework: Two generators sharing the same architecture and Predictor interconnected by Reinforcement Learning.

Generator

The input data for this model are SMILES strings. Hence, it is necessary to perform some encoding to transform each sequential or structural character into a numerical value, capable of being used by the model. Data pre-processing starts by doing its tokenization, followed by the padding and, finally, by transforming it to one-hot encoding vectors. Tokenization involves the conversion of each atom or bonds character into a char type (token). The vocabulary used in the construction of SMILES strings contained 45 tokens. Then, to standardize all the strings, a starting and ending characters were added. The padding of the sequences ensures that all SMILES strings have 65 tokens. In this case, the starting character is 'G', the ending is 'E', and the padding is the space character. Finally, the SMILES strings are transformed into a one-hot encoding vector.

The architecture is similar to the one seen in the work of Gupta et al ([21]). It included an input layer, two LSTM layers with 256 units and 0.3 for the dropout value, applied between each layer. The dropout application can be seen as a regularization strategy that helps to minimize the learning inter-dependency and enhance

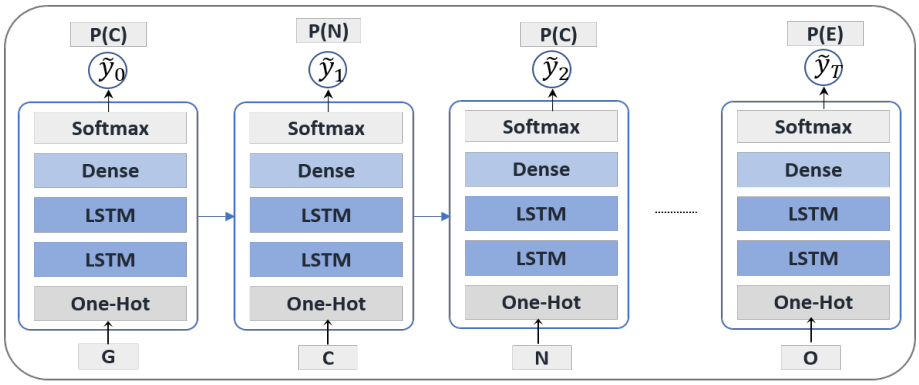


Figure 2 Flowchart for the training procedure for the SMILES string 'GCNC(C)=OE'. A vectorized token of the molecule is input as x_t in a time step t, and the probability of the output to x_{t+1} as the next token is maximized.

the generalization of the model. Also, it has a densely connected layer with 43 units and a neuron unit with a Softmax activation function. Data was divided into batches of 16 SMILES, during 25 training iterations and the optimizer employed to update the weights was Adam with a learning rate of 0.001.

The "Teacher Forcing" algorithm is used during training. It means that the correct token is always inserted in the next input, which minimizes the maximum-likelihood loss at each training step [22]. Consider $y = \{y_1, y_2, \ldots, y_T\}$ as the correct output sequence for a given input sequence X, and $\hat{y}$ is the Generator output vector. Suppose we have T samples with each sample indexed by $t = 1, \ldots, T$. The goal of training is to minimize the following cross-entropy loss function:

$$J(\theta) = -\frac{1}{T}\sum_{t=1}^{T}[y_t \log \hat{y}_t + (1 - y_t)\log(1 - \hat{y}_t)] \tag{1}$$

The loss function (Equation 1) is the negative log-likelihood ratio between the correct and competing tokens to guarantee that, after the training step, the outputted token is more likely to be chosen in future generations steps. The gradients are computed according to the clipping gradient method to avoid instability during training [23]. The combination of RNNs and sequential data such as SMILES notation as brought successful results in several fields, including in the generation of syntactically valid molecules [11]. This capability to learn the rules and dependencies inherent in the process of building molecules is explained by the ability of this type of architecture to learn essential input sections and retain them in their long-term memory to be used as appropriate. [7]. Figure 2 shows the simplified depiction of the training procedure.

The last step is the output generation in which the new molecules are built, by predicting token by token. Therefore, in each step, a new symbol is predicted depending solely on the previously predicted symbols. Therefore, each next token is indicated, taking into account the remaining structure already generated and, finally, the molecules are syntactically and biochemically validated by the RDKit molecule sanitizer (http://www.rdkit.org).

Predictor

The Predictor is a QSAR model that performs the mapping between the structure of the molecules and its binding affinity against the targets of interest. Two distinct approaches were tested to determine the best architecture and molecular descriptor of the Predictor.

The first approach, used as the baseline, employs the Extended Connectivity Fingerprint (ECFP) as molecular representation. These bit vectors are widely used in the prediction of physicochemical properties, biological activity or toxicity of chemical compounds [24]. The model output is a real number, which is the estimated pIC_{50}. The four developed algorithms are Support Vector Regression (SVR), Random Forest (RF), K-Neighrest Neighbors (KNN) and a deep Fully Connected Neural Network (FCNN). The input data was ECFP6 (vectors with 4096 elements), calculated with the RDkit Morgan Fingerprint algorithm with a three bonds radius [25]. The first three models have been implemented with the Scikit-learning tool (https://scikit-learn.org/). The parameters and hyperparameters applied in these ML-based QSARs are described in Table 1.

Table 1 Optimal hyperparameters for the standard QSAR models

SVR			RF		K-NN	
Kernel	C	Gamma	NumEstimators [1]	MaxFeatures [2]	K	Metric
'poly'	0.125	8	500	'sqrt'	11	Euclidean

The FCNN was implemented with three fully connected hidden layers (with 8000, 4000, and 2000 neurons) activated by Rectified Linear Unit (ReLU) with dropout technique between each fully connected layer to reduce the overfitting. The architecture includes an output layer, which consists of a single neuron that returns the estimation of the biological activity. Figure 3 depicts the model details.

The second approach is depicted in Figure 4, and it uses the SMILES strings as input data without converting them into any other form of descriptors. The model architecture consisted of the embedding layer that converts each token into a vector of 128 elements, two GRU layers (128 units), one dense layer (128 units) and an output with a linear activation function. Since the input data are SMILES, some encoding is required to transform it into a numerical values, starting by the tokenization and padding of each string. Then, a dictionary containing the different tokens is settled. Based on its position on the dictionary, each token of the SMILES is transformed into an integer value. This representation preserves the structural information, character and order, and has a low computational cost given the number of different tokens. Besides, it overcomes the issue of using indirect representations of chemical structures, which thereby adds human bias and in some cases, it can misrepresent the relationship between the compounds and the desired property [26].

Similarly to the Generator, the deep learning abilities of GRU/LSTM cells can be used to learn how to synthesize molecular structures directly from the SMILES representation. They are particularly advantageous as they can work with training data that have inputs of varying lengths. In opposition, traditional QSARs have a

[1] Number of decision trees in the forest.
[2] Maximum number of features considered for splitting a node. In this case, MaxFeatures = sqrt(n_features)

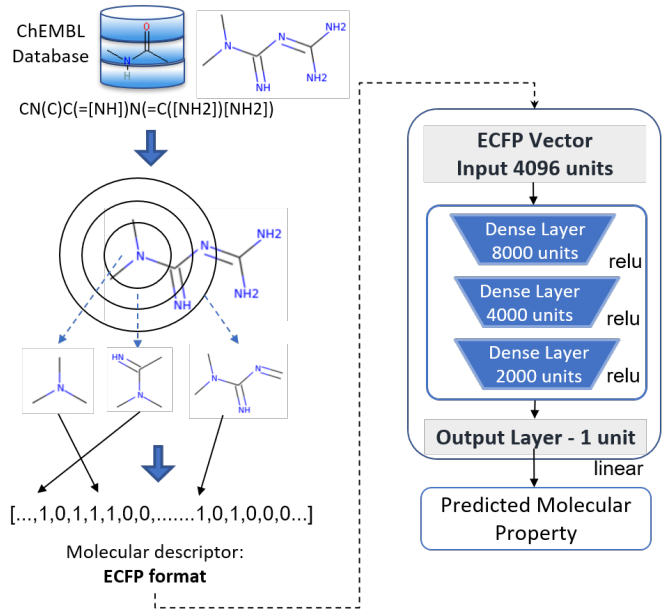


Figure 3 General schema of Predictor with FCNN architecture. ECFP vector is employed as input, calculated with the Morgan Fingerprint algorithm with a three bonds radius.

descriptor matrix with a fixed number of columns and the column position of every descriptor should remain fixed.

After determining the best parameters using a grid-search strategy, the implementation of the regression was performed using five-fold cross-validation to split the data and avoid unwanted overfitting. The data is divided into 85% for training/validation and 15% for testing. Then, the training/validation SMILES are divided into five folds to train an equal number of models. On each fold, data is randomly divided into 85% of the training data 15% for validation. The test set evaluates the robustness of the model in predicting the new molecule binding affinity. The loss function in this regression-like problem is the mean squared error. It helps to measure how close the Predictor learns to get the actual result. Moreover, the early stopping method is employed to allow specifying the arbitrarily large number of training epochs and stop training once the model performance stops improving on a validation subset. An important aspect that should be mentioned is the labels standardization of the data that the QSAR model will predict.

Reinforcement Learning

The RL framework is implemented using the REINFORCE algorithm, and the aim is to teach the Generator the chemical spaces that guarantee the generation of molecules with bespoken properties. This learning process can be seen as an experience-driven change in behaviour. When a specific action brings us benefits, we learn to repeat it. This is the basis of RL, i.e., an agent that learns how to map states into actions through the maximization of the reward, while interacting with the environment [27].

In other words, the goal of this type of problem is accomplished when the best policy/behaviour is progressively achieved, which is reflected in the maximization

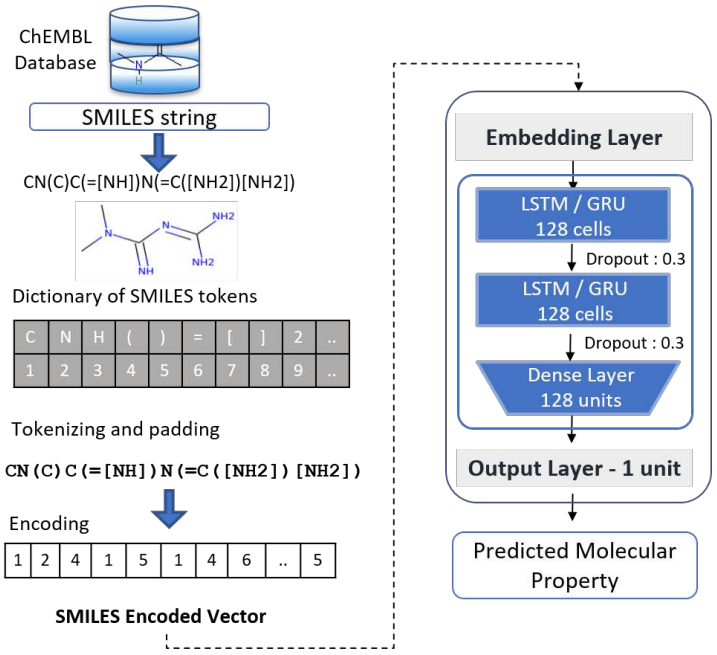


Figure 4 General schema of RNN-based Predictor architecture. SMILES encoding is transformed into an integer vector to be used as input.

of the accumulated discounted reward from various actions [27]. In general, this formulation can be described using the Equation 2. On that account, a lower reward leads to incorrect behaviour/policy, whereas more substantial reward means that the behaviour/policy is evolving in the right direction.

$$R_t = \sum_{k=0}^{T} \gamma^k r_{t+k+1} \tag{2}$$

where R_t is the return, t is the time step, T is the final time step, and γ is a discount factor. It determines how much the future reward worths in the present [27]. Thus, it is a parameter varying from $0 \leq \gamma < 1$ that makes the rewards in near time more desirable than those in the distant future.

The RL is based on the formal framework of the Markov decision problems (MDP). In this formalism, the agent interacts with its exterior, the environment, in order to select the best action depending on the state of the environment. At each step, the agent is in some state $s \in S$, and, it is necessary to choose an available action. In the next step, a numerical reward is assigned to this choice, evaluating the consequences of the previously taken action. In addition, the environment is updated, and the new state is presented to the agent to repeat the process [28]. The process includes the idea of cause and effect, a sense of non-determinism, and the existence of explicit goals.

This formalism can be adapted for the generation of molecules with SMILES strings, and we will specify the parallels between classical formalism and the deep generative field. Thus, the set of actions that the agent can choose corresponds

to all characters and symbols that are used in the construction of valid SMILES. The states through which the agent can pass corresponds to the set of all SMILES strings that can be constructed during the generation process. The policy (π) maps the current state to the distribution of probabilities for choosing the next action [28]. Thus, the main objective of RL is to find an optimal policy, i.e., a policy that selects the actions to maximize the expected reward. The policy is the cornerstone of RL, and, in this case, it corresponds to *the Generator*. The weights of the Generator will be updated based on the gradient of a scalar performance measure ($J(\theta)$ in Equation 3) with respect to the policy parameters. The aim is to maximize this performance objective so that their updates approximate gradient ascent in J:

$$\theta_{t+1} = \theta_t + \alpha \nabla J(\theta_t), \tag{3}$$

where t represents the time step, θ the policy parameters, α the learning rate, and $\nabla J(\theta_t)$ is an estimate, through its expectation, of the gradient of the performance measure with respect to θ_t [27].

Equation 4 represents the REINFORCE update which is achieved by using this sample to instantiate our generic stochastic gradient ascent algorithm (Equation 3).

$$\theta_{t+1} = \theta_t + \alpha \gamma^t R_t \nabla ln \pi(A_t | S_t, \theta_t) \tag{4}$$

Learning Process

The learning process will begin with the sampling of the new compounds. At this point, batches of SMILES are generated, and the analysis of each action is done token by token. In other words, the molecule is created and is assigned with a reward from the Predictor. Then, since each sampled token corresponds to an action, the molecule is "decomposed" into each of its tokens and "reconstructed" again to analyze the probability of each action being taken to calculate the loss function. The cumulative loss is the result of the sum of the probabilities analysis from each taken action in each molecule of the generated batch. On that account, the formulation approximates an MDP since, at each step of the molecule reconstruction, an action is chosen. In response to this action, a discounted format of the initial reward is assigned, and a new state is presented to the agent. This state corresponds to the junction of the partially reconstructed molecule with the selected token from the previous step. The newly created state is used to predict the next token/action.

As previously mentioned, the reward is not attributed to each action but assigned to each molecule. The better the prediction according to the objective, the greater will be the reward. After following this procedure for the batch of molecules, the loss function is calculated and, following the gradient descent method, the weights of the Generator are updated using the following loss function:

$$J(\theta_t) = -\frac{1}{n} \sum_{i=1}^{|S|} \sum_{j=1}^{\text{length}(s_i)} R_i \cdot \gamma^i \cdot \ln\left(p(s_j | s_0 \dots s_{j-1}, \theta)\right), \tag{5}$$

However, since we are employing RL, the exploration/exploitation dilemma has to be considered to guarantee a reliable model. The exploration seeks to collect more information about the environment to select better actions in the future, while the exploitation favours the maximization of immediate rewards. As the agent perceives what gives him more and less reward, it is possible to decide if it is more important to continue learning or to pursue the most promising strategy with the experience gained so far [27]. We consider three strategies to address this issue throughout the work.

First, to make the sampling of new SMILES strings, we use a Softmax activation with a temperature parameter. The temperature allows to precisely control the randomness of predictions by scaling the logits before applying Softmax [21]. If the temperature is reduced, the actions that are already more likely to be chosen are favoured, making the model less susceptible to discovering new actions and, as such, it favours more the exploitation. On the other hand, using higher temperatures, the probabilities of each action tend to get closer, so it is possible to select different actions, and, as such, it is a strategy that gives more priority to the exploration.

The second way to handle this trade-off is through the alternate use of two policies to predict the next token. The innovation of this strategy relies on the fact that it takes into account not only the goal of discovering promising molecules but also the purpose of preserving the diversity of the compounds. In other words, the token selection in the generation of new SMILES strings is determined by two generators: the Generator biased, G_b, that has been updated by the RL approach, and a second fixed Generator, initially trained without RL, Generator unbiased, G_u.

Since the SMILES are being constructed token by token and depending only on the part of the molecule hitherto built, it is possible to intercalate both Generators to predict the next token. The G_u was only trained to generate synthesizable molecules, while the G_b is being trained to maximize a cumulative reward. In this regard, the G_u is associated with exploration, and the G_b is responsible for exploitation. The frequency by which G_b or G_u are used is dynamically determined and depends on the evolution of the reward. Specifically, a random number ϵ between 0 and 1 is generated at each step. If the value is smaller than a threshold λ, the G_u predicts the next token otherwise G_b will be selected. Note that this threshold, λ is dynamically determined as the RL process progresses. As a rule, if the averaged reward is increasing for the two last batches of generated molecules (the Generator is being updated in the right direction), the λ decreases and exploitation is favoured. In contrast, if the reward is not improving as expected, the λ increases in order to the G_u policy be more often employed. The description of this process is depicted in Figure 5.

Lastly, another procedure to preserve an appropriate novelty in the generated compounds is to reflect in the reward value a penalty when the novelty decreases and a bonus when it remains at high levels during the RL process. This adjustment in the reward is performed based on the analysis of the similarity between each created molecule and the collection of the last 30 generated molecules that will act as a continuously updated memory. The similarity between each molecule will be determined using Tanimoto distance, which will analyze the respective ECFP of the molecules to assign a value between 0 (equal molecules) and 1 (totally different).

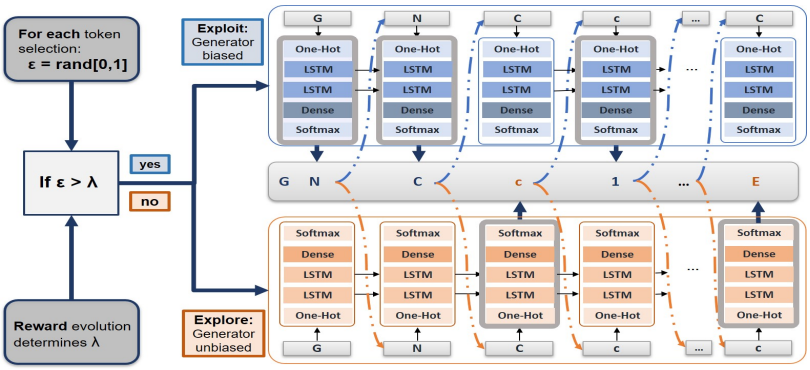


Figure 5 Process of generating SMILES using two Generators alternately.

Then, based on the calculated average diversity, the reward may suffer a penalty of 15% of its value if the reward is less than a threshold κ. Conversely, the reward for a given molecule will receive a bonus of 15% of its value if the average diversity relative to the memory cell is higher than a threshold β. In this regard, generating similar molecules consecutively will have negative repercussions on the reward value. Thereby, performing this correction, the weights will be adjusted in order to avoid the penalization, and the Generator will be able to get out of possible relative minimums. In the opposite situation, the Generator will benefit if it succeeds in adding novelty to the created molecules.

Evaluation Metrics

It is essential for the de novo targeted lead generation to preserve the validity of the compounds and their biological interest [11, 10]. Consequently, the obtained molecules should undergo a rigorous evaluation, namely, regarding their validity, synthetic accessibility, drug-likeness, uniqueness, and diversity.

In this work, a molecule is defined as "valid" if it can pass through the RDKit sanitizer. Also, as we intend to generate molecules with a specific optimized property, we define a metric to assess the percentage of molecules that fulfil this purpose. Therefore, as Liu et al., we determined 6.5 as the threshold for pIC_{50} above which the compound was defined as desirable [15]. Moreover, Tanimoto similarity (T_s) is computed by converting SMILES to ECFP3 to evaluate the diversity of the new compounds thoroughly. It varies from 0 to 1 and the lower the result computed between two molecules, the less similar they will be. Therefore, by defining Tanimoto distance as the inverse of Tanimoto similarity, $I(A, B)$ is the average of the Tanimoto distance of every pair of molecules from the sets A and B [16]:

$$I(A,B) = \frac{1}{|A| \cdot |B|} \sum_{a \in A}^{|A|} \sum_{b \in B}^{|B|} (1 - T_s(a,b)) \tag{6}$$

We will use this concept to define internal and external diversities. The former will be used to assess the diversity within a set of molecules generated by the biased Generator and in that case, set A and B in Equation 6 will be the same. In the

latter case, we will evaluate the diversity between the molecules generated by the biased Generator compared with subsets of data extracted from the dataset used to train the model and also from the unbiased Generator.

Results

Datasets

In order to train the Generator, we extract the dataset from the Zinc database, which is a free database of commercially-available compounds that enables the selection of compounds according to specific properties [29]. For this work, 499,915 SMILES were collected, corresponding to molecules with a partition coefficient (logP) ranging from -2 to 6, and, molecular weight between 200 and 600g/mol. On the other hand, to train the Predictor network, we extract SMILES strings and a value that indicates bioactivity against the targets $A_{2A}R$ (4,872 compounds) and KOR (7,102 compounds) [30]. Both receptors are types of G Protein-Coupled Receptors (GPCRs), and the $A_{2A}R$ is involved in the treatment of conditions such as insomnia, pain, depression, Parkinson's disease, cardiovascular diseases, and inflammatory disorders [19], whereas the KOR mediates pain, mood motor control. The interest in these KOR receptors stems from the fact that they play a crucial role in pain control and drug addiction issues [31].

Experimental Analysis on Initial SMILES Generation

After training the Generator, we sampled 10,000 new SMILES to evaluate its performance with respect to the desirability, validity, diversity, and uniqueness. The results are summarized in Table 2. In this stage, the percentage of desirable molecules it's not the priority since we are only interested in obtaining a model that learns the building rules of molecules through SMILES strings. According to RDKit parsing, nearly 91% of the sampled molecules were chemically valid. Also, the internal and external diversities greater than 0.9, as well as the high rate of unique SMILES in the $10,000$ generated molecules, have demonstrated that the model can add variability and novelty compared to the dataset, which is an essential feature in the generation of lead compounds [7]. Hence, we can infer that the mentioned results are quite promising, i.e., they prove that recurrent architecture is an appropriate method to learn the SMILES grammar. The unbiased Generator is the cornerstone of this work since it will be used as a starting point for all the subsequent experiments and must be reliable.

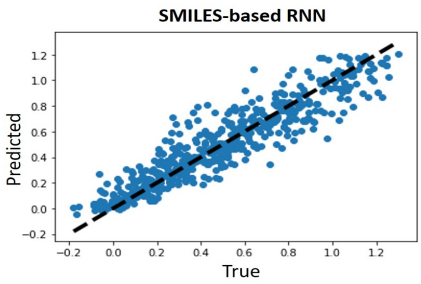
Table 2 Evaluation of Unbiased Generator

% Desirability	% Valid	Internal Diversity	External Diversity	% Uniqueness
86.3	91.1	0.91	0.91	99.9

QSAR Models Performance

Two different strategies were implemented to obtain the QSAR model that established the mapping between the newly generated molecules and its affinity for the target. The aim was to verify that, by employing an apparently more rudimentary and straightforward to obtain descriptor such as SMILES, it was possible to exceed the performance of the models that used the traditional ECFP vectors as a molecular descriptor.

Different algorithms for the QSAR implementation have been evaluated by computing regression-like metrics such as the Mean Squared Error (MSE) and Q^2 [32]. Figure 6 summarizes the obtained results for the $A_{2A}R$ Predictor.



	RNN	SVM	FCNN	RF	KNN
MSE	0.024	0.031	0.042	0.032	0.048
Q^2	0.81	0.79	0.67	0.73	0.63

Figure 6 Performance of the different approaches to implement the QSAR: scatter diagram and analysis of MSE and Q^2

From the analysis of both metrics, it's noticeable that the SMILES-based QSAR provides more reliable information regarding the biological affinity of new compounds. This strategy outperforms the traditional ECFP-based methods (both the standard approaches and the FCNN), which demonstrates that SMILES notation contains valuable embedded information about the compounds for the construction of QSARs and thereby, it will be used in the subsequent experiments.

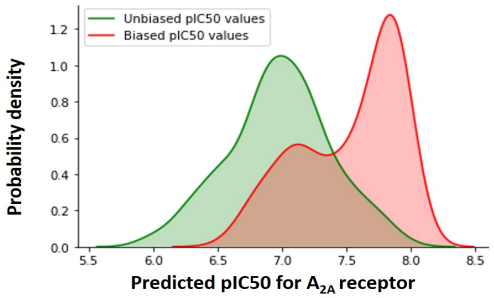
Biased SMILES Generation

Firstly, we started by implementing a grid-search strategy to fix some specific hyperparameters that ensure the proper behaviour of the RL method while minimizing the loss function. Hence, our method was repeated for 85 iterations, using Adam optimizer with 0.001 as the learning rate. Moreover, each batch contained ten molecules, the Softmax temperature was fixed at 0.9, and gradients were clipped to [-3, 3]. Finally, the conversion from the predicted pIC_{50} of the molecule to the assigned reward is performed using the following rule: $R_t = exp(\frac{pIC_{50}}{4} - 1)$

Shifting biological affinities for adenosine A_{2A} and κ - opioid receptors

Throughout this work, the binding affinity of a compound in relation to its target was evaluated using IC_{50}. This parameter stands for the half-maximal inhibitory concentration, and it indicates how much of a substance is needed to inhibit 50% of a given receptor. By using the pIC_{50} which is -log(IC_{50}), higher values of pIC_{50} indicate exponentially more powerful inhibitors.

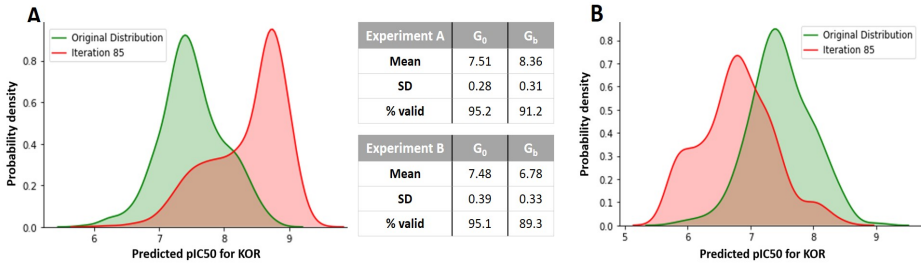
In the first proof-of-concept experiment, we aimed to generate molecules that were likely to inhibit $A_{2A}R$ antagonistically by maximizing the pIC_{50} of the generated molecules. The biasing of the Generator instilled by the application of RL is represented in Figure 7. It compares the probability density of the predicted pIC_{50} for $A_{2A}R$ obtained from the unbiased Generator and after the RL training process. It's noticeable that after retraining the Generator with RL, the likelihood of generating molecules with a higher pIC_{50} increases, and the validity rate remains nearly the same for both distributions. Hence, the newly created molecules have a more significant potential to inhibit the receptor mentioned above.



Generator	G_0	G_b
Mean	6.98	7.53
SD	0.35	0.39
% valid	96.7	94.3

Figure 7 Comparison of pIC_{50} distributions for $A_{2A}R$ before and after applying RL.

Afterwards, the same procedures were applied again for a different receptor to demonstrate the versatility of this framework. Hence, KOR was used as a target for the new compounds, and we perform the training of the corresponding Predictor. However, in this case, in addition to an experiment that aimed to maximize the affinity for this receptor, another one was carried out with the opposite objective of minimizing the pIC_{50}. In the latter, the Generator weights are updated in such a way that it would favour the generation of compounds having a low affinity for the KOR. This type of optimization can be used to avoid the off-target effects since it can reduce the affinity of a potential drug to known competing targets. The result of applying the RL can be seen in Figure 8 and demonstrates that in both cases, the distributions were correctly skewed, and the percentage of chemically valid molecules was kept.



Experiment A	G_0	G_b
Mean	7.51	8.36
SD	0.28	0.31
% valid	95.2	91.2

Experiment B	G_0	G_b
Mean	7.48	6.78
SD	0.39	0.33
% valid	95.1	89.3

Figure 8 Comparison of pIC_{50} distributions for KOR before and after applying RL. **A:** Maximization of biological affinity. **B:** Minimization of biological affinity.

Novelty evaluation - Comparison with other methods

The developed strategy to ensure greater diversity in the generated compounds includes, on the one hand, the manipulation of the reward value, by assigning a penalty to the model when it starts to output similar compounds and benefit it when it manages to add substantial diversity. The search space for the diversity threshold (κ) below which the model is penalized is [0.7, 0.75]. As regards the threshold (β) for valuing the diversity, we define [0.85, 0.9] as the possibilities. On the other hand, the use of two generators alternately, depending on the evolution of the assigned reward, is a straightforward way to conduct the search process

through promising chemical spaces and, at the same time, to maintain the novelty in the resulting compounds. A threshold (λ) from a set of user-defined thresholds is applied to determine the Generator that will be selected to predict the next token. In the case of the averaged reward of the previous batches of molecules is increasing, the smaller λ will be used. Alternatively, if the averaged reward is decreasing, we select the higher λ and, if the reward does not show a defined trend, an intermediate value for λ will be selected. The set of thresholds will be called τ and the verified alternatives were [0,0,0] (τ_1) to work as a baseline, [0.05,0.2,0.1] (τ_2), and [0.15,0.3,0.2] (τ_3). Table 3 outlines the obtained results when the parameters described above are modified. The baseline approach was implemented without both strategies to control the exploratory behaviour of the Generator to perceive their influence on the properties of the obtained molecules. The evaluation metrics were computed after the generation of 10,000 molecules.

Table 3 Results obtained employing different configurations of the parameters that affect the exploratory behaviour of the model.

	Results								
τ	1	2	2	2	2	3	3	3	3
κ	-	0.70	0.70	0.75	0.75	0.70	0.70	0.75	0.75
β	-	0.85	0.90	0.85	0.90	0.85	0.90	0.85	0.90
% Desirable	99.8	95.3	91.1	96.4	93.3	95.1	96.3	95.6	95.8
% Uniqueness	10.3	53.6	50.4	73.6	96.4	93.8	89.2	88.1	83.3
% Validity	97.4	87.4	82.7	79.9	90.4	75.1	74.3	81.4	82.2
Diversity	0.768	0.838	0.827	0.881	0.879	0.880	0.887	0.893	0.891

The first evidence that can be extracted from all experiments is the increase in the percentage of desirable molecules compared to the previously trained Generator without RL (Table 2). Besides, the rate of chemically valid molecules has slightly decreased. However, it is possible to identify specific configurations in which this percentage remained close to 90%, which demonstrates the robustness of our method. The novelty of the newly generated molecules was measured by computing the internal Tanimoto diversity and the percentage of unique molecules. These indicators are intended to ascertain whether the policy falls into a relative minimum of the loss that only reproduces very similar molecules or if it manages to generate a set of lead compounds with interesting diversity. Table 3 shows specific parameter configurations that produce valuable Generators, i.e., models with uniqueness and diversity that corresponds to molecules with the potential to be lead compounds. Moreover, as the selected τ favours more the exploratory behaviour of the model (from τ_1 to τ_3), it's visible an increase in the diversity and percentage of unique molecules. In opposition, the desirability tends to decrease when one chooses to guarantee more significant novelty. As far as the parameters that influence the penalty or bonus of the model are concerned, we see that when the penalty occurs for similarities less than 0.75, the percentage of unique molecules is higher, and the diversity slightly increases when compared with the 0.7 α. This is a foreseen evidence since by penalizing the similarity more vehemently (from 0.75 Tanimoto distance), the Generator will obtain greater diversity in the molecules at the end of the training process. For the bonus threshold, in general, it is not possible to identify an evident trend in all experiments. However, for τ_3, we see that the novelty of molecules is higher when using 0.85 λ.

The adjustment of parameters that ensured the most appropriate compromise between the desired properties occurred when τ_2 was used, and 0.75 and 0.9 were defined for the values of κ and β, respectively. It should be noted that the best policy was not obtained at the end of the training process, but at an earlier stage, when the compromise between diversity and desirability was more favourable. Therefore, it was possible to achieve a high percentage of molecules having a pIC_{50} higher than 6.5, preserving the validity rate. Moreover, our exploratory strategies have the wanted consequence, as Tanimoto diversity significantly increased when compared to the other approaches, as shown in Table 4. The results achieved by REINVENT, ORGANIC and DrugEx were collected from the work of Liu et al. regarding the $A_{2A}R$. The models were executed using their default parameters, and then 10,000 molecules were generated. On that account, Table 4 shows that those works have percentages of chemically valid higher than our approach. In terms of desirability and uniqueness, our work managed to reach comparable results. However, diversity has increased considerably compared to all studies, which demonstrates the efficacy of this solution.

Table 4 Comparison of our approach with other generative methods.

	DrugEx	REINVENT	ORGANIC	Best configuration
τ	-	-	-	2
α	-	-	-	0.7
λ	-	-	-	0.9
% Desirable	98.5	98.2	99.8	93.3
% Unique	99.1	95.8	94.8	96.4
% Validity	99.0	98.8	99.8	90.4
Diversity	0.74	0.75	0.67	0.87

Discussion

In summary, the performed experiments have often shown a trade-off between how skewed the property was and the novelty associated with the generated molecules. It means that, the more the desired property is shifted in the right direction, the more the diversity decreased, with a noticeable rise in the number of repeated molecules. This occurs because, as the weights of the Generator are updated, it is possible to reach specific minimums of the loss function that receive high rewards from the Predictor but reduce the diversity of the compounds. Likewise, if the minimum corresponds to a molecular space that does not respect the rules of construction of molecules, the percentage of invalid molecules will be higher.

However, the trade-off between desirability and diversity is complex to handle since, despite the fact the Generator's main goal is to produce molecules with the correctly biased properties, it is no less critical to keep the variability. This trade-off, which ultimately reveals itself in the molecule's characteristics, has its origin in the exploration/exploitation dilemma. From the analysis of the evolution of all experiments, it's clear that, when the Generator is giving more importance to exploitation, one of two unwanted things can happen. On the one hand, the policy can "fall" into a local minimum, with no interest in the problem context, which continues to exploit the reward function fruitlessly. For instance, the model can produce SMILES strings that are sequences of merely carbon atoms, without any biological significance whatsoever. On the other hand, the continuous updates of

the policy can lead the Generator to a minimum, which, having a significant interest in the problem, substantially reduces the variability of the molecules produced by the Generator. Hence, the main challenge to tackle is how to deal with the exploration/exploitation dilemma. To find the balance between not having sufficient exploitative behaviour for the model to converge in the right direction and having excessive exploitation that causes the model to fall into one of the unwanted situations described above, it's necessary a precise adjustment of the parameters. The baseline approach allowed us to understand the importance of this precise balance. In this case, only the updated Generator was being used to predict the next token during the training process and, as a result, only exploitation was being taken into account. Moreover, the strategy of using thresholds to minimize the repetitive generation of molecules has not been used in this configuration. Therefore, the results demonstrate a remarkable decrease in the novelty of the generated molecules, which confirms the usefulness of these strategies. This finding is corroborated by the best Generator in which the trade-off between desirability and novelty was correctly established.

Figure 9 exemplifies molecules produced by the best biased Generator, namely, it's chemical formula and the respective affinity for $A_{2A}R$.

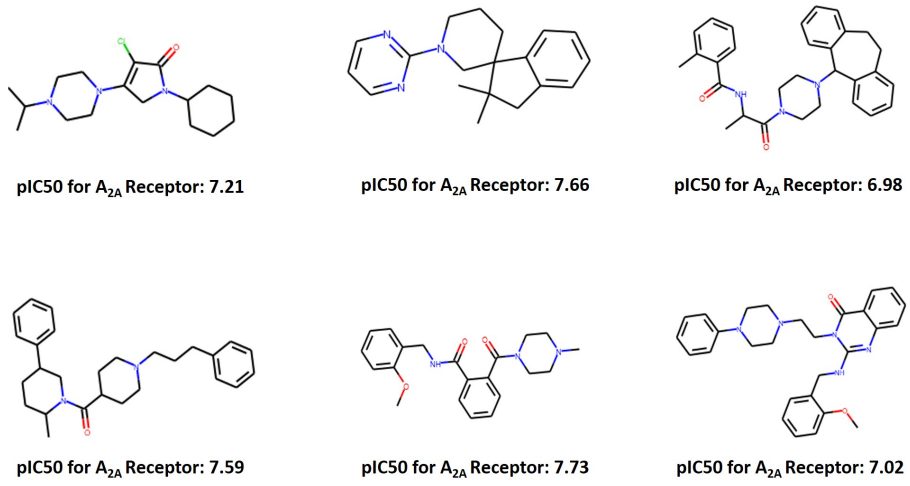


Figure 9 Examples newly generated molecules and respective affinity for the target.

Conclusions and Future Work

In this work, we proposed a molecule generation framework that combines SMILES-based models developed with recurrent architectures and RL for the targeted generation of molecules.

Both the Generator and the Predictor were initially trained with supervised learning. The purpose of the former is to learn the rules for building valid molecules, and for the latter, the goal was to predict the biological affinity of molecules for the target of interest. Both models have demonstrated the ability of this type of architecture to operate with SMILES notation.

We also extended other works that have already used RL to generate molecules with optimized properties. Thus, by integrating the REINFORCE method with exploratory strategies implemented during the training process, it was possible to obtain a set of active molecules for the desired target and to get more diversity than in other comparable methods. In this framework, the Predictor acted as an evaluator of the newly created molecules and contributed to give them a reward based on the goodness of the desired property. On that account, we have shown that the model employing SMILES strings as input data is more robust predicting biological properties than the alternative models based on the standard descriptor (ECFP) such as FCNN, SVR, RF, and KNN.

Throughout this work, the model's versatility has been demonstrated by changing the properties distribution of the molecules in different contexts by maximizing a reward function. In addition, with our exploratory strategies, it was possible to not only bias the generation process towards our goal but also to preserve the novelty and validity of the compounds at quite reasonable levels. Furthermore, the comparison with other methods allowed to confirm the effectiveness of the implemented strategy since it was possible to obtain molecules having more Tanimoto diversity and a similar percentage of desirability. We aimed to demonstrate the importance of the exploration/exploitation dilemma in RL to avoid the fruitless exploitation of the reward function. Nevertheless, although there is room to improve the model, the implemented framework for de novo lead generation is another step forward towards a more prominent application of computational methods in the drug discovery process.

Future Work

After the creation of this framework for the targeted generation of molecules, one of the obvious next steps is the extension of this method to optimize more than one property at the same time. In addition to biological properties, the next version of this work may also consider the physical and chemical features of the generated molecules. This improvement would integrate RL with multi-objective optimization and thereby bring the complexity of the method closer to the demanding standards required in the development of new candidate drugs.

Availability of data and materials

The datasets analysed during the current study are available in the ZINC (http://zinc15.docking.org/) and CHEmbl (https://www.ebi.ac.uk/chembl/) repositories. All code is publicly available in the https://github.com/larngroup/DiverseDRL

Competing interests

The authors declare that they have no competing interests.

Author's contributions

T.P. and M.A. implemented the model, the computational framework and analysed the data; B.R. and J.P.A. designed and directed the project; All authors discussed the results and contributed to the final manuscript.

Acknowledgements

This research has been funded by the Portuguese Research Agency FCT, through D4 - Deep Drug Discovery and Deployment (CENTRO-01-0145-FEDER-029266).

References

1. Rifaioglu, A.S., Atas, H., Martin, M.J., Cetin-Atalay, R., Atalay, V., Doğan, T.: Recent applications of deep learning and machine intelligence on in silico drug discovery: methods, tools and databases. Briefings in Bioinformatics **20**(5), 1878–1912 (2019). doi:10.1093/bib/bby061
2. Segler, M.H.S., Kogej, T., Tyrchan, C., Waller, M.P.: Generating Focused Molecule Libraries for Drug Discovery with Recurrent Neural Networks. ACS Central Science **4**(1), 120–131 (2018). doi:10.1021/acscentsci.7b00512. 1701.01329
3. Ståhl, N., Falkman, G., Karlsson, A., Mathiason, G., Boström, J.: Deep Reinforcement Learning for Multiparameter Optimization in de novo Drug Design. Journal of Chemical Information and Modeling **59**(7), 3166–3176 (2019). doi:10.1021/acs.jcim.9b00325
4. Hessler, G., Baringhaus, K.-H.: Artificial Intelligence in Drug Design. Molecules **23**(10), 2520 (2018). doi:10.3390/molecules23102520
5. Li, Y., Zhang, L., Liu, Z.: Multi-objective de novo drug design with conditional graph generative model. Journal of Cheminformatics **10**(1), 33 (2018). doi:10.1186/s13321-018-0287-6
6. Mauser, H., Stahl, M.: Chemical fragment spaces for de novo design. Journal of Chemical Information and Modeling (2007). doi:10.1021/ci6003652
7. Elton, D.C., Boukouvalas, Z., Fuge, M.D., Chung, P.W.: Deep learning for molecular design—a review of the state of the art. Molecular Systems Design & Engineering **4**(4), 828–849 (2019). doi:10.1039/C9ME00039A. 1903.04388
8. Nicolaou, C.A., Apostolakis, J., Pattichis, C.S.: De novo drug design using multiobjective evolutionary graphs. Journal of Chemical Information and Modeling (2009). doi:10.1021/ci800308h
9. Sanchez-Lengeling, B., Outeiral, C., Guimaraes, G.L., Aspuru-Guzik, A.: Optimizing distributions over molecular space. An Objective-Reinforced Generative Adversarial Network for Inverse-design Chemistry (ORGANIC). ChemRxiv (2017). doi:10.26434/chemrxiv.5309668.v3
10. Olivecrona, M., Blaschke, T., Engkvist, O., Chen, H.: Molecular de-novo design through deep reinforcement learning. Journal of Cheminformatics **9**(1), 48 (2017). doi:10.1186/s13321-017-0235-x
11. Popova, M., Isayev, O., Tropsha, A.: Deep reinforcement learning for de novo drug design. Science Advances **4**(7), 7885 (2018). doi:10.1126/sciadv.aap7885
12. Zhou, Z., Kearnes, S., Li, L., Zare, R.N., Riley, P.: Optimization of Molecules via Deep Reinforcement Learning. Scientific Reports (2019). doi:10.1038/s41598-019-47148-x. 1810.08678
13. Panteleev, J., Gao, H., Jia, L.: Recent applications of machine learning in medicinal chemistry. Bioorganic & Medicinal Chemistry Letters **28**(17), 2807–2815 (2018). doi:10.1016/j.bmcl.2018.06.046
14. Preuer, K., Renz, P., Unterthiner, T., Hochreiter, S., Klambauer, G.: Fréchet chemnet distance: a metric for generative models for molecules in drug discovery. Journal of chemical information and modeling **58**(9), 1736–1741 (2018)
15. Liu, X., Ye, K., van Vlijmen, H.W.T., IJzerman, A.P., van Westen, G.J.P.: 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 (2019). doi:10.1186/s13321-019-0355-6
16. Benhenda, M.: ChemGAN challenge for drug discovery: can AI reproduce natural chemical diversity? (2017). 1708.08227
17. Li, J., Murray, C.W., Waszkowycz, B., Young, S.C.: Targeted molecular diversity in drug discovery: Integration of structure-based design and combinatorial chemistry. Drug Discovery Today **3**(3), 105–112 (1998). doi:10.1016/S1359-6446(97)01138-0
18. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine Learning **8**(3-4), 229–256 (1992). doi:10.1007/BF00992696
19. Chen, J.F., Eltzschig, H.K., Fredholm, B.B.: Adenosine receptors as drug targets-what are the challenges? Nature Reviews Drug Discovery (2013). doi:10.1038/nrd3955
20. Shang, Y., Filizola, M.: Opioid receptors: Structural and mechanistic insights into pharmacology and signaling. European Journal of Pharmacology **763**, 206–213 (2015). doi:10.1016/j.ejphar.2015.05.012
21. Gupta, A., Müller, A.T., Huisman, B.J.H., Fuchs, J.A., Schneider, P., Schneider, G.: Generative Recurrent Networks for De Novo Drug Design. Molecular Informatics **37**(1-2), 1700111 (2018). doi:10.1002/minf.201700111
22. Williams, R.J., Zipser, D.: A learning algorithm for continually running fully recurrent neural networks. Neural Computation **1**(2), 270–280 (1989). doi:10.1162/neco.1989.1.2.270
23. Pascanu, R., Mikolov, T., Bengio, Y.: On the difficulty of training recurrent neural networks. In: International Conference on Machine Learning, pp. 1310–1318 (2013)
24. Gao, K., Nguyen, D.D., Sresht, V., Mathiowetz, A.M., Tu, M., Wei, G.W.: Are 2D fingerprints still valuable for drug discovery? Physical Chemistry Chemical Physics (2020). doi:10.1039/d0cp00305k. 1911.00930
25. Rogers, D., Hahn, M.: Extended-Connectivity Fingerprints. Journal of Chemical Information and Modeling **50**(5), 742–754 (2010). doi:10.1021/ci100050t
26. Chakravarti, S.K., Alla, S.R.M.: Descriptor Free QSAR Modeling Using Deep Learning With Long Short-Term Memory Neural Networks. Frontiers in Artificial Intelligence (2019). doi:10.3389/frai.2019.00017
27. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. IEEE Transactions on Neural Networks **9**(5), 1054–1054 (1998). doi:10.1109/TNN.1998.712192. arXiv:1011.1669v3

28. François-Lavet, V., Henderson, P., Islam, R., Bellemare, M.G., Pineau, J.: An Introduction to Deep Reinforcement Learning. Foundations and Trends® in Machine Learning **11**(3-4), 219–354 (2018). doi:10.1561/2200000071
29. Sterling, T., Irwin, J.J.: ZINC 15 - Ligand Discovery for Everyone. Journal of Chemical Information and Modeling (2015). doi:10.1021/acs.jcim.5b00559
30. Mendez, D., Gaulton, A., Bento, A.P., Chambers, J., De Veij, M., Félix, E., Magariños, M.P., Mosquera, J.F., Mutowo, P., Nowotka, M., Gordillo-Marañón, M., Hunter, F., Junco, L., Mugumbate, G., Rodriguez-Lopez, M., Atkinson, F., Bosc, N., Radoux, C.J., Segura-Cabrera, A., Hersey, A., Leach, A.R.: ChEMBL: Towards direct deposition of bioassay data. Nucleic Acids Research (2019). doi:10.1093/nar/gky1075
31. Beck, T.C., Hapstack, M.A., Beck, K.R., Dix, T.A.: Therapeutic Potential of Kappa Opioid Agonists. Pharmaceuticals **12**(2), 95 (2019). doi:10.3390/ph12020095
32. Gramatica, P., Sangion, A.: A Historical Excursus on the Statistical Validation Parameters for QSAR Models: A Clarification Concerning Metrics and Terminology. Journal of Chemical Information and Modeling **56**(6), 1127–1131 (2016). doi:10.1021/acs.jcim.6b00088

Figures

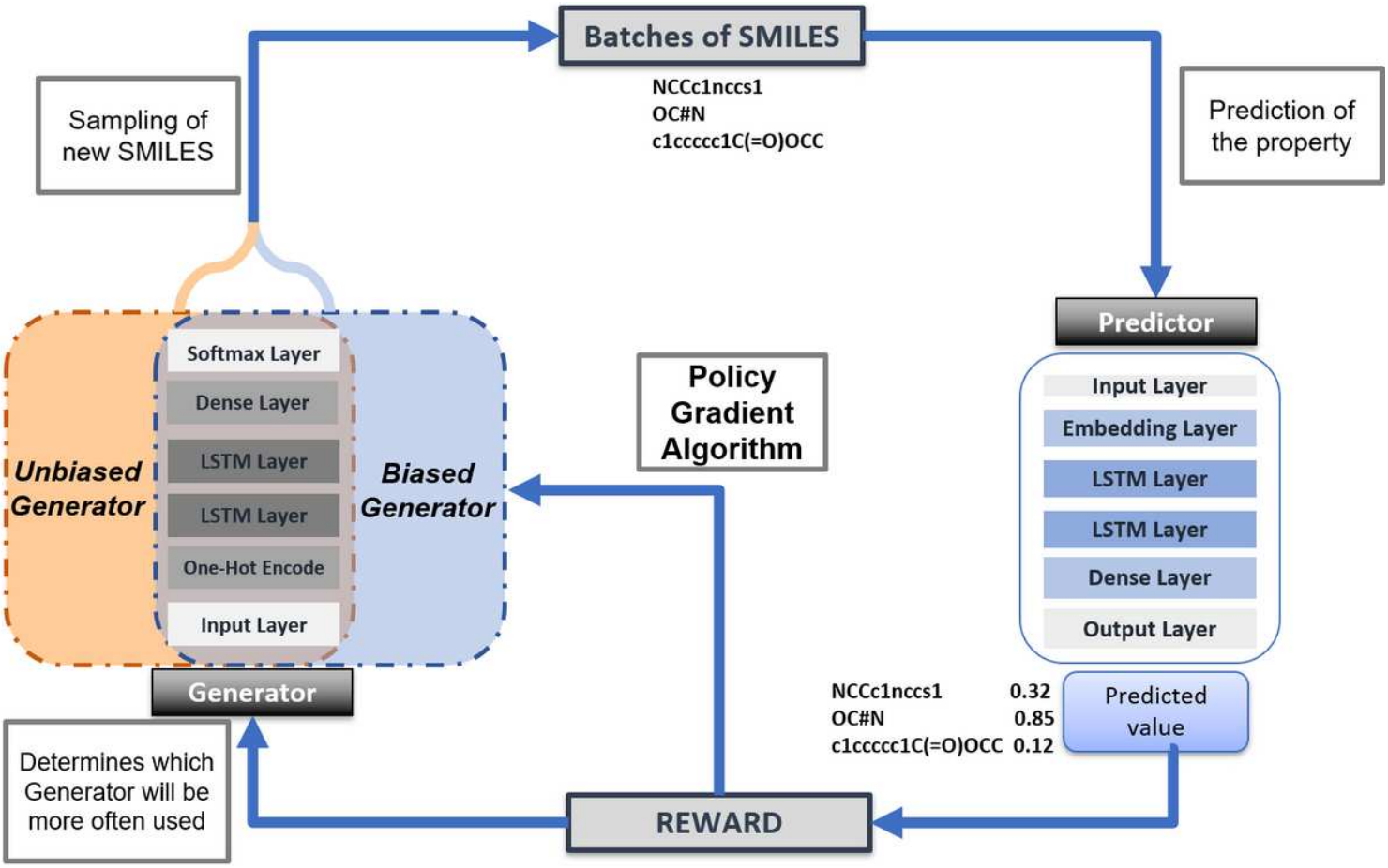


Figure 1

General overview of the framework: Two generators sharing the same architecture and Predictor interconnected by Reinforcement Learning.

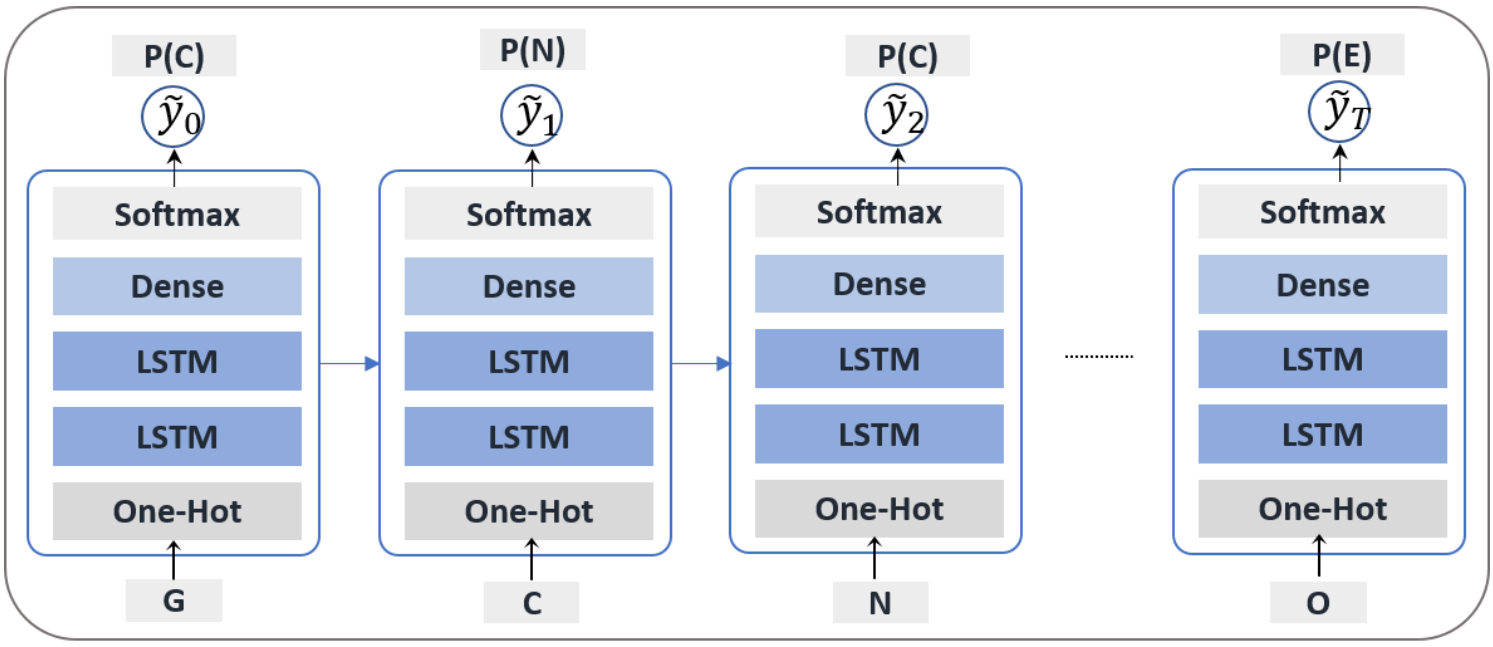


Figure 2

Flowchart for the training procedure for the SMILES string ‘GCNC(C)=OE’. A vectorized token of the molecule is input as xt in a time step t, and the probability of the output to xt+1 as the next token is maximized.

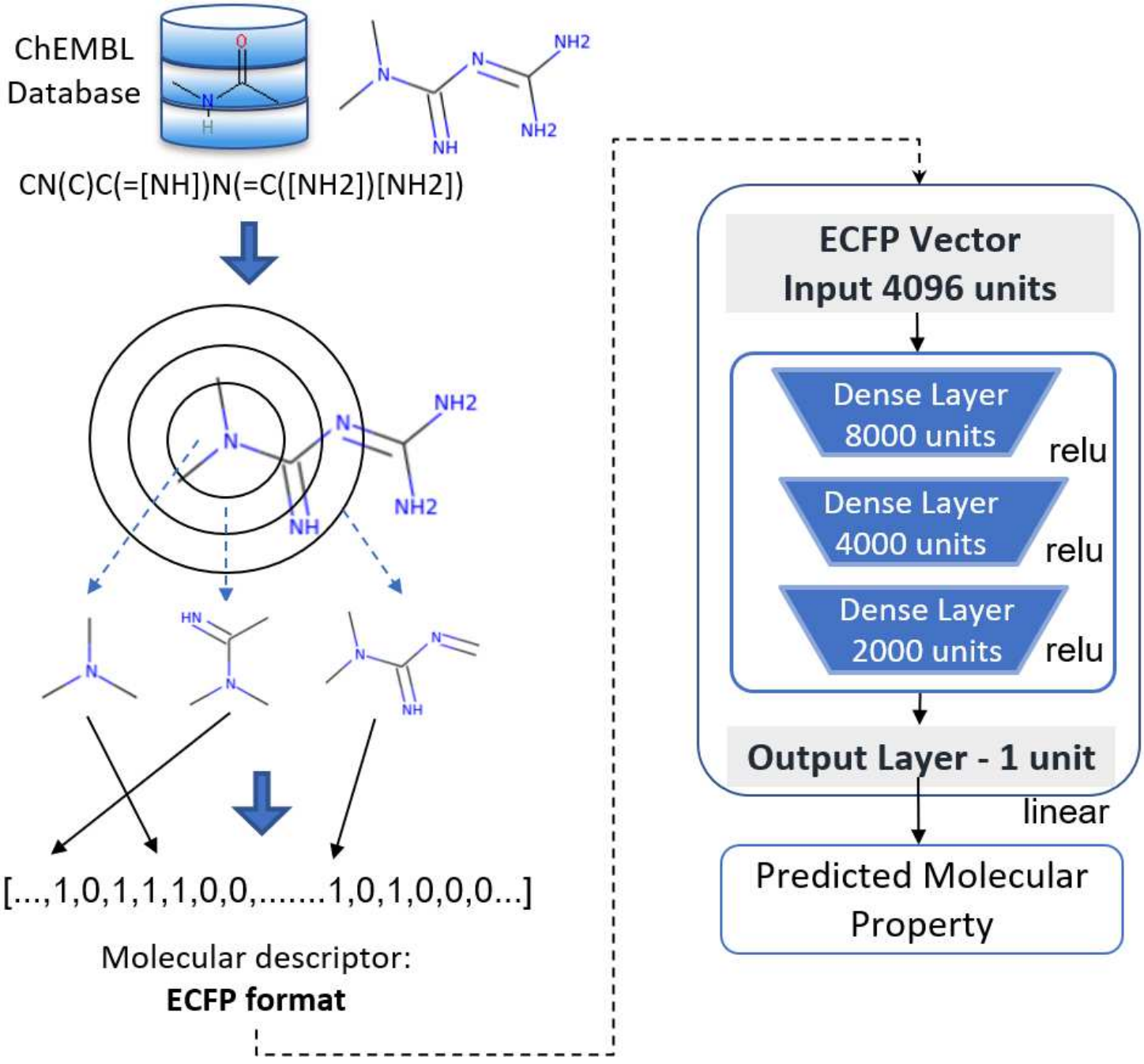


Figure 3

General schema of Predictor with FCNN architecture. ECFP vector is employed as input, calculated with the Morgan Fingerprint algorithm with a three bonds radius.

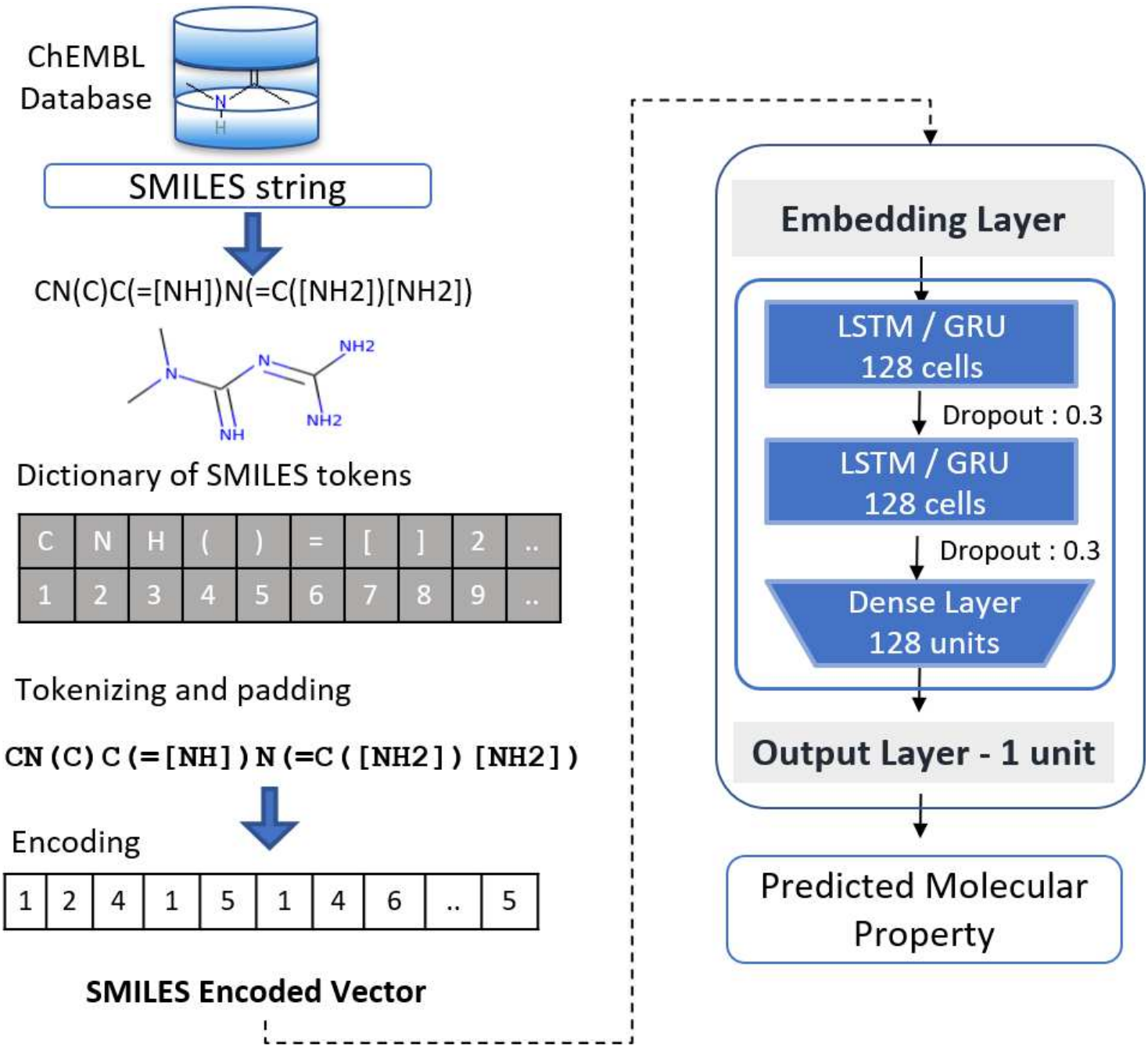


Figure 4

General schema of RNN-based Predictor architecture. SMILES encoding is transformed into an integer vector to be used as input.

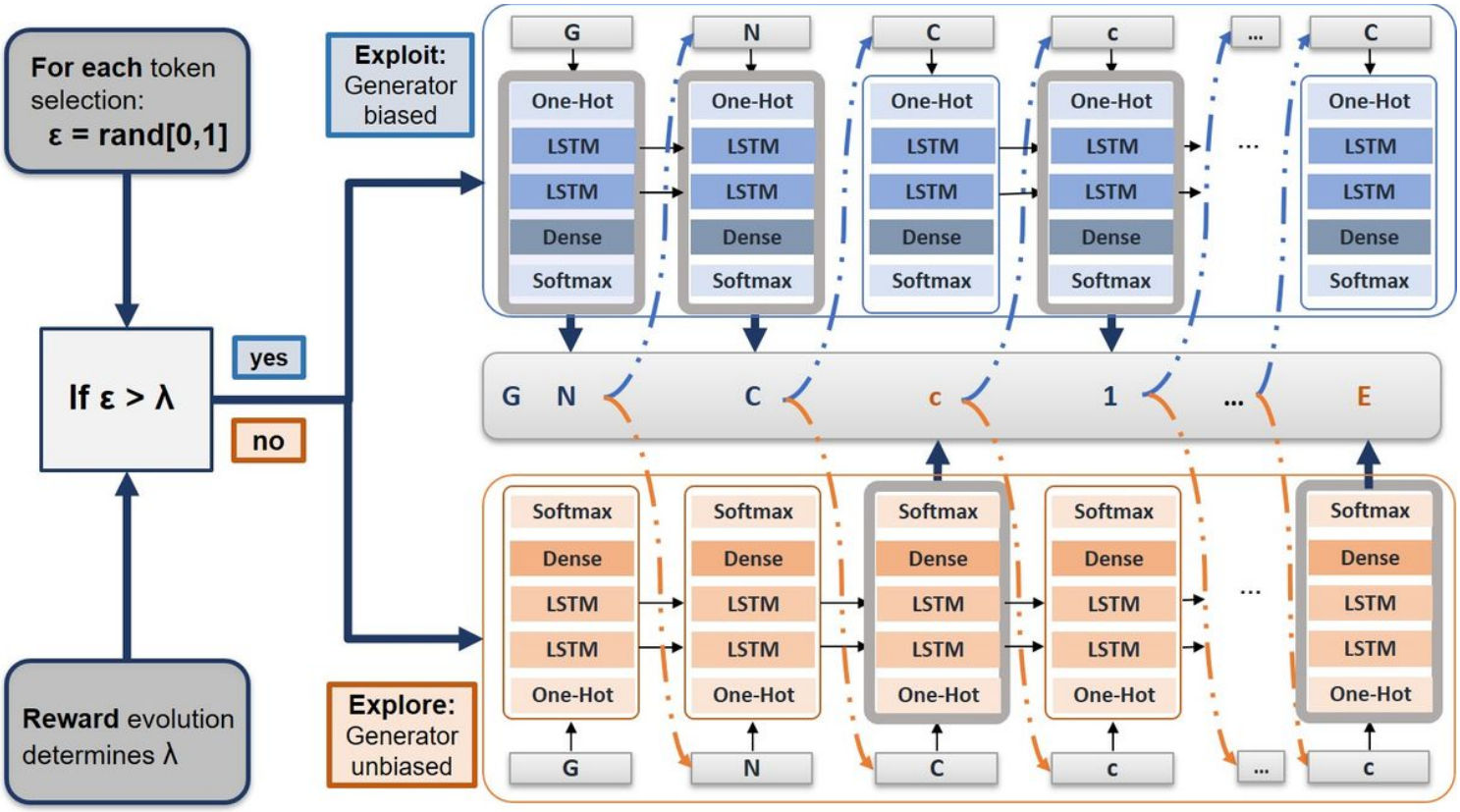
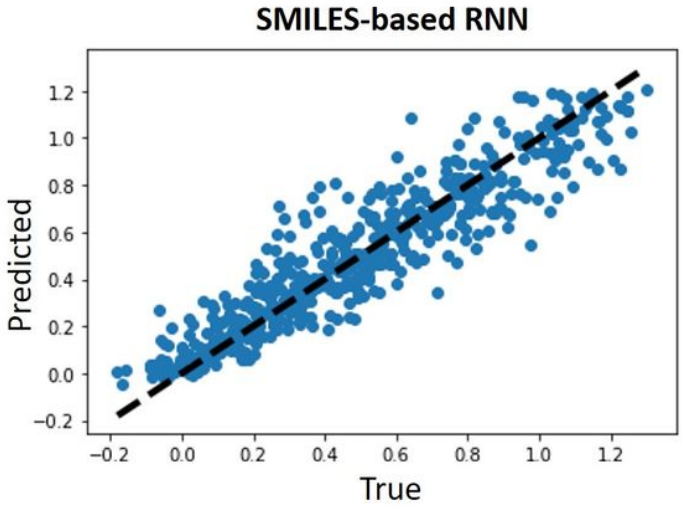


Figure 5

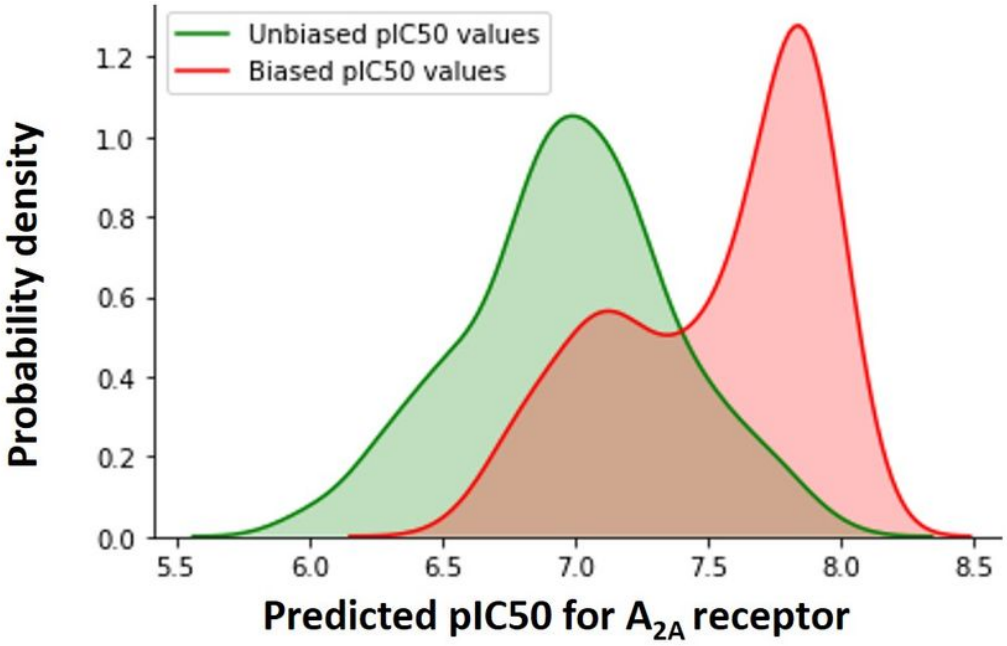
Process of generating SMILES using two Generators alternately.



	RNN	SVM	FCNN	RF	KNN
MSE	0.024	0.031	0.042	0.032	0.048
Q^2	0.81	0.79	0.67	0.73	0.63

Figure 6

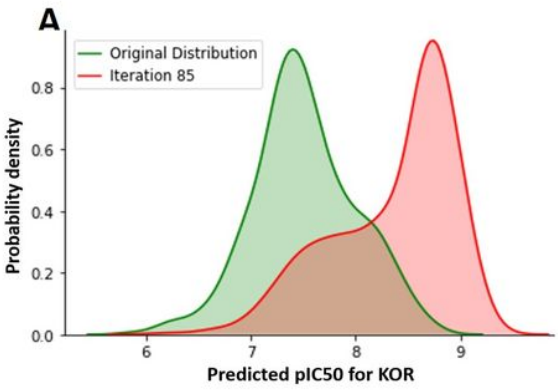
Performance of the different approaches to implement the QSAR: scatter diagram and analysis of MSE and Q2



Generator	G_0	G_b
Mean	6.98	7.53
SD	0.35	0.39
% valid	96.7	94.3

Figure 7

Comparison of pIC50 distributions for A2AR before and after applying RL.



Experiment A	G_0	G_b
Mean	7.51	8.36
SD	0.28	0.31
% valid	95.2	91.2

Experiment B	G_0	G_b
Mean	7.48	6.78
SD	0.39	0.33
% valid	95.1	89.3

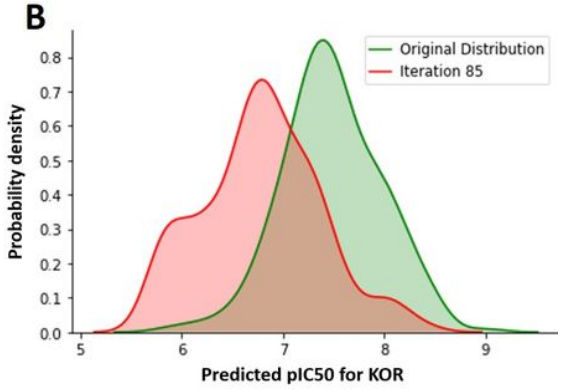
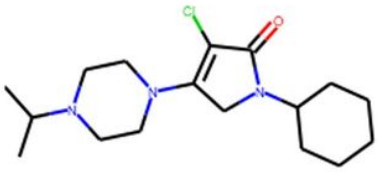
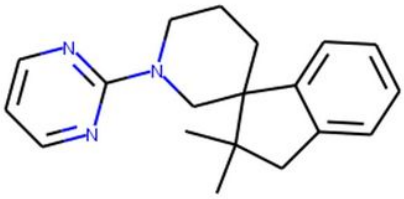


Figure 8

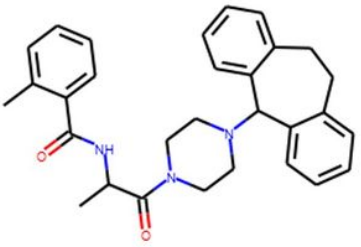
Comparison of pIC50 distributions for KOR before and after applying RL. A: Maximization of biological affinity. B: Minimization of biological affinity.



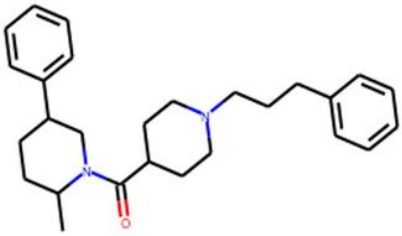
pIC50 for A_{2A} Receptor: 7.21



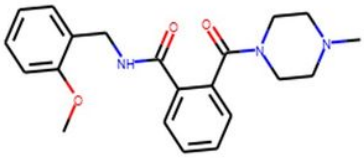
pIC50 for A_{2A} Receptor: 7.66



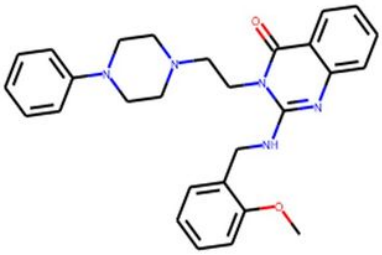
pIC50 for A_{2A} Receptor: 6.98



pIC50 for A_{2A} Receptor: 7.59



pIC50 for A_{2A} Receptor: 7.73



pIC50 for A_{2A} Receptor: 7.02

Figure 9

Examples newly generated molecules and respective affinity for the target.