De Novo Molecular Design by Combining Deep Autoencoder Recurrent Neural Networks with Generative Topographic Mapping

Boris Sattarov [a], Igor I. Baskin [b], Dragos Horvath [a], Gilles Marcou [a], Esben Jannik Bjerrum [c], Alexandre Varnek [a]*

[a] Laboratory of Chemoinformatics, UMR 7177 University of Strasbourg/CNRS, 4 rue B. Pascal, 67000 Strasbourg, France
[b] Faculty of Physics, M.V. Lomonosov Moscow State University, Leninskie Gory, Moscow 19991, Russia
[c] Wildcard Pharmaceutical Consulting, Zeaborg Science Center, Frødings Allé 41, 2860 Søborg, Denmark

1. Abstract.

Here we show that Generative Topographic Mapping (GTM) [1] can be used to explore the latent space of the SMILES-based autoencoders and generate focused molecular libraries of interest. We have built a sequence-to-sequence neural network with Bidirectional Long Short-Term Memory layers and trained it on the SMILES strings from ChEMBL23. Very high reconstruction rates of the test set molecules were achieved (>98%), which are comparable to the ones reported in related publications [2,3]. Using GTM, we have visualized the autoencoder latent space on the two-dimensional topographic map. Targeted map zones can be used for generating novel molecular structures by sampling associated latent space points and decoding them to SMILES. The sampling method based on a genetic algorithm was introduced to optimize compound properties "on the fly". The generated focused molecular libraries were shown to contain original and *a priori* feasible compounds which, pending actual synthesis and testing, showed encouraging behavior in independent "structure-based" affinity estimation procedures (pharmacophore matching, docking).

Keywords: Deep Autoencoder Recurrent Neural Networks, Generative Topographic Mapping, de novo design

1. Introduction.

The development of new chemical compounds with desired properties drives the field of chemoinformatics.[1] The two main approaches to solve this problem are *virtual screening*,[2-5] which is based on the processing of sets of pre-prepared molecules (*e.g.* databases of already synthesized or easily synthesizable compounds, virtual combinatorial libraries) and *de novo* design,[6-8] which involves the generation of new molecules. The *de novo* design can be either *receptor-based*, which requires knowledge of the spatial structure of protein receptors (biological targets) and in which new molecules are assembled within receptor binding pockets, or *ligand-based*, which does not require such knowledge. The latter approach is at the focus of this publication. Methods involving the use of pseudo-receptors can be considered as a combination of both approaches.[9]

The field of ligand-based *de novo* design is currently rather broad. The first-generation research in this area began with the advent of the “inverse QSAR” concept.[10-17] Within the framework of this methodology, the model allowing to predict the property/activity value for a compound is used in the "reverse direction" to restore the structures of compounds possessing, according to this model, a given property/activity value. Inverse QSAR methods are usually based on topological (including fragment) descriptors and include the use of either an exhaustive or a stochastic molecular graph generator.

The second-generation methods in ligand-based *de novo* design are of a broader nature.[8] They are based on the use of general-purpose scoring functions, which can also include similarity measures with respect to active compounds assessed at either topological or 3D levels, in combination with predominantly stochastic search procedures, such as genetic algorithm,

genetic programming, Monte-Carlo techniques, etc. New molecules can be generated in this case by applying formal molecular-graph mutation operations, recombining atoms and fragments (building blocks), as well as combining synthetic reactions.[18]

The third-generation methods have recently been developed under the influence of the "deep learning revolution",[19] which enabled novel opportunities for drug discovery.[20-21] Most of such methods are based on the use of recurrent neural networks (RNNs) for generating SMILES strings, which encode chemical structures.[22] An alternative approach based on directly generating molecular graphs instead of SMILES should also be mentioned.[23-25] Although SMILES were used in several earlier publications to represent molecules in QSAR modeling,[26-29] only since 2017 they are actively used to generate chemical structures.[30-37] In order to perform structure generation, an RNN learns to predict the probability of the next character (or token) in a current SMILES, given the previous ones.[30] Sampling characters (tokens) from predicted probability distributions using a temperature parameter can be used to control the diversity of the molecules being generated.[33] A large number of SMILES strings are needed for training such networks in order to be able to generate correct SMILES strings, because the grammar of the SMILES language is not encoded explicitly in RNNs and needs to be learned. Once trained, such RNN can generate SMILES strings corresponding to new molecules resembling training set examples. A practical procedure for generating focused libraries of chemical compounds active against a chosen target consists of two stages. First, a network is trained on a large amount of data to generate correct SMILES strings. The, the pre-trained network is fine-tuned using chemical structures of compounds known to be active with respect to a certain target, and the resulting network acquires the ability to generate focused libraries of compounds being active

against the same target.[30, 33] New molecules can be generated in this case not only from scratch, but also by growing specified seed fragments.[33]

A more powerful approach to perform *de novo* design of molecules with desired properties is based on the use of Reinforcement Learning (RL) [38] in conjunction with RNNs. RL is a general machine-learning framework that considers "agents" taking "actions" to maximize "rewards". In this case, the "agent" is a RNN being used to generate SMILES strings, the "action" is appending a new character (token) to a partially formed SMILES string, while the "reward" is gained whenever a completely formed SMILES string corresponds to a molecule with desired properties. Unlike the afore-mentioned approach requiring fine-tuning of pre-trained networks, the use of RL offers the additional opportunity to drive inverse QSAR studies in association with quantitative QSAR modes. Olivecrona et al. applied RL to tune a RNN towards generating compounds predicted to be active against a certain target.[34] Popova et al. suggested to integrate two deep neural networks – generative and predictive – for this purpose.[36] The generative network is trained to generate correct SMILES strings, the predictive network is trained to predict desired properties, and then both networks are trained jointly with RL in order to generate molecules with desired properties. In several other publications the same goal was reached by combining RL with the Generative Adversarial Net (GAN) framework. GANs is a recently introduced class of unsupervised machine learning algorithms enabling generating new samples from the distribution represented by the training examples without the need to define and manipulate the corresponding probability distribution explicitly.[39] This makes this approach very attractive for generating high-dimensional or unstructured data, such as high-resolution images in computer vision and molecules in chemoinformatics. This is achieved by combining two neural networks, so-called "generator" and "discriminator" nets, competing with each other

(by trying to “fool” each other) in a zero-sum game. The “generator” net generates new molecules, while the “discriminator” net tries to detect whether it belongs to the same distribution as the molecules from the training set, *e.g.* the distribution of correct SMILES for drug-like and synthesizable compounds. The role of the RL in this case is to tune the “generator” to generate molecules with desired properties. RL was combined with GANs for performing *de novo* design of molecules by Guimaraes et al. in the ORGAN (Objective-Reinforced Generative Adversarial Networks) system[40] and in the RANC (Reinforced Adversarial Neural Computer) system.[35]

Although the use of RL in conjunction with the formation of SMILES using RNN is a universal approach to the design of chemical compounds with desired properties, it has certain drawbacks. It cannot be expected that a very indirect association of the property of compounds with the next added symbol to a gradually growing SMILES can be comparable, in predictive performance and interpretability, with the models obtained in a more traditional way (by directly correlating the property with molecular descriptor values). Moreover, the use of molecular descriptors would make it possible to involve a rich arsenal of chemoinformatics tools and methodologies in the *de novo* design of new compounds and materials.

A fundamental alternative to the use of RL in *de novo* molecular design are the schemes based on the formation of "revertible" molecular descriptors, which, on the one hand, could be used to construct structure-property models, and on the other hand, could be used to reconstruct structures of chemical compounds from their values. Such descriptors can be produced by a special architecture of neural networks - autoencoders.[41-42] An autoencoder encodes an initial object to its higher-level compressed representation (code, i.e. descriptors), while its associated decoder would have the ability to reconstruct the initial object from its code. The code produced

by an autoencoder for a molecule can be considered as a fixed-length vector of “revertible” descriptors, further on referred as “seq2seq” or “latent” descriptors. A simple autoencoder was already used earlier for performing virtual screening based on one-class classification with traditional molecular descriptors.[43] Xu et al. suggested, following the guidelines of the sequence-to-sequence learning model in neural machine translation,[44] to form such “revertible” descriptors by extracting the values of hidden units of RNNs after perception its entire SMILES string.[45] The possibility to use the seq2seq descriptors for generating new molecules in de novo design is, however, not discussed in the cited paper. For this purpose, autoencoders should be generative, *i.e.* should be able to generate new objects. The most popular way to do this is to use “variational autoencoders” (VAE),[46] in which the codes of training objects are fitted to a multidimensional Gaussian probability distribution, which can be sampled to generate new objects. VAEs were proposed by Gómez-Bombarelli et al.[47]for finding new chemical compounds with desired properties using the “data-driven continuous representation”. An adversarial modification of VAE (adversarial autoencoders, AAE),[48] in which an autoencoder is coupled with GANs, was suggested by Blaschke et al.[32] as a general approach to *de novo* design.

Nevertheless, despite the great success in using VAEs and its modifications in various application areas, including generation of new texts, images, sounds, music and other types of objects, one should be aware of the potential danger of the mechanical transfer of this method to the domain of molecular design. The "property-molecule" relation in general case is "one-to-many", because the same property value(s) may be possessed by very dissimilar chemical structures. This results in multimodal statistical distributions of the molecules possessing the desired properties, as it was shown.[49] However, the mathematical theory of VAE, in which any point distribution is approximated by a single Gaussian function, implies the unimodality of this

distribution. This contradiction can potentially lead to a sharp deterioration in the results. In any case, the problem of the multimodality of statistical distributions in *de novo* molecular design should be addressed.

In this article, we put forward a new method of *de novo* molecular design based on combining autoencoder RNNs with Generative Topographic Mapping (GTM). We believe that this method can cope with the problems discussed above for the use of VAEs and RL in molecular design. GTM is a dimensionality reduction and data visualization technique based on variational Bayesian statistics.[50-51] This method has been successfully used in a variety of studies related to the visualization of chemical data, mapping chemical space, analyzing and comparing chemical databases, building classification and regression QSAR models, visualizing activity landscapes, performing inverse QSAR, etc.[49, 52-57] The ability to replace VAE with GTM follows from the similarity of underlying mathematical theory of both methods based on variational Bayesian statistics. Unlike VAE, GTM can successfully be used to approximate multimodal distributions, which is very important for performing *de novo* molecular design. In this case, the application of GTM will allow using the strong features and advantages of this approach in the design of new molecules with the desired properties.

This paper starts with the description of our approach, followed by the analysis and verification of the results obtained in a case study.

2. Data.

The ChEMBL23 database was used both as source of SMILES strings for autoencoder and labeled data for GTM landscapes and framesets. Before training, molecules from ChEMBL23 were standardized as already described[57].

The Adenosine A2a receptor (CHEMBL251) was chosen as the target for this *de novo* design simulation. The activity cutoff for A2a was set to 50 nM. An uncertainty corridor from 50 to 500 nM was allowed, and molecules with K_i inside this range were discarded. Molecules with K_i > 500 nM were treated as inactives.

Further 617 ChEMBL ligand series, each associated to a distinct *H. sapiens* single protein with available K_i or IC_{50} values, were extracted and subject to a similar conversion of activity values into active/inactive labels, for a more general assessment of the predictive power of latent descriptors, beyond the current design problem.

3. Methods.

3.1 Molecule representation.

To be processed by seq2seq models, molecules must be represented as sequences of characters – such as the Simplified Molecular-Input Line-Entry System (SMILES) strings.[22] SMILES were recently used as input for generative autoencoder models, which proved to be able to generate new drug-like hits [15]. Besides, a seq2seq model was applied to SMILES strings to extract data-driven molecular descriptors [8]. Although in principle one canonical SMILES string is enough for an unambiguous description of molecule, in practice, however, different canonization algorithms produce different canonical representations. We used RDKit [16] canonical SMILES in this study.

Since neural networks can process only numeric data, some initial pre-processing of alphanumeric SMILES strings is required. "One-hot" encoding is a technique that converts categorical variables into arrays of binary values ('1' and '0') with strictly a single '1' in each of them (because exactly one value in the array is "hot").[58] Each SMILES string can be represented in this case as a matrix of binary values, in which each row corresponds to an item in the

dictionary of valid symbols, each column corresponds to a character in the SMILES string, and '1' at their intersection denotes the correspondence between them (Figure 1).

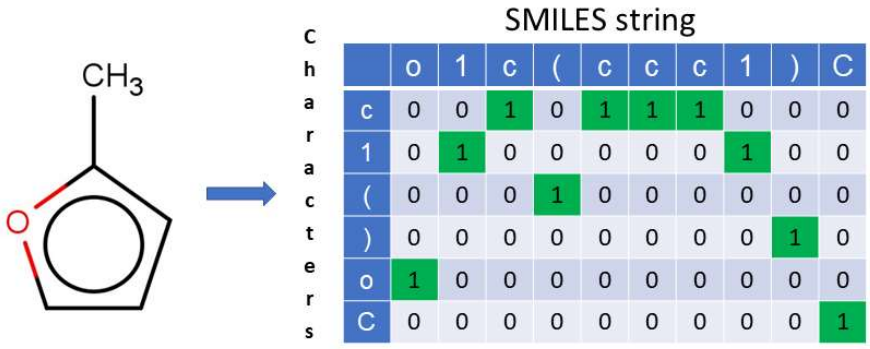


Figure 1. The "one-hot" encoding of a SMILES string.

Atomic symbols (tokens) that contain two characters are encoded as a single special character [17], for example, the chlorine symbol, "Cl", is represented as "X". This significantly simplifies data processing by avoiding the misuse of the "l" character in interpreting and generating SMILES strings.

3.2 Autoencoders.

An autoencoder is a neural network that can learn to produce an internal representation of an input sample in the form of a vector, called *code*, from which the sample can be reconstructed with the smallest possible error (Figure 2).[41-42] A part of the autoencoder called *encoder* computes the function $\mathbf{d} = f_{\varphi}(s)$ for each sample s, where $\mathbf{d}$ is the code, called also *latent vector*, which is the internal representation of the sample s in the *latent space* of the autoencoder. The second part of the autoencoder called decoder tries to reconstruct the sample from its code using the function: $s' = g_{\theta}(\mathbf{d})$. The encoder and the decoder are trained jointly by learning the joint parameter set (φ, θ) to achieve the lowest possible value of the reconstruction loss function

$R(s, s')$, which is the measure of the discrepancy between the sample s and its reconstruction s'.[42] It should be emphasized, however, that not all solutions that can be obtained by minimizing $R(s, s')$ are of practical value. For example, there is a trivial solution for this minimization problem when the network simply learns to copy *s* to *s'* without doing anything else. Obviously, the operation of copying objects could be carried out much easier without any machine learning. Therefore, it is always necessary to avoid obtaining such trivial and useless solutions when training an autoencoder. Therefore, along with the task of minimizing the reconstruction loss function $R(s, s')$, the task of obtaining internal representations **d** that maximally facilitate the solution of additional practically important problems must also be solved.

In most practical applications, autoencoders work with data represented as vectors in a multidimensional space. To avoid the trivial solution, the dimensionality of the autoencoder latent space, which is equal to the number of neurons in the code layer, should be lower than the dimensionality of the data. The autoencoder performs in this case data compression from *s* to **d** and decompression from **d** to *s'*. If the dimensionality of the latent space is equal or higher than some "intrinsic" data dimensionality, the decompression is lossless. Otherwise, the code layer with too small number of neurons becomes the information "bottleneck" of the network, and the loss of information becomes inevitable. The neural network tries in this case to minimize the loss of the information that is important for data reconstruction by performing analysis of data structure, revealing hidden patterns, disentangling hidden factors, and performing noise-signal separation.[42] A new data representation formed in this case in the code layer of the autoencoder can be very useful for further data processing. A similar effect can also be achieved by introducing regularization terms into the minimized function without reducing the actual number

of neurons in the code layer.[59] In the latter case, the effective dimension of the latent space of the autoencoder decreases.

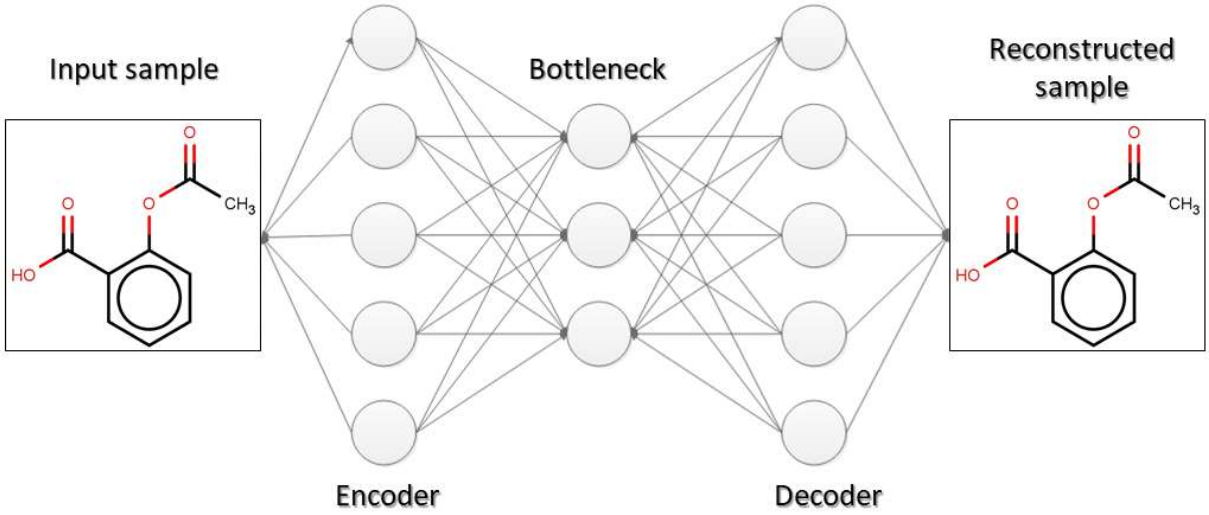


Figure 2. General scheme of the autoencoder neural network architecture

Autoencoders based on neural networks with special architecture can also be used to process complex discrete structures with variable number of elements, such as graphs and sequences of characters. In this case, the formation of vectors in the code layer can be considered as an embedding of such structures into the Euclidean space of fixed dimensionality. In chemoinformatics, the latent vectors **d** are viewed in this case as "revertible" data-driven molecular descriptors, which form a continuous representation of discrete chemical objects, e.g. molecules and chemical reactions.

3.3 Sequence-to-sequence models

The autoencoders used by us in this study belongs to the family of sequence-to-sequence (seq2seq) encoders-decoders, which first encode an input sequence (e.g. a sequence of characters) into a latent vector of fixed size (a code), and then decodes it to an output sequence.[44]

This architecture has proven to be very efficient in natural language processing tasks, such as English-French translation[44] and conversational modeling.[60] If such an encoder-decoder learns to reproduce the input sequences at its output, it is called a seq2seq autoencoder. A general scheme of the seq2seq autoencoders is represented in Figure 3.[61] An input sequence is fed to the encoder based on a recurrent neural network (RNN). The encoder updates its internal state after reading and processing each element of the input sequence, so the last state of the encoder will contain information about the whole input sequence. This state is then passed to the decoder RNN through a fully connected layer, which serves as an information bottleneck and recombines the state into the latent vector (code). This latent vector is used as an initial state for the decoder, which carries out reconstruction of the sequence. At the first timestep, a dummy input character is fed to the decoder. At each next timestep, the expected target sequence character at the previous timestep is fed as an input to the decoder. This technique is widely used in sequence-to-sequence learning and called “teacher forcing”.[62] The outputs of the decoder are passed to the fully connected layer with the “softmax” activation function at every timestep to match the size of the target sequence. After the model converges, the outputs of the dense layer between the encoder and the decoder act as a learnt representation of the sequence, its embedding into the latent space of the autoencoder.

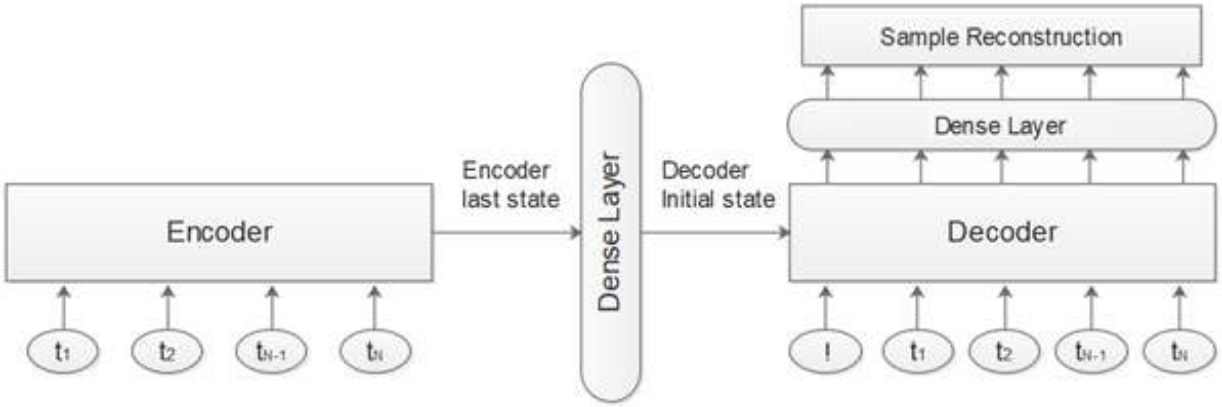


Figure 3. General scheme of a sequence-to-sequence autoencoder

In this study, we used a variation of the SMILES-based seq2seq autoencoder architecture introduced by Xu et al. for seq2seq fingerprint extraction.[45] Molecules were represented as sequences of one-hot encoded SMILES characters and fed to the encoder. Sequences were padded with a special end-symbol "E" at the end to a constant length. This padding enables network training in batch mode and acts as a stop symbol when new sequences are being generated. Besides, the dummy symbol "!" was added to the beginning of each input sequence to enable the teacher forcing by offsetting the input from the target sequence by 1 timestep.

The encoder in our implementation contains two Bidirectional Long Short-Term Memory (LSTM) layers (Figure 4).[63-64] While the ordinary LSTM networks are only able to utilize information from the previous context in the sequence learning, the Bidirectional LSTM[64] can also take advantage of the future context, by processing the sequences in both directions (using two isolated hidden layers connected to the same output). The decoder is composed of two ordinary LSTM layers with a same number of neurons as the encoder. At the end of the sequence, final cell states and hidden states of both layers are concatenated and passed to a dense layer. The outputs of this layer form the latent vector for the input sequence. Subsequently, it is

decoded by four parallel dense layers to form the initial cell and hidden states for each layer LSTM layer of the decoder. The number of nodes in the bottleneck dense layer is lower than the dimensionality of the LSTM states to avoid obtaining trivial solutions. Application of each non-linear activation function was followed by Batch Normalization (BN),[65] which was shown to accelerate training by reducing internal covariance shift.

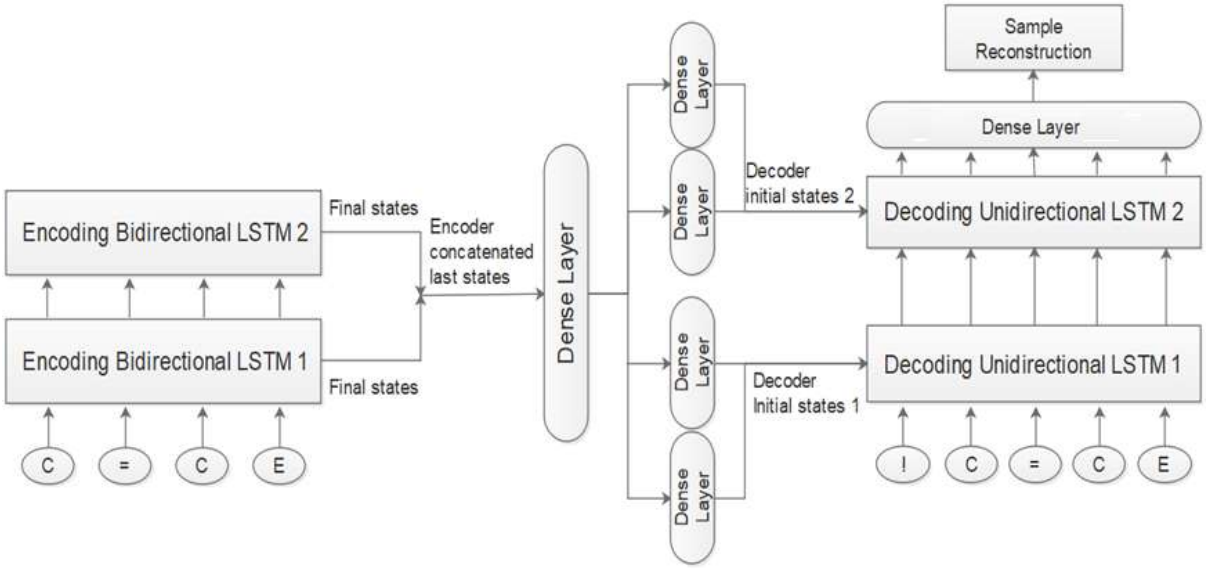


Figure 4. Architecture of the sequence-to-sequence autoencoder used in this study.

3.4 Generative Topographic Mapping

Generative Topographic Mapping of GTM was introduced by C. Bishop as probabilistic counterpart of Self-Organizing Maps (SOM) based on Bayesian learning.[50-51] It has proven to be an effective tool for chemical space analysis,[55-57, 66-70] comparing chemical databases,[52] building classification[54, 56-57, 68] and regression[53, 71] models, building and analyzing structure-activity landscapes,[53, 56-57, 71] solving the "inverse" QSAR problem.[49]

The GTM algorithm operates by embedding a non-linear two-dimensional manifold into a *D*-dimensional descriptor space. First, a squared grid with *K* nodes is created in a two-dimensional GTM latent space, called also a *map*, in which the position of the *k*-th node of the grid is defined by its 2D coordinates $x_k = (l_x, l_y)$. Each node x_k is mapped to its image y_k on the manifold using a non-linear function $y(x; W)$, which is approximated as a linear combination of several predefined Radial Basis Functions (RBF) $\varphi(x)$:

$$y(x; W) = W\varphi(x)$$

where *W* is a matrix of the tunable parameters that define the manifold. Each point y_k serves as a center of a multidimensional isotropic Gaussian function for approximating the distribution of data points (e.g., molecules) around it:

$$p(\mathbf{d}|k) = \left(\frac{\beta}{2\pi}\right)^{D/2} exp\left\{-\frac{\beta}{2}\|\mathbf{d} - y(x_k, W)\|^2\right\},$$

where β is the inverse variance of this distribution. Sampling from it allows generating new data points, e.g. molecules in the descriptor space. GTM implies that all data points have non-zero probability to be generated from the image of any node and hence be mapped to any node on the map, which is "responsible" for it. This probability is called responsibility R_{nk}, k being the index of the node, and n – index of the data point d_n. It can be calculated using the Bayes' theorem:

$$R_{nk} = p(k|\, d_n) = \frac{\exp\left\{\frac{-\beta}{2}\|d_n - y(x_k, W)\|^2\right\}}{\sum_{k'=1}^{K} \exp\left\{\frac{-\beta}{2}\|d_n - y(x_{k'}, W)\|^2\right\}}$$

The compliance of the GTM model with data is measured by L, log(likelihood), calculated as the logarithm of the probability with which the data could be generated using the model, which is a function of two parameters: W and β

$$L(W, \beta) = \sum_{n=1}^{N} \log\left(\frac{1}{K}\sum_{k=1}^{K} \exp\left\{\frac{-\beta}{2}\|d_n - y(x_k, W)\|^2\right\}\right.$$

The optimal values of these parameters can be found by maximizing this function for fixed set of hyperparameters using a special modification of the EM algorithm.[50] The afore-mentioned set of the hyperparameters includes the number of grid nodes, *K*, the number and the half-width of the basis functions, $\varphi(x)$, and a regularization coefficient. The optimal values of these hyperparameters, as well as optimal descriptor sets, can be found by maximizing a "usefulness" score of GTM models, such as an average predictive performance if regression or classification models based on them.[57, 71] The simplest way to deduce such models from GTM models is to use so-called *activity landscapes* as support for regression, and respectively *class landscapes* supporting classification models.[53, 57, 72] Besides visualization, such landscapes can be used for predicting the activity values or class labels for new molecules by projecting their descriptors onto the GTM and taking the local average property as predicted value.[53, 72] Activity landscapes are typically visualized by associating coloring each map point according to a spectrum matching the mapped property range, while the local transparency (alpha channel) is modulated to reflect data density (cumulated responsibility over training set compounds at that point). This allows defining both the model applicability domain and revealing the zones highly enriched with active molecules. Generating new molecules within the latter zones leads to production of focused virtual libraries.

In this study, the GTM technology was used to create maps using the latent vectors **d** of the above-discussed SMILES-based autoencoder as "revertible" molecular descriptors. The strategy used to create "universal" maps[57] based on classical fragment descriptors has been reenacted using the latent vectors instead. The initial descriptor space being fixed, the evolutionary algorithm only searched through the space of GTM hyperparameters in order to create manifolds with optimal propensity to "host" predictive fuzzy class landscapes for 236

randomly selected ChEMBL activities. For a given manifold, the quality of the predictive fuzzy class landscape of every target is estimated by the cross-validated balanced accuracy BA_{CV}. Two thirds of the associated ligand set are projected on the manifold and serve to "color" the landscape by the relative density of actives vs inactives residing at each map point. Then, the kept-out tier is projected *a posteriorly*, to read predicted activity class assignments of its ligands from the landscape. The three tiers are iteratively kept out, then the procedure is repeated four more times, with reshuffling of the ligand set (varying the way in which ligands are being regrouped into the cross-validation tiers). BA_{CV} is estimated by comparing actual and map-read activity labels for kept-out compounds. The top map is the manifolds allowing for a maximal mean $< BA_{CV} >$ over all the 236 considered biological activities. It was then challenged to also host predictive landscapes of the remaining 618-236 structure-activity class series, associated to targets that were in part completely unrelated to the 236 "selection" sets.

3.5 Structure generation.

Given a latent vector, it can be interpreted by the trained decoder to produce a SMILES string for a new molecule. Because the GTM activity landscapes allows us to locate zones enriched with molecules with properties of interest, we can generate new latent vectors for them and, hence, the SMILES strings of novel molecules with desired properties (Figure 5).

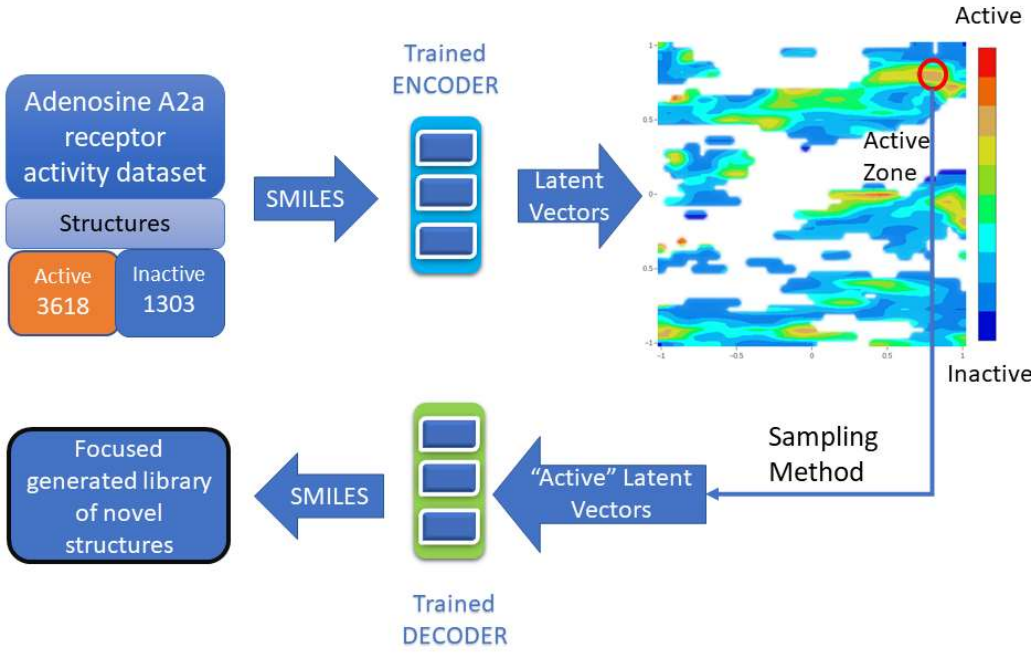


Figure 5. Generation of the focused library of novel active structures of A2a ligands using GTM map and the Encoder/Decoder parts of trained autoencoder

In this study, we have generated two libraries of putative A2a ligands, using two different sampling methods of the latent space – Genetic Algorithm (GA-sampling) and GTM node-based sampling (β-sampling). Several properties of the libraries, such as synthetic accessibility and internal diversity of compounds, were assessed and compared with corresponding properties of the ChEMBL23 database.

Each of these sampling algorithms generate new latent vectors that might correspond to active molecules. These vectors are decoded into the SMILES strings, which are then checked for validity. The percent of valid structures was estimated using the ChemAxon's structure checker with default settings.

For each valid chemical structure, the Synthetic Accessibility score (SA) was estimated according to P. Ertl's method[73] as implemented in RDKit. The SA score indicates how difficult it would be to synthesize a given compound, 1 being very easy and 10 – almost impossible. The

ChemAxon's Structure Checker was then used to discard molecules with reactive or "unwanted" groups. The remaining compounds were collected as potential A2a ligand candidates.

3.5.1 Genetic algorithm-based sampling (GA – sampling)

The GA-sampling explores the latent space by means of the genetic algorithm with the latent-space vectors coded as chromosomes. The fitness function for the GA algorithm was taken to consist of several terms. Its first term verifies the compliance to the applicability domain (AD) of A2a GTM landscape, including two aspects. First, manifold-based log(likelihood), which evaluates whether the latent space point encoded by the chromosome is close to the GTM manifold (at least 5% of frame set compounds must be further away from the manifold, or otherwise the current point is considered an "outlier" and its fitness evaluation stops, returning a default minimal score). Second, it is verified whether the projection of the latent point on the map matches an area populated (with sufficient density) by A2A training set compounds, in order to ensure a meaningful prediction of its probability to be active (default density threshold of the GTM tool being applied). If the above closeness criteria are met, the second term is responsible for checking whether the SMILES strings decoded from the latent vector represent valid molecules, as assessed using RDKit. Should the second check point be passed (otherwise, default minimal fitness returned), the third term evaluates probability P_{A2A} of the sample to be of desired class (A2A active) according to the activity class landscape zone into which it is projected. The last two terms of the fitness score are the synthetic accessibility score *SA* and, respectively, the number *UNW* of "unwanted" groups found by the ChemAxon structure checker in the decoded molecules. By trial and error, the fitness score *Fit* was empirically defined such as to increase with $P_{A2A,}$ all while being penalized with increasing *SA* scores and abruptly decreasing with the number of detected unwanted groups: $Fit = P_{A2A}(1 - SA/100)e^{-UNW}$. Following this

fitness score, the evolutionary simulation was seen to progress towards latent space zones that are in the well-sampled neighborhood of the manifold, project into GTM areas predominantly populated by A2A actives, have a low SA score and are void of unwanted groups.

3.5.2 Random normal node-based sampling (β – sampling)

This is a normal sampling of the initial latent space using coordinates of the GTM nodes as the mean and the GTM parameter 1/β as the variance. Since this method can be applied to any node on the map, only top 8 active nodes of the A2A landscape were selected for sampling. The sampled latent vectors were decoded, and only SMILES strings corresponding to valid molecules were selected.

4. Validation of the generated molecules as A2A ligands by independent *in Silico* approaches

Pending experimental results, the only alternative to check whether generated molecules are potential A2a binders is to computationally confront them to the A2A binding site model. Two independent structure-based methods were employed: structure-based pharmacophore extraction and docking:

4.1 Structure-based pharmacophore filters

Using LigandScout 4.2.1, we have analyzed several PDB structures of the A2A receptor with co-crystallized ligands: 3EML, 3UZC, 3UZA, 5UIG, 5OLH, 5IUA. Then key pharmacophore features were identified for every structure, and corresponding pharmacophore models were created. The pharmacophores were aligned, and a single consensus pharmacophore (containing only the common features) was constructed.

4.2 Candidate Validation by Docking

Two different docking approaches were applied: the in-house program S4MPLE[74-76], a general tool which handles docking simulations as a special case of the wider range of conformational sampling problems it may tackle, *versus* the state-of-art FlexX[77] docking software licensed by BioSolveIT. A subset of randomly selected 1900 ligands of the herein extracted CHEMBL251 (A2a) structure-activity set served as a benchmark in order to verify whether either of these programs is able to effectively discriminate between A2a "actives" and "inactives". If the actives are indeed assigned more favorable docking scores, this is reflected by an increased area-under-curve (AUC) value of the resulting Receiver Operating Characteristic (ROC) curve. Both tools were used in default configurations.

4.2.1 Protein Preparation

LeadIt 2.3.2 (BioSolveIT) tools were used to prepare the 2YDO structure of the A2a receptor from the Protein Data Bank[78]. The standard workflow for protein site curation was followed, all water molecules were removed, amino acid side chain protonation states fixed.

4.2.2 Ligand Preparation

The 1900 ChEMBL251 ligands selected for docking were subjected to an automated conversion to protonated initial 3D structures, by means of an in-house tool developed on the basis of the ChemAxon API. The tool relies on the tautomer[79] and pKa plugins[80] to generate the most probable microspecies of the expected main tautomer. Explicit hydrogen atoms were added, and a single conformer was then generated using the conformer plugin. Eventually, the charge plugin[81] was used to assign Gasteiger charges to all atoms in this structure (to be used by S4MPLE, but not by FlexX). Last, the tool detected flexible rings and proposed the single bonds to be formally "broken" in order to enable intra-cyclic torsional axes to be driven by S4MPLE

(macrocyclic sampling was, by contrast, disabled in FlexX, as its default setting prones). Next, antechamber[82] and other utilities, as called by GAFF pilot scripts, were used to assign GAFF[83] ligand types and to automatically set associated FF parameters to the internal coordinates found in the ligands. The FF types and Gasteiger charges were added as data S4MPLE-readable fields to the MDL sd file used to store the proposed initial conformer of the ligand. The same sd files were used by docking with FlexX, which however ignores the S4MPLE-specific parameters and uses its internal parameters for docking.

4.2.3 Active Site Specifications

The docking program S4MPLE uses a predefined cutoff of 12 Å for non-bonded interactions. Therefore, a selection of relevant residues that have at least one atom at less than 10 Å from any of the co-crystallized 2YOD ligand was used as site model for S4MPLE calculations. Protein atoms were explicitly declared as fixed (by default, all degrees of freedom are considered in S4MPLE). Moreover, S4MPLE requires the user-specified input of "hot spots", which are used for random prepositioning of the ligand into the active site. These were site atoms seen to make contacts to the cocrystallized PDB ligand, including the aromatic ring of Phe168 and His250, the hydrophobic pocket defined by Leu249, plus the key hydrogen bond donor/acceptor Asn253. Their choice has no impact on the docking energy function (they are not used to tether the ligand).

For FlexX, the active site was automatically defined by the subset of protein residues interacting with the adenosine ligand present in 2YDO according to the procedure implemented in LeadIt.

4.2.4 Docking Protocols.

Docking and scoring with FlexX was driven by the batch file dock_list taken from the program distribution. The only non-default option it includes is a user-imposed pose optimization of 500 steps. Docking with S4MPLE followed the published[84] standard protocol. Therefore, the docking score ΔE for the current ligand can be directly estimated as the energy difference between the lowest energy levels of bound and respectively free ligand geometries, $min\langle E_i\rangle^{ligand@site} - min\langle E_i\rangle^{ligand}$. Based on benchmarking results, S4MPLE has been used to dock a library of 394 *de novo* candidates that successfully passed the pharmacophore matching test. Their docking scores were compared to the scores of known A2a actives and inactives from the benchmark set, respectively.

5. Results and discussion.

5.1 Autoencoder architectures and reconstruction errors.

Different architectures of the neural sequence-to-sequence autoencoders for SMILES string were developed and tested, see Table 1. Reconstruction accuracies were accessed and compared. We have found out that Bidirectional LSTM encoders demonstrate the highest SMILES reconstruction accuracies (ACC) and the optimal architectures should contain 2 hidden layers in both encoder and decoder. Besides, it turned out that Batch Normalization is crucial for reaching the highest reconstruction rates in a reasonable time of training (< 10 hours on Nvidia GTX1080) for this type of autoencoders. High reconstruction accuracies achieved on the test set imply that the autoencoder is well generalized and can effectively be used for representing most of the SMILES strings of ChEMBL23 molecules.

Table 1. SMILES reconstruction accuracy (ACC) and percentage of valid generated SMILES strings (VAL) provided by different autoencoder architectures

LSTM type	Enc\Dec layers	Units in encoder	Bottleneck Layer	Units in decoder	ACC, %	VAL, %
Unidirectional	2	256+256[1]	128	256+256[1]	95.80	96.5
Bidirectional	2	(128+128) + (128+128)[2]	256	256+256[1]	99.71	99.65

[1]two layers with 256 units in each of them; [2]two layers each consisting of two sublayers with 128 units in each of them (see description in text)

Latent descriptors provided by the trained autoencoder represent a relevant chemical space, in which it is possible to simulate "walks" from one point (chemical structure) to another by visiting interpolated latent space points which translate to hybrid structures "mixing" features of departure and arrival compound. We simulated a "walk" in the latent vector space, from penicillin V to ibuprofen, see Figure 6. Latent space points along the path were sampled and decoded into corresponding SMILES strings. Out of the 22 sampled latent points, 19 corresponded to valid molecules. This simulation is a chemical version of the popular "morphing" of one portrait into another in image processing, so the intermediate molecules can be considered in this sense as "imorph frames" between penicillin and ibuprofen.

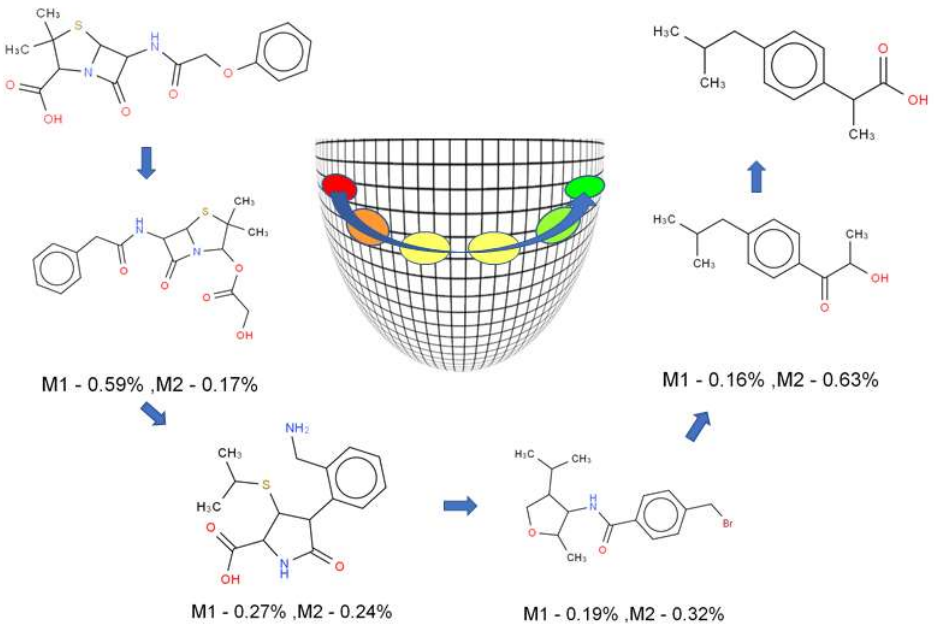


Figure 6. Interpolation between the penicillin V and the ibuprofen molecules in the latent space. When smallest distance path from one molecule to another is sampled, new molecules between them can be discovered. M1 – Tanimoto similarity of the given decoded structure to penicillin, M2 – to ibuprofen

We have tracked similarity [Tanimoto coefficient over all pairs of molecules using ECFC4-like Morgan circular fingerprint hashed to 1024-bit vectors as implemented in RDKit] of each structure picked along the path from penicillin V to the ibuprofen with respect to both these molecules. Figure 7 indicates that when moving from the penicillin to the ibuprofen the similarity with respect to the penicillin continuously decreases, while the similarity to the ibuprofen decreases. This test proves that in the latent space similar molecules are indeed located close to each other.

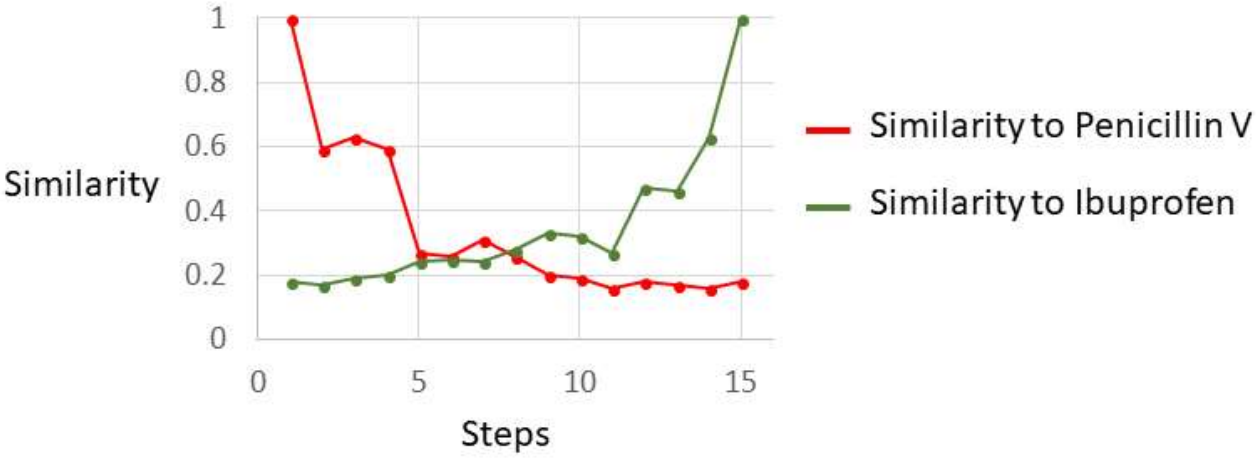


Figure 7. The similarity of the two terminal compounds to molecules generated along the path in the latent space from penicillin V to ibuprofen.

We have also checked whether minor random changes of latent vectors result in minor structural changes in molecular structures. Eight chemical structures generated by perturbing the latent vector representation of ibuprofen using random values sampled from the normal distribution with very small standard deviation are shown in Figure 8, along with the values of the Tanimoto coefficients for them with respect to ibuprofen. One can see that all eight molecules are indeed rather similar to ibuprofen, with the average value of the Tanimoto coefficient being 0.62.

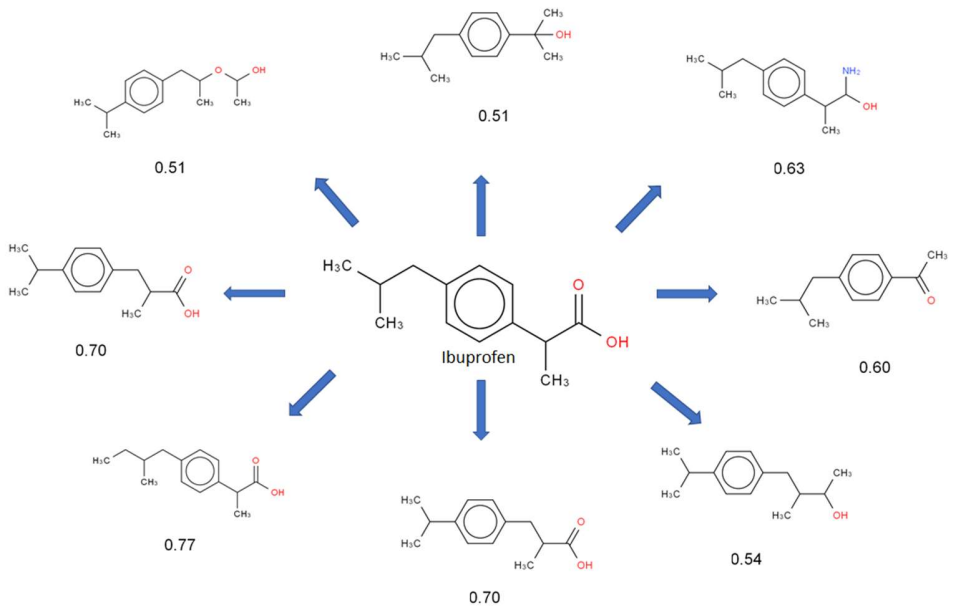


Figure 8. Molecules generated by perturbing the latent vector representation of ibuprofen using random values sampled from the normal distribution with very small standard deviation. A number below each structure indicates the values of the Tanimoto coefficients respect to ibuprofen.

The results of the above tests indicate a good neighborhood behavior of the space, determined by latent vectors, and the possibility of using it as a continuous representation of molecules both to solve the problem of optimizing the target property and to generate focused libraries of compounds with the desired properties.

5.2 GTM built on latent descriptors.

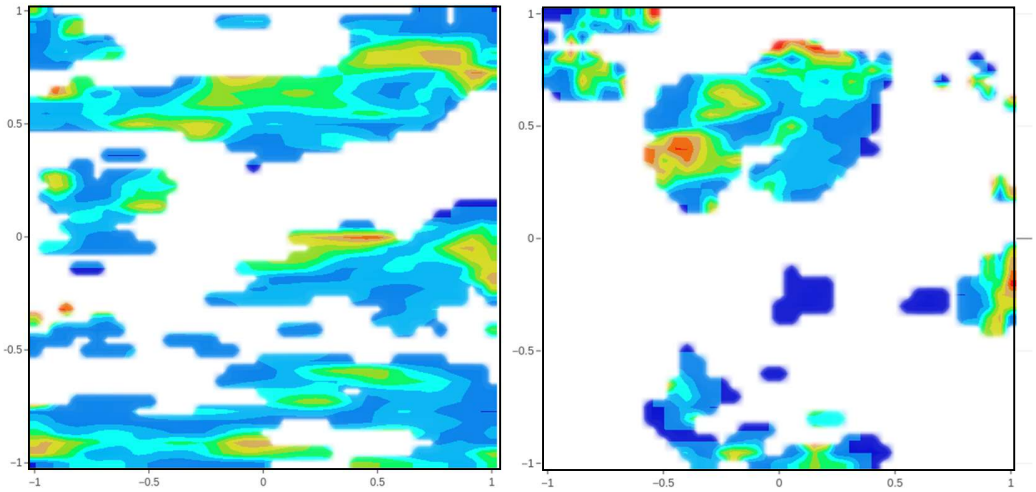


Figure 9. GTM class landscapes based on the latent vectors (left) compared to one example of a "universal" map based on ISIDA fragment descriptors (right). The blue zones on the maps indicate mostly inactive compounds, while the red zone – mostly active.

Using the GTM methodology, we have built "universal maps" for the same sets of ChEMBL compounds using both (a) the latent vectors produced by the autoencoder with the chosen architecture, and (b) standard ISIDA fragment descriptors[85] successfully used in most previous applications concerning the use of GTM for chemical data analysis. Then the same set of A2a binders CHEMBL251 was used to color the maps to produce the class activity landscapes. The activity landscape for the autoencoder latent vectors is depicted at the left side of Figure 10, whereas one of the analogous activity landscapes built using ISIDA fragment descriptors is given at the right side of Figure 9. Both descriptor types appear to produce landscapes with comparable predictive performance – the cross-validated balanced accuracy, BA_{CV}, values are 0.72 for the latent vector map *versus* 0.68-0.76, depending on the specific ISIDA "universal" map based on fragment descriptors. Note that the target of interest, A2A (CHEMBL251), is an "external" set, which was never used for map selection (neither for latent-vector, nor for ISIDA descriptor maps). The BA_{CV} values for the remaining 617 ChEMBL

activity sets are also distributed very similarly, as indicated in Figure 10. This indicates that the latent vectors are competitive with respect to state-of-the-art molecular descriptors for building structure-activity models.

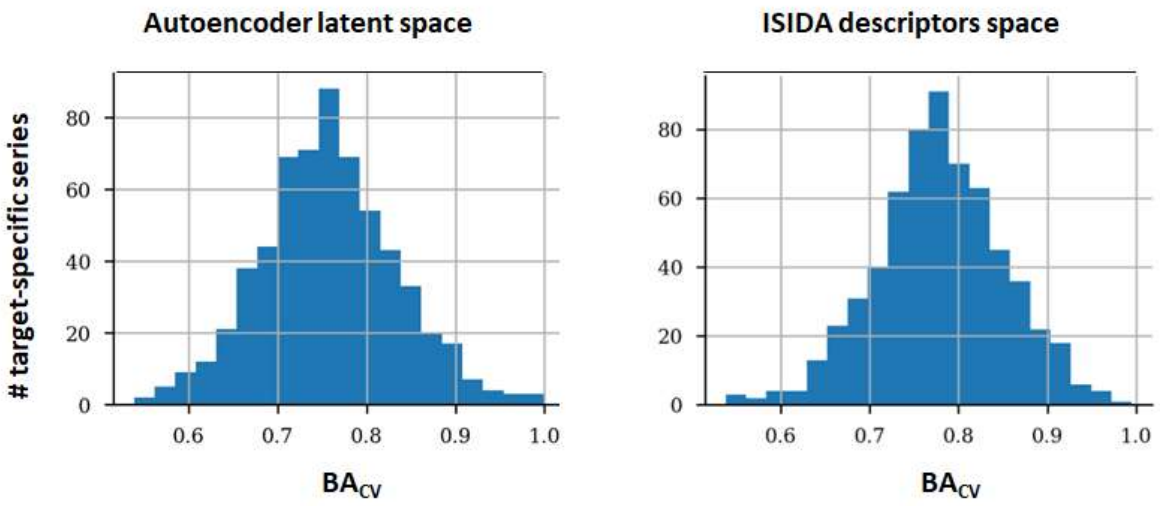


Figure 10. Comparison of the BA_{CV} (Cross-Validated Balanced Accuracy) histograms for classification models built on 617 ChEMBL target series using latent vector-based GTM (on the left) and the "universal" map 1 based on ISIDA fragment descriptors (on the right).

As can be seen from the comparison of the two landscapes shown in Figure 9, when using latent vectors instead of fragment descriptors, a much larger part of the map turns out to be colored, which makes the map much more informative. Perhaps this is because the descriptors learnt from data are generally better suited for describing data distribution, because some of the details of this distribution are implicitly embedded in the descriptors themselves.

5.3 Generation of the focused libraries of A2a-receptor binders

The process of generating focused schematic libraries described above is shown in Figure 5. Projection of the generated compounds back on the A2A landscape based on the GTM map built using autoencoder latent vectors indicates that the generated structures expectedly fall into the selected active zone on the map.

Assessment of the properties of the generated focused libraries for the A2a receptor has shown that the compounds generated using the genetic algorithm (GA-sampling) are significantly easier accessible synthetically than the β–sampled ones, the former being almost equal to the ChEMBL23 baseline (see Table 2), which can be explained by the fact that the fitness function of genetic algorithm explicitly favored low SA scores (i.e. more synthetically accessible).

Besides synthetic accessibility, it is also important to assess the diversity of the generated focused libraries, because for discovering new drugs by screening the focused libraries of generated compounds it is necessary to be able to obtain new chemotypes with possible new modes of action. Since molecules were generated to cover A2a active zones of the chemical space, the increase in the average pairwise similarity in comparison with ChEMBL baseline is completely expected. However, as it follows from Table 2, the library generated by β-sampling is more diverse than the other one produced by genetic algorithm and almost as diverse as the compounds extracted randomly from ChEMBL23. Thus, as follows from Table 2, the synthetic availability and diversity of the generated focused libraries is comparable to the similar characteristics of compounds from ChEMBL, which indicates the possibility of using them instead of the sets of known compounds in the process of virtual screening.

Table 1. Comparison of synthetic accessibility and diversity of the compounds in generated libraries vs compounds from ChEMBL23 dataset. Average Diversity refers to the Soergel

distance (1-Tanimoto Coefficient) over all pairs of molecules using ECFC4-like Morgan circular fingerprint hashed to 1024-bit vectors as implemented in RDKit.

	GA-sampling	β-sampling	ChEMBL23 baseline
Average SA score	2.94	4.59	2.99
Average Diversity	0.78	0.85	0.88

5.4 Validation of the generated compounds using structure-based methods

Pending real synthesis and experimental validation, we have applied alternative chemoinformatics approaches that do not rely on machine-learned models (pharmacophore screening and docking) to validate that the generated compounds could be binders to the adenosine A2A receptor.

5.4.1 Pharmacophore screening

The structure-based pharmacophore model built by us for A2A binders include 4 features that are crucial for the activity and are present in all co-crystallized A2A ligands (see Figure 11 for a protein-ligand complex overlapped with the pharmacophore features):

1. hydrogen-bond donor interacting with ASN253A;
2. acceptor interacting with ASN253A;
3. aromatic ring interacting with PHE168A;
4. hydrophobic group interacting with the hydrophobic part of the binding pocket (LEU85A, LEU249A, MET177A, VAL84A).

We have observed that compounds from the focused libraries are approximately three times more likely to comply with this pharmacophore model, as compared to the compounds selected randomly from the ChEMBL database (see Table 3). This suggests that developed workflow

could produce focused libraries enriched with compounds potentially possessing the desired biological activity.

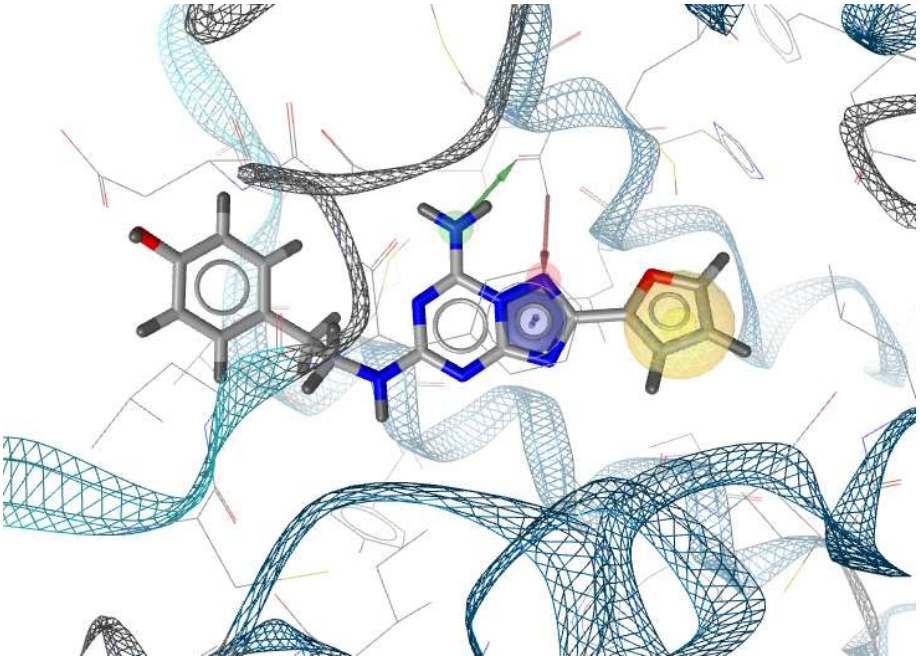


Figure 11. The binding site of the A2a receptor co-crystallized with ligand (3EML PDB structure) overlapped with the developed pharmacophore filter.

Table 3. Percentage of molecules complying with the developed pharmacophore

Source of molecules	β-sampling	GA–sampling	CheMBL23,
Pharmacophore hit rate, in %	15.13	16.05	5.37

5.4.2 Docking of ligands to protein

The Receiver Operating Characteristic (ROC) curves for our docking experiments for the benchmark subset of 1900 ChEMBL A2a binders are presented in Figure 12. As can be seen, both FlexX and S4MPLE are better-than-random selectors of active A2a ligands. However, their

performance is not high. According to the Area Under the Curve (AUC), S4MPLE (AUC=0.64) slightly outperforms FlexX (AUC=0.62). We have also found that S4MPLE reproduces correct correctly binding modes of ligands in the crystal structures 2YDO, 4EIY, 3EML, 5IU4 and 2YDV, but fails with the highly flexible ligand in 3QAK, for which the standard 500 evolutionary generations are clearly not sufficient to discover its native pose.

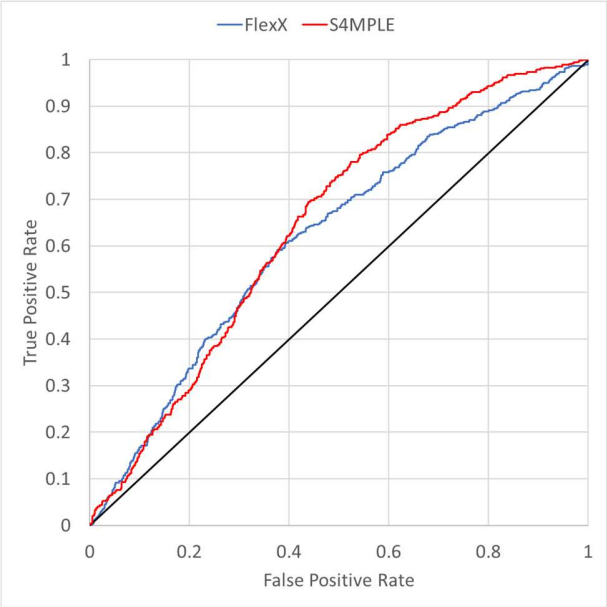


Figure 1: ROC curves for docking experiments with the benchmark set with 1900 compounds with annotated A2a activity selected at random from ChEMBL

Since protein-ligand docking is a time-consuming process, only a subset of 394 compounds produced by the GA-sampling method and passed the pharmacophore matching tests were docked using the S4MPLE program. The GA-sampling was chosen because of the higher pharmacophore hit rate and better synthetic accessibility in comparison with the β-sampling. The S4MPLE docking method was chosen because of the higher values of AUC in the above-discussed benchmarking tests.

The results of the docking studies given in Figure 13 (at left) show that the average docking score of the generated compounds (-63.89) is slightly better than the average score of the active molecules from ChEMBL23 (-62.54), which means that the generated library is enriched with potential A2a binders. Visual pose analysis of the top-ranked generated compounds shows the interaction pattern (with ASN253, PHE168A and hydrophobic parts of the pocket) is similar to that in the PDB structures of the complexes of the A2a receptors with the active ligands used by us for developing the pharmacophore model.

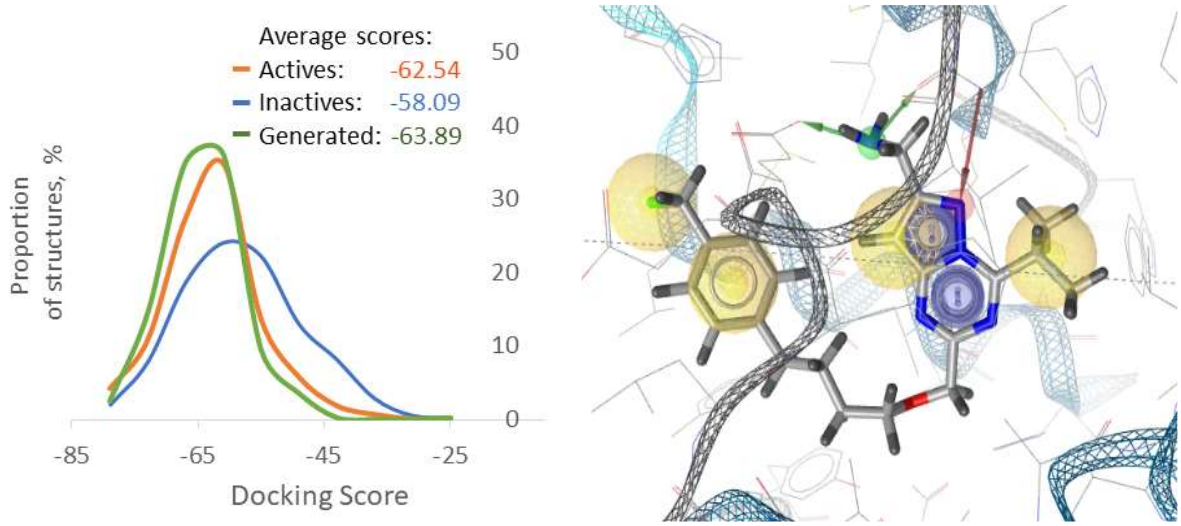


Figure 13. (Left) Histogram of docking scores of the generated compounds and compounds with experimentally measured activity for the binding pocket of the adenosine receptor A2a (PDB 2YDO). (Right) An example of the top-scored docking pose of one of the generated molecules

6. Conclusions

We have developed a method of *de novo* design of biologically active molecules based on the autoencoder sequence-to-sequence neural network and Generative Topographic Mapping. Encoding of SMILES strings into a latent space of about 100 dimensions has been proven to occur with virtually no information loss (excellent SMILES reconstruction rates). Furthermore,

this work showed that the obtained "latent" vectors are molecular descriptors on par with any chemical relevant, information-rich molecular descriptors, such as ISIDA fragment counts. They are equally able to support highly predictive, polypharmacologically competent "universal" maps as the latter. Thus, the herein constructed latent-space-driven "universal" map GTM provided a direct and intuitive access to the latent descriptor space, which means that sampling of chemical space can be "driven" by the map, towards the highly relevant zones of a drug discovery project. Chemical space sampling can be intuitively "monitored" in 2D on the map. Driving through the chemical space can be achieved both by stochastic methods (here, an evolutionary algorithm) guided by the GTM manifold, or by direct normal sampling of the neighborhood of relevant GTM nodes, having predominantly active ligands as residents. The robust link from the herewith visited latent space zones to actual SMILES strings is provided by a well-trained decoder, with very high propensity to generate chemically valid structures. The unique combination of these two features promised to generate *de novo* focused molecular libraries, the quality of which can be rigorously tuned according to multiple criteria:

- proximity to the GTM manifold, as an implicit check of the AD of the GTM model (the activity class landscape)
- the probability to be active, returned as a function of the likelihood to encounter confirmed A2a actives in the area of projection of the new candidate (with implicit rejection of candidates projected in empty zones of the GTM landscape, as a second AD criterion)
- synthetic accessibility, as encoded by Peter Ertl's SA criterion
- absence of "unwanted" reactive groups

For the herein addressed study case, the A2A receptor, *de novo* compounds of correct synthetic accessibility (both by score and by human evaluation) were discovered to reside in chemical space zones predominantly populated by A2A actives, all while not being structurally redundant (in terms of Tanimoto dissimilarity) to the latter. Pending real synthesis and experimental validation, alternative chemoinformatics approaches that do not rely on machine-learned models (pharmacophore screening and docking) seem to corroborate the high likelihood of real, original actives being present in these focused libraries.

7. Acknowledgements

BS thanks the French Embassy in Russia for the MSc fellowship

8. References

1. Varnek, A.; Baskin, I. I., Chemoinformatics as a Theoretical Chemistry Discipline. *Mol. Inf.* **2011,** *30* (1), 20-32.
2. Walters, W. P.; Stahl, M. T.; Murcko, M. A., Virtual screening - an overview. *Drug Discov. Today* **1998,** *3* (4), 160-178.
3. Seifert, M. H.; Kraus, J.; Kramer, B., Virtual high-throughput screening of molecular databases. *Curr. Opin. Drug. Discov. Devel.* **2007,** *10* (3), 298-307.
4. Ripphausen, P.; Nisius, B.; Peltason, L.; Bajorath, J., Quo vadis, virtual screening? A comprehensive survey of prospective applications. *J. Med. Chem.* **2010,** *53* (24), 8461-8467.
5. Ripphausen, P.; Nisius, B.; Bajorath, J., State-of-the-art in ligand-based virtual screening. *Drug Discovery Today* **2011,** *16* (9-10), 372-376.
6. Böhm, H. J., Current computational tools for *de novo* ligand design. *Curr. Opin. Biothechnology* **1996,** *7*, 433-436.
7. Schneider, G.; Lee, M.-L.; Stahl, M.; Schneider, P., De novo design of molecular architectures by evolutionary assembly of drug-derived building blocks. *J. Comput.-Aided Mol. Des.* **2000,** *14* (5), 487-494.
8. Schneider, G.; Fechner, U., Computer-based de novo design of drug-like molecules. *Nat. Rev. Drug Discovery* **2005,** *4*, 649.
9. Tanrikulu, Y.; Schneider, G., Pseudoreceptor models in drug design: bridging ligand- and receptor-based virtual screening. *Nat. Rev. Drug Discovery* **2008,** *7*, 667.
10. Baskin, I. I.; Gordeeva, E. V.; Devdariani, R. O.; Zefirov, N. S.; Palyulin, V. A.; Stankevich, M. I., Solving the inverse problem of structure-property relations for the case of topological indexes. *Dokl. Akad. Nauk SSSR* **1989,** *307* (3), 613-17.

11. Gordeeva, E. V.; Molchanova, M. S.; Zefirov, N. S., General methodology and computer program for the exhaustive restoring of chemical structures by molecular connectivity indexes. Solution of the inverse problem in QSAR/QSPR. *Tetrahedron Comput. Methodol.* **1990,** *3* (6), 389-415.
12. Skvortsova, M. I.; Baskin, I. I.; Slovokhotova, O. L.; Palyulin, V. A.; Zefirov, N. S., Inverse problem in QSAR/QSPR studies for the case of topological indexes characterizing molecular shape (Kier indices). *J. Chem. Inf. Comput. Sci.* **1993,** *33* (4), 630-634.
13. Skvortsova, M. I.; Baskin, I. I.; Palyulin, V. A.; Slovokhotova, O. L.; Zefirov, N. S., Structural design. Inverse problems for topological indices in QSAR/QSPR studies. In *AIP Conf. Proc. 330. E.C.C.C.1 Comput. Chem. F.E.C.S. Conf., Nancy, France*, Bernardi, F.; Rivail, J.-L., Eds. AIP Press: Woodbury, N.Y., 1995; pp 486-499.
14. Churchwell, C. J.; Rintoul, M. D.; Martin, S.; Visco Jr, D. P.; Kotu, A.; Larson, R. S.; Sillerud, L. O.; Brown, D. C.; Faulon, J.-L., The signature molecular descriptor: 3. Inverse-quantitative structureвЂ“activity relationship of ICAM-1 inhibitory peptides. *J. Mol. Graphics Modell.* **2004,** *22* (4), 263-273.
15. Brown, N.; McKay, B.; Gasteiger, J., A novel workflow for the inverse QSPR problem using multiobjective optimization. *J. Comput.-Aided Mol. Des.* **2006,** *20* (5), 333-341.
16. Miyao, T.; Arakawa, M.; Funatsu, K., Exhaustive structure generation for inverse-QSPR/QSAR. *Mol. Inf.* **2010,** *29* (1-2), 111-125.
17. Miyao, T.; Kaneko, H.; Funatsu, K., Inverse QSPR/QSAR Analysis for Chemical Structure Generation (from y to x). *J. Chem. Inf. Mod.* **2016,** *56* (2), 286-299.
18. Hartenfeller, M.; Zettl, H.; Walter, M.; Rupp, M.; Reisen, F.; Proschak, E.; Weggen, S.; Stark, H.; Schneider, G., DOGS: reaction-driven de novo design of bioactive compounds. *PLoS Comput Biol* **2012,** *8* (2), e1002380.
19. LeCun, Y.; Bengio, Y.; Hinton, G., Deep learning. *Nature* **2015,** *521* (7553), 436-444.
20. Baskin, I. I.; Winkler, D.; Tetko, I. V., A renaissance of neural networks in drug discovery. *Expert Opin. Drug Discovery* **2016,** *11* (8), 785-795.
21. Gawehn, E.; Hiss, J. A.; Schneider, G., Deep Learning in Drug Discovery. *Mol. Inf.* **2016,** *35* (1), 3-14.
22. Weininger, D., SMILES, A Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules. *J. Chem. Inf. Comput. Sci.* **1988,** *28* (1), 31-36.
23. Jin, W.; Barzilay, R.; Jaakkola, T., Junction Tree Variational Autoencoder for Molecular Graph Generation. *ArXiv preprint arXiv:1802.04364* **2018**.
24. De Cao, N.; Kipf, T., MolGAN: An implicit generative model for small molecular graphs. *ArXiv preprint arXiv:1805.11973* **2018**.
25. Simonovsky, M.; Komodakis, N., GraphVAE: Towards Generation of Small Graphs Using Variational Autoencoders. *ArXiv preprint arXiv:1802.03480* **2018**.
26. Convard, T.; Dubost, J.-P.; Le Solleu, H.; Kummer, E., SmilogP: A Program for a Fast Evaluation of Theoretical Log P from the Smiles Code of a Molecule. *Quant. Struct.-Act. Rel.* **1994,** *13* (1), 34-37.
27. Toropov, A. A.; Rasulev, B. F.; Leszczynska, D.; Leszczynski, J., Multiplicative SMILES-based optimal descriptors: QSPR modeling of fullerene C(60) solubility in organic solvents. *Chemical Physics Letters* **2008,** *457* (4-6), 332-336.
28. Toropov, A. A.; Toropova, A. P.; Benfenati, E., SMILES-based optimal descriptors: QSAR modeling of carcinogenicity by balance of correlations with ideal slopes. *European Journal of Medicinal Chemistry* **2010,** *45* (9), 3581-3587.
29. Worachartcheewan, A.; Mandi, P.; Prachayasittikul, V.; Toropova, A. P.; Toropov, A. A.; Nantasenamat, C., Large-scale QSAR study of aromatase inhibitors using SMILES-based descriptors. *Chemom. Intell. Lab. Syst.* **2014,** *138*, 120-126.

30. 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* **2018,** *4* (1), 120-131.
31. Yuan, W.; Jiang, D.; Nambiar, D. K.; Liew, L. P.; Hay, M. P.; Bloomstein, J.; Lu, P.; Turner, B.; Le, Q.-T.; Tibshirani, R.; Khatri, P.; Moloney, M. G.; Koong, A. C., Chemical Space Mimicry for Drug Discovery. *J. Chem. Inf. Mod.* **2017,** *57* (4), 875-882.
32. Thomas, B.; Marcus, O.; Ola, E.; Jürgen, B.; Hongming, C., Application of Generative Autoencoder in De Novo Molecular Design. *Mol. Inf.* **2018,** *37* (1-2), 1700123.
33. Anvita, G.; T., M. A.; H., H. B. J.; A., F. J.; Petra, S.; Gisbert, S., Generative Recurrent Networks for De Novo Drug Design. *Mol. Inf.* **2018,** *37* (1-2), 1700111.
34. Olivecrona, M.; Blaschke, T.; Engkvist, O.; Chen, H., Molecular de-novo design through deep reinforcement learning. *J. Cheminf.* **2017,** *9* (1), 48.
35. Putin, E.; Asadulaev, A.; Ivanenkov, Y.; Aladinskiy, V.; Sánchez-Lengeling, B.; Aspuru-Guzik, A.; Zhavoronkov, A., Reinforced Adversarial Neural Computer for De Novo Molecular Design. *J. Chem. Inf. Mod.* **2018**.
36. Popova, M.; Isayev, O.; Tropsha, A., *Deep Reinforcement Learning for De-Novo Drug Design*. 2017.
37. Ertl, P.; Lewis, R.; Martin, E.; Polyakov, V., *In silico generation of novel, drug-like chemical matter using the LSTM neural network*. 2017.
38. Sutton, R. S.; Barto, A. G., *Reinforcement learning: An introduction*. Cambridge ed.; MIT Press: Cambridge, 1998.
39. Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y., Generative Adversarial Nets. *arxiv preprint arXiv:1406.2661* **2014**.
40. Guimaraes, G. L.; Sanchez-Lengeling, B.; Farias, P. L. C.; Aspuru-Guzik, A., Objective-Reinforced Generative Adversarial Networks (ORGAN) for Sequence Generation Models. *ArXiv preprint arXiv:1705.10843* **2017**.
41. Hinton, G. E.; Zemel, R. S., Autoencoders, minimum description length and Helmholtz free energy. In *Proceedings of the 6th International Conference on Neural Information Processing Systems*, Morgan Kaufmann Publishers Inc.: Denver, Colorado, 1993; pp 3-10.
42. Bengio, Y.; Courville, A.; Vincent, P., Representation learning: A review and new perspectives. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **2013,** *35* (8), 1798-1828.
43. Karpov, P. V.; Osolodkin, D. I.; Baskin, I. I.; Palyulin, V. A.; Zefirov, N. S., One-class classification as a novel method of ligand-based virtual screening: The case of glycogen synthase kinase 3OI inhibitors. *Bioorg. Med. Chem. Lett.* **2011,** *21* (22), 6728-6731.
44. Sutskever, I.; Vinyals, O.; Le, Q. V., Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, MIT Press: Montreal, Canada, 2014; pp 3104-3112.
45. Xu, Z.; Wang, S.; Zhu, F.; Huang, J., Seq2seq Fingerprint: An Unsupervised Deep Molecular Embedding for Drug Discovery. In *Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology,and Health Informatics*, ACM: Boston, Massachusetts, USA, 2017; pp 285-294.
46. Kingma, D. P.; Welling, M., Auto-Encoding Variational Bayes. *ArXiv preprint arXiv:1312.6114* **2014**.
47. Gómez-Bombarelli, R.; Wei, J. N.; Duvenaud, D.; Hernández-Lobato, J. M.; Sánchez-Lengeling, B.; Sheberla, D.; Aguilera-Iparraguirre, J.; Hirzel, T. D.; Adams, R. P.; Aspuru-Guzik, A., Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Central Science* **2018,** *4* (2), 268-276.
48. Makhzani, A.; Shlens, J.; Jaitly, N.; Goodfellow, I.; Frey, B., Adversarial Autoencoders. *ArXiv preprint arXiv:1511.05644* **2016**.

49. Gaspar, H. A.; Baskin, I. I.; Marcou, G.; Horvath, D.; Varnek, A., Stargate GTM: Bridging Descriptor and Activity Spaces. *J. Chem. Inf. Mod.* **2015,** *55* (11), 2403-2410.
50. Bishop, C. M.; Svensén, M.; Williams, C. K. I., GTM: The Generative Topographic Mapping. *Neural Comput.* **1998,** *10* (1), 215-234.
51. Bishop, C. M.; Svensén, M.; Williams, C. K. I., Developments of the generative topographic mapping. *Neurocomputing* **1998,** *21* (1-3), 203-224.
52. Gaspar, H. A.; Baskin, I. I.; Marcou, G.; Horvath, D.; Varnek, A., Chemical Data Visualization and Analysis with Incremental Generative Topographic Mapping: Big Data Challenge. *J. Chem. Inf. Mod.* **2015,** *55* (1), 84-94.
53. Gaspar, H. A.; Baskin, I. I.; Marcou, G.; Horvath, D.; Varnek, A., GTM-Based QSAR Models and Their Applicability Domains. *Mol. Inf.* **2015,** *34* (6-7), 348-356.
54. Gaspar, H. A.; Marcou, G.; Horvath, D.; Arault, A.; Lozano, S.; Vayer, P.; Varnek, A., Generative Topographic Mapping-Based Classification Models and Their Applicability Domain: Application to the Biopharmaceutics Drug Disposition Classification System (BDDCS). *J. Chem. Inf. Mod.* **2013,** *53* (12), 3318-3325.
55. Gaspar, H. A.; Sidorov, P.; Horvath, D.; Baskin, I. I.; Marcou, G.; Varnek, A., Generative Topographic Mapping Approach to Chemical Space Analysis. In *Frontiers in Molecular Design and Chemical Information Science - Herman Skolnik Award Symposium 2015: Jürgen Bajorath*, American Chemical Society: 2016; Vol. 1222, pp 211-241.
56. Klimenko, K.; Marcou, G.; Horvath, D.; Varnek, A., Chemical Space Mapping and Structure-Activity Analysis of the ChEMBL Antiviral Compound Set. *J. Chem. Inf. Mod.* **2016,** *56* (8), 1438-1454.
57. Sidorov, P.; Gaspar, H.; Marcou, G.; Varnek, A.; Horvath, D., Mappability of drug-like space: towards a polypharmacologically competent map of drug-relevant compounds. *J. Comput.-Aided Mol. Des.* **2015,** *29* (12), 1087-1108.
58. Hollaar, L. A., Direct Implementation of Asynchronous Control Units. *IEEE Trans. Comput.* **1982,** *31* (12), 1133-1141.
59. Alain, G.; Bengio, Y., What regularized auto-encoders learn from the data-generating distribution. *J. Mach. Learn. Res.* **2014,** *15* (1), 3563-3593.
60. Vinyals, O.; Le, Q., A neural conversational model. *arXiv preprint arXiv:1506.05869* **2015**.
61. Amiriparian, S.; Freitag, M.; Cummins, N.; Schuller, B. In *Sequence to sequence autoencoders for unsupervised representation learning from audio*, Proc. of the DCASE 2017 Workshop, 2017.
62. Williams, R. J.; Zipser, D., A learning algorithm for continually running fully recurrent neural networks. *Neural Comput.* **1989,** *1* (2), 270-280.
63. Hochreiter, S.; #252; Schmidhuber, r., Long Short-Term Memory. *Neural Comput.* **1997,** *9* (8), 1735-1780.
64. Schuster, M.; Paliwal, K. K., Bidirectional recurrent neural networks. *IEEE Trans. Signal Proces.* **1997,** *45* (11), 2673-2681.
65. Ioffe, S.; Szegedy, C., Batch normalization: accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, JMLR.org: Lille, France, 2015; pp 448-456.
66. Maniyar, D. M.; Nabney, I. T.; Williams, B. S.; Sewing, A., Data Visualization during the Early Stages of Drug Discovery. *J. Chem. Inf. Mod.* **2006,** *46* (4), 1806-1818.
67. Owen, J. R.; Nabney, I. T.; Medina-Franco, J. L.; López-Vallejo, F., Visualization of Molecular Fingerprints. *J. Chem. Inf. Mod.* **2011,** *51* (7), 1552-1563.
68. Kireeva, N.; Baskin, I. I.; Gaspar, H. A.; Horvath, D.; Marcou, G.; Varnek, A., Generative Topographic Mapping (GTM): Universal Tool for Data Visualization, Structure-Activity Modeling and Dataset Comparison. *Mol. Inf.* **2012,** *31* (3-4), 301-312.

69. Glavatskikh, M.; Madzhidov, T.; Baskin, I. I.; Horvath, D.; Nugmanov, R.; Gimadiev, T.; Marcou, G.; Varnek, A., Visualization and Analysis of Complex Reaction Data: The Case of Tautomeric Equilibria. *Mol. Inf.* **2018,** *37* (9-10), 1800056.
70. Horvath, D.; Baskin, I.; Marcou, G.; Varnek, A., Generative Topographic Mapping of Conformational Space. *Mol. Inf.* **2017,** *36* (10), 1700036.
71. Baskin, I. I.; Solov'ev, V. P.; Bagatur'yants, A. A.; Varnek, A., Predictive cartography of metal binders using generative topographic mapping. *J. Comput.-Aided Mol. Des.* **2017,** *31* (8), 701-714.
72. Gaspar, H. A.; Baskin, I. I.; Varnek, A., Visualization of a Multidimensional Descriptor Space. In *Frontiers in Molecular Design and Chemical Information Science - Herman Skolnik Award Symposium 2015: Juergen Bajorath*, American Chemical Society: 2016; Vol. 1222, pp 243-267.
73. Ertl, P.; Schuffenhauer, A., Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of Cheminformatics* **2009,** *1* (1), 8.
74. Hoffer, L.; Chira, C.; Marcou, G.; Varnek, A.; Horvath, D., S4MPLE-Sampler for Multiple Protein-Ligand Entities: Methodology and Rigid-Site Docking Benchmarking. *Molecules (Basel, Switzerland)* **2015,** *20* (5), 8997-9028.
75. Hoffer, L.; Renaud, J.-P.; Horvath, D., In Silico Fragment-Based Drug Discovery: Setup and Validation of a Fragment-to-Lead Computational Protocol Using S4MPLE. *J. Chem. Inf. Model.* **2013,** *53* (4), 836-51.
76. Hoffer, L.; Horvath, D., S4MPLE - Sampler For Multiple Protein-Ligand Entities: Simultaneous docking of several entities. *J Chem Inf Model* **2012,** *53* (1), 88-102.
77. BioSolveIt FlexX relelase 3. http://www.biosolveit.de/FlexX/.
78. Rcsb Protein Data Bank.
79. ChemAxon Tautomer Plugin. http://www.chemaxon.com/marvin-archive/4.1.3/marvin/chemaxon/marvin/help/calculator-plugins.html#tautomer (accessed Oct. 2011).
80. ChemAxon pKa Calculator Plugin. https://www.chemaxon.com/products/calculator-plugins/property-predictors/ (accessed Feb. 2013).
81. ChemAxon Calculation of Partial Charge Distributions. http://www.chemaxon.com/marvin/help/calculations/charge.html (accessed Feb. 2009).
82. Case, D. A.; Darden, T. A.; Cheatham, T. E.; Simmerling, C. L.; Wang, J.; Duke, R. E.; Luo, R.; Walker, R. C.; Zhang, W.; Merz, K. M.; Roberts, B.; Hayik, S.; Roitberg, A.; Seabra, G.; Swails, J.; Goetz, A. W.; Kolossvary, I.; Wong, K. F.; Paesani, F.; Vanicek, J.; Wolf, R. M.; Liu, J.; Wu, X.; Brozell, S. R.; Steinbrecher, T.; Gohlke, H.; Cai, Q.; Ye, X.; Hsieh, M.-J.; Cui, G.; Roe, D. R.; Mathews, D. H.; Seetin, M. G.; Salomon-Ferrer, R.; Sagui, C.; Babin, V.; Luchko, T.; Gusarov, S.; Kovalenko, A.; Kollman, P. A., AMBER 12. *University of California* **2012**.
83. Wang, J. M.; Wolf, R. M.; Caldwell, J. W.; Kollman, P. A.; Case, D. A., Development and testing of a general amber force field. *Journal of Computational Chemistry* **2004,** *25* (9), 1157-1174.
84. Zhenin, M.; Bahia, M. S.; Marcou, G.; Varnek, A.; Senderowitz, H.; Horvath, D., Rescoring of docking poses under Occam's Razor: are there simpler solutions? *Journal of Computer-Aided Molecular Design* **2018**.
85. Varnek, A.; Fourches, D.; Hoonakker, F.; Solov'ev, V., Substructural fragments: an universal language to encode reactions, molecular and supramolecular structures. *J. Comput.-Aided Mol. Des.* **2005,** *19* (9), 693-703.