Skolkovo Institute of Science and Technology
Skolkovo Instutute of Science and Technology

EXPLORATION OF CHEMICAL SPACE BY MACHINE LEARNING

Doctoral Thesis
by
Sergey Sosnin

Doctoral Program in Computational and Data Science and Engineering

Supervisor
Vice-President for Artificial Intelligence and Mathematical Modelling,
Professor, Maxim Fedorov

Moscow – 2020

I hereby declare that the work presented in this thesis was carried out by myself at Skolkovo Institute of Science and Technology, Moscow, except where due acknowledgement is made, and has not been submitted for any other degree.

Candidate: Sergey Sosnin

Supervisor (Prof. Maxim V. Fedorov)

EXPLORATION OF CHEMICAL SPACE BY MACHINE LEARNING

by

Sergey Sosnin

Submitted to the Center for Computational and Data-Intensive Science and Engineering and Innovation
on September 2020, in partial fulfillment of the requirements for the Doctoral Program in Computational and Data Science and Engineering

Abstract

The enormous size of potentially reachable chemical space is a challenge for chemists who develop new drugs and materials. It was estimated as 10^{60} and, given such large numbers, there is no way to analyze chemical space by brute-force search. However, the extensive development of techniques for data analysis provides a basis to create methods and tools for the AI-inspired exploration of chemical space.

Deep learning revolutionized many areas of science and technology in recent years, i.e., computer vision, natural language processing, and machine translation. However, the potential of application of these methods for solving chemoinformatics challenges has not been fully realized yet. In this research, we developed several methods and tools to probe chemical space for predictions of properties of organic compounds as well as for visualizing regions of chemical space and for the sampling of new compounds. We explored a new type of 3D spatial descriptors and demonstrated that one can use these descriptors with 3D Convolutional neural networks for the bioconcentration factor prediction. In our research, we proved that multitask deep learning can achieve better performance been compared with single-task learning. To improve the navigation trough chemical space, we have developed a parametric t-SNE method for visualization of large chemical datasets. We developed molecular grammars for the generation of the organic structures and implemented a library to work with these grammars: Legogram. These methods and tools provide the environment for the AI-driven exploration of chemical space. We believe that our findings will accelerate the drug discovery process.

Publications

The main results of the work have been published in papers:

1. Sergey Sosnin, Dmitry Karlov, Igor V. Tetko, and Maxim V. Fedorov. Comparative Study of Multitask Toxicity Modeling on a Broad Chemical Space. *Journal of Chemical Information and Modeling*, 2018a. ISSN 1549-9596; 1549-960X. doi:10.1021/acs.jcim.8b00685. Place: United States Publisher: United States (**Q1** journal, IF=4.54)

2. Sergey Sosnin, Mariia Vashurina, Michael Withnall, Pavel Karpov, Maxim Fedorov, and Igor V. Tetko. A Survey of Multi-Task Learning Methods in Chemoinformatics. *Molecular informatics*, 37, 2018c. ISSN 1868-1751; 1868-1743. doi:10.1002/minf.201800108. Place: Weinheim, Germany, Germany Publisher: Weinheim, Germany, Germany (**Q2** journal, IF=2.74)

3. Sergey Sosnin, Maksim Misin, David Palmer, and Maxim Fedorov. 3D matters! 3D-RISM and 3D convolutional neural network for accurate bioaccumulation prediction. *Journal of Physics Condensed Matter*, 30(32), 2018b. ISSN 1361-648X; 0953-8984. doi:10.1088/1361-648x/aad076. Place: United Kingdom Publisher: United Kingdom (**Q1** journal, IF=2.7)

4. Dmitry S. Karlov, Sergey Sosnin, Igor V. Tetko, and Maxim V. Fedorov. Chemical space exploration guided by deep neural networks. *RSC advances*, (9): 5151–5157, 2019. ISSN 2046-2069. doi:10.1039/c8ra10182e. Place: United Kingdom Publisher: United Kingdom (**Q1** journal, IF=3.0)

These publications describe findings which are related to the study:

1. Yury I. Kostyukevich, Gleb Vladimirov, Elena Stekolschikova, Daniil G. Ivanov, Arthur Yablokov, Alexander Ya Zherebker, Sergey Sosnin, Alexey Orlov, Maxim Fedorov, Philipp Khaitovich, and Evgeny N. Nikolaev. Hydrogen/Deuterium exchange aids compounds identification for LC-MS and MALDI imaging lipidomics. *Analytical Chemistry*, 2019. ISSN 1520-6882; 0003-2700. doi:10.1021/acs.analchem.9b02461. Place: United States Publisher: United States (**Q1** journal, IF=6.35)

2. Dmitry S. Karlov, Sergey Sosnin, Maxim V. Fedorov, and Petr Popov. graphDelta: MPNN scoring function for the affinity prediction of protein–ligand complexes. *ACS Omega*, 5(10):5150–5159, March 2020. doi:10.1021/acsomega.9b04162. URL https://doi.org/10.1021/acsomega.9b04162 (**Q1** journal, IF=2.87)

3. Sergey Osipenko, Inga Bashkirova, Sergey Sosnin, Oxana Kovaleva, Maxim Fedorov, Eugene Nikolaev, and Yury Kostyukevich. Machine learning to predict retention time of small molecules in nano-HPLC. *Analytical and Bioanalytical Chemistry*, August 2020. doi:10.1007/s00216-020-02905-0. URL https://doi.org/10.1007/s00216-020-02905-0 (**Q1** journal, IF=3.63)

Acknowledgments

Firstly, I would like to express my sincere gratitude to my advisor, Vice-President for Artificial Intelligence and Mathematical Modelling of Skoltech, Prof. Maxim Fedorov, for the support during the PhD program in Skoltech. His openness to new ideas inspired me for this research project. I deeply grateful for Prof. Igor V. Tetko for the fruitful cooperation and ultimate support during this research. I acknowledge my co-authors and colleagues Dmitry Karlov, Petr Popov, Olga Novitskaya and Yury I. Kostyukevich for the beneficial teamwork. I am very grateful to my wife and colleague, Ekaterina Sosnina, who supported me a lot during my PhD program.

Contents

1 Introduction **9**

2 The Literature Review **17**
2.1 Molecular Descriptors 20
2.2 Machine Learning Methods in QSAR/QSPR Studies 24
2.3 Models Validation and Performance Measurement 31
2.4 3D Reference Interaction Site Model (3D-RISM) 36
2.5 Multitask Learning for Chemical Data Analysis 37

3 3D RISM and 3D CNNs for bioconcentration prediction **44**
3.1 Materials and Methods 46
3.2 Results and Discussion 52
3.3 Conclusions 55

4 Multitask learning for acute toxicity modelling **56**
4.1 Materials and Methods 58
4.2 RTECS Chemical Space 60
4.3 Correlation Analysis of Endpoints 64
4.4 Comparison of Models 64
4.5 Attributed Modeling 68
4.6 Conclusions 73

5 Chemical space visualization guided by deep learning **75**
5.1 Materials and Methods 76
5.1.1 Datasets 77
5.1.2 Parametric t-SNE 78
5.1.3 Dimensionality Reduction Methods 82
5.1.4 Validation protocols 82
5.2 Results and Discussion 82
5.3 Conclusions 87

6 Legogram: Molecular grammars **88**
6.1 Formal Definition of Molecular Grammars 90
6.2 Implementation of Molecular Grammars 92
6.3 Validation of the Algorithm 97
6.4 Grammar Compression 98
6.5 Generative Models 100

6.5.1 Legogram-based Generative Modeling 101
6.5.2 Optimization of Drug-likeness 102
6.5.3 Synthetic Accessibility 104
6.6 Sampling Compounds from Chemical Space 106
6.7 Conclusions 110

7 Conclusions **114**

Glossary **120**

Bibliography **123**

8 Supplementary Material **147**

Chapter 1

Introduction

There is a common opinion that pharmaceutical R & D is in crisis Pammolli et al. [2011]. Companies have to spend more than ten years and more than a billion dollars on marketing a new drug globally. But the problem is not only about high costs and risks. The "gold-mines" – scaffolds that produced blockbuster drugs – are dried out. Investigations of new, unexplored regions of chemical space is a risky business because the clinical trial failure ratio is high. A possible solution is the intensive investigation of new areas of chemical space – to find new scaffolds for new drugs. The researchers should go to *terra-incognita* of chemical space, because there is a supreme request for exploration of chemical space for a search of new drug candidates. COVID-19 outbreak stressed the basic fact that humanity does not have an adequate response to viral diseases, and infection outbreaks are not something that we thought was a problem of the XIX century. Antibiotics resistance is another possible pain spot for humankind Ventola [2015]. Antibiotics saved millions of lives, but now we faced with the rise of resistant bacteria. Horizontal gene transfer allows bacteria to exchange genes responsible for the deactivation of antibacterial drugs Barlow [2009], Lerminiaux and Cameron [2019]. On the other hand, the pharmaceutical industry requires time for clinical trials, limiting our chances to combat the next infection crisis. Non-communicable diseases are the leading cause of death all around the world. Despite the developed treatment strategies, there are no "silver bullets" for the majority of chronic diseases. Individuals vary in their response to therapy, which limits the successful treatment abilities. To provide new abilities

for doctors, one needs to develop a number of different medicines, with different mechanisms of actions, if possible.

The glaring example of neediness for new treatments is orphan diseases. The economic reasons restrict the interest of pharmaceutical companies to invest money in research projects in this field Meekings et al. [2012]. Many countries apply special laws to motivate drug development (i.e. US Rare Diseases Act of 2002, or EC Regulation No 141/2000). Under these laws, orphan drugs are eligible to fast track approval procedure. Computations can provide a theoretical basis to support fast track.

Computational methods can boost the drug discovery pipeline. But there are fundamental problems behind the direct application of calculations to drug discovery, but to reveal them, we should make a short philosophical introduction.

David Deutsch, in his famous book *The Fabric of Reality*, discussed two possible views on the world. The first one is a *holistic view* – when one regards a phenomenon been a result of the highest possible level interactions. Another possible view is *reductionism*: the idea that a phenomenon can be explained by reduction to the simplest essences. These extreme views exist in chemistry, and, maybe in molecular science, the border between these views is quite sharp. In molecular science *reductionism* is so-called "bottom-up" approaches. Under this idea, we regard complex systems as reducible to small parts with the defined behavior. The origin of this approach goes deep into the past from Greek philosophers, who introduced the first atomistic essence of the matter though classic molecular models until the quantum mechanical view. Quantum Mechanics gave researchers a unique tool for the rational calculation of properties of molecules; however, the computational costs of such calculations are very high, and the application of QM calculations to the prediction of various macroscopic chemical and biological properties is limited. It is appropriate here to cite Walter Kohn's Nobel lecture: *"There is an oral tradition that, shortly after Schroedinger's equation for the electronic wave-function Ψ had been put forward and spectacularly validated for simple small systems like He and H_2 P.M. Dirac declared that chemistry had come to an end - its content was entirely contained in that powerful equation. Too bad, he is said to have added, that in al-*

most all cases, this equation was far too complex to allow a solution." Koch [1999]. The quantum representation of molecules is the most precise one; however, there are two significant limitations on the way: i) one can obtain an analytical solution only for the most straightforward cases like Hydrogen atom and Helium ions. ii) the extreme complexity of traditional (wavefunction-based methods). With modern computational power, it is hard to model large clusters of molecules by quantum computing methods. To combat this problem, several approximation methods have been developed. Semi-empirical methods use some parametric-based approximations to speed-up the computations. One of the most popular techniques in the analysis of middle-scale molecular systems is Molecular Mechanics (MM). In Molecular Mechanics, the dynamic of a molecular system is determined by the integration of kinematic equations. Force fields – parametric models that describe the interactions between atoms are used. If anyone wants to analyze even larger molecular systems, he/she has to study it by coarse-grained models. At this level, one can model extensive systems, but in the price of the accuracy of the models and the narrowing of the applicability domain. We can see a trade-off between the accuracy of models, the speed of computations, and the applicability domain. In machine learning the trade-off between the accuracy of models and applicability domain is well-known as *bias-variance trade-off* and discussed in detail in *Section* 2.3.

Indeed, modern QM approaches can model molecular systems up to 10^3 atoms in a reasonable time, but that is far from the modeling of complex biological interactions on an organism level. To combat this problem, one can boost the calculations by approximating some parts of them by parametric models[1]. It is a holistic or "top-down" approach. Under this idea, one does not build a reducible model of the process but approximates the process by statistical or machine learning methods. These methods are highly parametric so that some researchers regard them as "black-boxes." The latter means that it is often tricky to analyze and understand the influence of the input variables on the process (and the model of the process too). Generally, top-down approaches are quick, but they suffer heavily from *bias-*

[1]It is worth mentioning that QM calculations are and parametric too, and a basis defines the parameterization scheme

variance trade-off, and it is challenging to build a model with high accuracy and applicability. This idea lies behind semi-empirical approaches and molecular mechanics. Molecular mechanics can be applied to model systems with millions of atoms; by the price of neglecting the quantum correlations. There are many successful applications of Molecular Mechanics in drug discovery, mostly for exploring drug-target interactions, but the applicability for the complex biological systems is still unreachable. Another concern of parametric models is the limited applicability domain. Generally, these models can have good (and guaranteed) performance only within a local chemical domain. Beyond this domain, the robustness of parametric models can be unsatisfactory.

But the problem is much more challenging; for drug discovery, one should sample compounds with desired properties from chemical space. Chemical space is ultimately large; it is estimated as 10^{60} of organic compounds that possibly exist and follow the basic rules for drug-like compounds Kirkpatrick and Ellis [2004]. This number is so vast that there are no ways to explore chemical space by experimental or even computational brute-force. Traditional experimental procedures can cover only a tiny part of the chemical space. Computational approaches that are based on quantum or molecular modeling are computationally expensive and can not be used extensively for the exploration of the chemical space. General statistical methods in chemoinformatics, known as Quantitative Structure–Activity/Property Relationship (QSAR/QSPR), cover separate regions of the chemical space and can be used only within their applicability domain. However, the distribution of desired and undesired molecules in chemical space is not uniform. Some parts of chemical space are filled with molecules with appropriate chemical/biological properties, and the exploration for these regions makes sense. In contrast, there are regions filled with "non-drug-like" molecules, and they are of no interest in further exploration. One can imagine "drug-like" parts of chemical space as full of life archipelagos, among the lifeless ocean of non-druglike compounds. But if someone wants to cross the ocean, he needs tools for navigation. It is also right for chemical space. There is an ultimate need to build tools that can guide chemists in this space to obtain compounds with desired properties and to avoid undesirable ones. Drug candidates

should, on the one hand, have desired bioactivity profiles, and on the other hand, should satisfy many additional criteria: be non-toxic, have desired Absorption, Distribution, Metabolism, and Excretion (ADME) properties, etc. That is not all; these compounds should follow additional restrictions: be patentable, synthetically accessible, eco-friendly for manufacturing. Since XIX century there is an empirical rule in organic chemistry that: structure determines properties of organic compounds. But the formal description of structure-activity relationship became possible only in the second part of XX century. QSAR/QSPR opened the doors for the rational design of drugs and chemical compounds, but the potential of these methods has not been fully revealed yet. The complexity and non-linearity of the relations between chemical structures and corresponding activities/properties limit the applicability of these methods to the real problems. However, even in the case of pure computational experiments, this problem remains changeable. Thus, it is absolutely impossible to build a computer that will keep in memory 10^{60} structures. What is the solution? It is worth remembering that chemists worked before the computational era on the base of such ephemeral matter as chemical intuition Pedreira et al. [2019], Gomez [2018]. This intuition navigates chemists though the chemical space but been personal and unquantified, it limits the abilities for the high-performance exploration of chemical space. Moreover, comparing chemists and Machine learning-based methods Kutchukian et al. [2012] found out that "*chemists greatly simplified the problem, typically using only 1–2 of many possible parameters when making their selections.*"

Machine (and especially deep) learning is a possible way of quantifying chemical intuition on a universal and reproducible basis Moosavi et al. [2019], Jaeger et al. [2018]. This challenge motivates us to develop new methods that can materialize the chemical intuition and use it to AI-inspired molecules discovery.

The doctoral study aims to develop new ways for ML-based exploration of chemical space to accelerate drug-discovery process.

In **Chapter 1** we justify the significance of the development of new AI-based methods and tools for the exploration of chemical space.

In **Chapter 2** we review methods and materials that are used in this research, briefly describe machine learning and deep learning algorithms, along with ap-

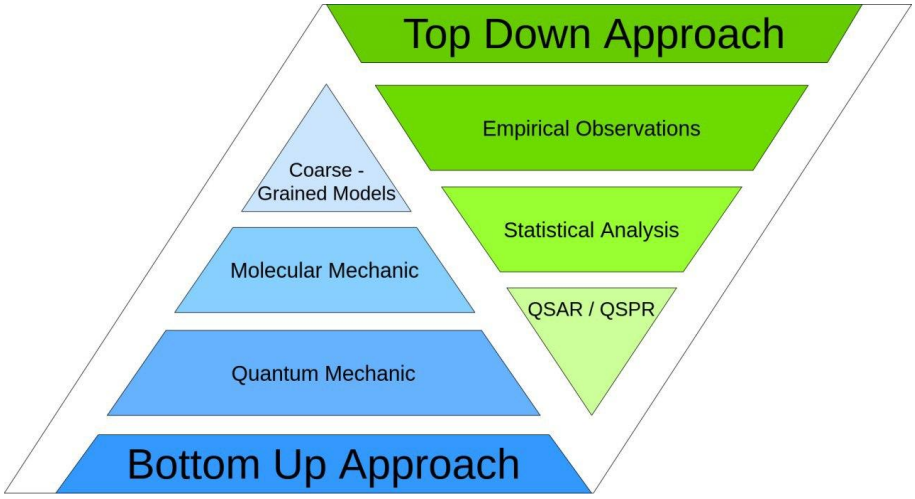


Figure 1-1: "Bottom-up" vs "Top-down" approaches

proaches for the estimation of the performance of models. Also, we describe 3D RISM method that is the basis for our 3D spatial descriptors.

In **Chapter 3** we discuss a new method for Bioconcentration factor (BCF) prediction based on 3D Convolutional neural networks. As inputs for CNNs we use 3D scalar fields that represent the density of solvent sites around solute. We designed 3D CNN ActiveNet4 and compared the performance of our method with standard QSPR approaches. We demonstrated that 3D CNNs can successfully be applied to biological-related problems.

In **Chapter 4** we analyze the application of multitask learning to the animal acute toxicity modeling. We performed a comparative study of multitask toxicity modeling on a broad chemical space. Our experiments revealed the most efficient groups of descriptors for the acute toxicity modeling We showed that multitask learning is outperforming both single-task learning methods and common machine learning approaches. However, our experiments revealed that multitask learning can be emulated by attributed learning with similar performance. We discuss the current situation with regulations of QSAR/QSPR modeling and the neediness for multitask learning consideration in QSAR/QSPR regulation.

In **Chapter 5** we describe a method and a tool for the visualization of broad chemical space. We propose a parametric t-SNE method, that can project the chemical compounds from its original descriptors space onto a 2D surface, preserving

their local similarity. We demonstrated that the parametric t-SNE method can generate reasonable projections. On the base of this method, we created a tool for the generation of molecular maps. The tool can be used by chemists to analyze large chemical datasets in handily. We experimentally showed the meaningfulness of our method by training classifies on our 2D representations and comparing their performance.

In **Chapter 6** we describe a Legogram framework we describe a Legogram framework for the construction of molecular structures from "building blocks" preserving chemical validity. This framework is based on the concept of molecular grammars – a special type of graph grammars. We experimentally proved the absolute structural validity of generated molecules and demonstrated that our library could be used for the sampling of chemical compounds from regions of chemical space by reinforcement learning.

The novelty of the research is summed up below:

1. We demonstrated that a hybrid method based on 3D RISM and 3D CNNs can be applied for the Bioconcentration factor prediction

2. We showed the superiority of multitask learning in the prediction of acute toxicity of organic compounds on a broad chemical domain.

3. We developed a new methodology for the visualization of regions of chemical space

4. We developed a graph grammar framework for the generation of valid chemical structures with desired properties. We showed that one can sample compounds from chemical space using our graph grammar framework and reinforcement learning.

To support the practical application of this doctoral research we implemented an online platform – Syntelly[2]. Also, our animal acute toxicity models are freely accessible in OCHEM: `https://ochem.eu//multitox` (login required to

[2]the platform is accessible at app.syntelly.com, is in a very earlier stage and under active development

access). Legogram library is published at GitHub https://github.com/sergsb/LegoGram

Chapter 2

The Literature Review

In this chapter, we propose a review of the approaches and methods used in this study. First, we provide a brief historical overview of statistics and machine learning methods in chemistry. Then we discuss common chemical descriptors. After that, we review machine learning methods and pay special attention to artificial neural networks and deep learning. We discuss 3D RISM method, which can generate 3D spatial molecular descriptors. Also, we discuss the problems behind the correct assessment of models quality.

Historical background of statistical modeling in chemistry (QSAR/QSPR)

Quantitative Structure–Activity/Property Relationship (QSAR/QSPR) – is a common name for many methods based on statistics and machine learning for revealing correlations between chemical structures of compounds and their chemical/biological properties. One can track the roots of QSAR/QSPR in Hammet research Hammett [1937], who carefully studied the effects of substitutes to chemical reactivity. The concept of predicting the properties of organic compounds by statistical modeling is originated from works of Hansch and Fujita [1964]. In these papers, they summed up their findings of correlations between the logarithm of octanol/water partition coefficient (logP) and the biological activity of chemical compounds. They demonstrated that there is an optimum in logP for a particular biological response. They proposed multiple linear regression analysis as a basis for modeling of structure-activity relations. Nowadays, Hansch and Fujita are commonly considered as pioneers of QSAR

studies Martin [2011]. The rise of interest in artificial neural networks in the 80th inspired chemists to apply it to QSAR/QSPR tasks. Prof. Gastaiger, who used neural networks for the prediction of properties of organic compounds, demonstrated the feasibility of this idea Zupan and Gasteiger [1993]. Before deep learning era (which started roughly in 2012) many machine learning methods have been applied to prediction of properties of organic compounds: *Support Vector Machines* Ivanciuc [2007], *Random Forest* Svetnik et al. [2003], *k-nearest neighbors* Kauffman and Jurs [2001], and many others Neves et al. [2018]. In 2012 Merck Inc. published on Kaggle (a popular platform for data analysis competitions) – a dataset of kinase inhibition activity and launched a competition for predicting these values. Hilton's team won this challenge by using a model based on deep neural networks, opening the door for deep learning applications in chemical information. Further, the experience of application of deep neural networks to QSAR problems were summarized in publications of the Merck team Ma et al. [2015a], Xu et al. [2017b].

Basis of QSAR/QSPR

Formally, direct QSAR/QSPR problem can be written as:

$$F(G) = y \tag{2.1}$$

where G is a molecular graph, F – is a QSAR/QSPR model and y is a vector of activities/properties. The problem is that most statistical and machine learning methods operate with numbers, but not graphs. To struggle with it one can calculate the invariants of molecular graphs: *molecular descriptors*. A typical pipeline is to calculate molecular descriptors first and use it for modeling by any machine learning method. So, the QSAR/QSPR problem consists of two parts:

- a transformation of chemical structures to numerical representations
- statistical modeling on these representations

The correct evaluation of the performance of ML models is an important part of QSAR/QSPR pipeline. The statistical nature of QSAR/QSPR models can lead

Figure 2-1: Common QSAR/QSPR pipeline. Molecular structures are converted to vectors of molecular descriptors. Stacked molecular descriptors form a matrix X, and activity/property vector y. Given X and y one can build a model $F(X) = y$

to *overfitting*: a situation when a model can successfully predict the values for the molecules from a training set but fails on a test set. Several validation techniques can be applied to combat overfitting. A brief review of these methods which have been used in the study is given in section 2.3

Inverse-QSAR/QSPR is a problem of generating molecules with desired properties.

$$\begin{gathered} \hat{F}(y) \longrightarrow \{M_1, M_2, ..., M_i\}, \\ \mathbb{E}\left[\{F(M_1), F(M_2), ..., F(M_i)\}\right] \longrightarrow y \end{gathered} \tag{2.2}$$

where y is a vector of chemical or biological properties M is a molecule, $F(M)$ is a direct QSAR/QSPR function which calculates properties of a molecule, $\hat{F}(M)$ – is an inverse QSAR/QSPR function which generates a number of molecular structures which properties are expected to be y, $\mathbb{E}\left[\{F(M_1), F(M_2), ..., F(M_i)\}\right]$ – is average property value over a generated molecular batch. In practice, for chemists solving the inverse task is more important, because it is a way for the "de-novo" generation of chemical compounds with desired properties. However, this problem is more challenging than direct QSAR/QSPR, generally because it requires a method for the generation of molecular structures. While the transformation of graph structures into numerical representation is well-studied (see section 2.1), graphs generation is a challenge. We describe our approach to generative models on the base of molecular grammars in Chapter 6.

2.1 Molecular Descriptors

The most common way how one can represent a molecule is a molecular graph. Under this representation, atoms are regarded as colored (labeled) nodes, and chemical bonds are considered to be undirected weighted edges of a graph. This way allows chemists to use all power of graph theory to analyze chemical compounds. It also simplifies the development of chemoinformatics software by designing it on the top of graph libraries. Invariants of these graphs are called *molecular descriptors* or *descriptors*. Commonly, molecular descriptors are inputs for machine learning models. Nowadays, thousands of molecular descriptors exist. Famous chemoinformatic software Dragon 7 calculates 5,270 molecular descriptors. There are many types of chemical descriptors: structural descriptors, physicochemical, fragmental, quantum descriptors, etc. A reader can familiarize with the whole landscape of molecular descriptors in a comprehensive (but quite outdated) monography Todeschini and Consonni [2000]. Below we will provide a brief explanation of some types of molecular descriptors. Let's start with *topological descriptors* – invariants of molecular graphs that commonly do not regard chemical features of these graphs. A good example of topological descriptors is structural indexes: *Wiener index* Wiener [1947] and *Randić index* Randic [1975]. Wiener index is a sum of the lengths of the shortest paths between all pairs of vertices:

$$W(G) = \sum_{\forall(i,j)} d(v_i, v_j) \tag{2.3}$$

where $d(v_i, v_j)$ – is the shortest path between v_i and v_j. Randić index is given by the Equation 2.4

$$R(G) = \sum_{\forall(i,j)} \frac{1}{\sqrt{deg(v_i)deg(v_j)}} \tag{2.4}$$

where $deg(v_i)$ and $deg(v_j)$ are degrees of a graph. One can see that these indexes do not concern about chemical features (atoms and bonds types) indeed. That is why the discrimination ability of topological indexes is low, and they are not used for QSAR/QSPR solely, but because of their simplicity and low computational costs,

they can be used as hash functions for molecular graphs.

Another popular type of descriptor is the fragment-based descriptors. The decomposition of a molecule in some fragment basis is the main idea of fragment-based descriptors. Formally, it is a decomposition of a molecular graph into a set of subgraphs. The number of occurrences of a certain fragment in a molecule (or just an indicator of occurrence) is a value of the descriptor. Free and Wilson [1964] approach was an early attempt to describe structure-activity relationships as a sum of contributions of individual subfragments in the molecules. One can define this method by equation:

$$Activity = \mu + \sum_i b_i F_i p_i \tag{2.5}$$

where F_i – is a molecular fragment, p_i – is a position of this fragment in a molecule, b_i – is a contribution of a fragment F into the activity. μ – is a mean activity over a dataset. Free-Wilson method is "scaffold-oriented", which means that it requires a dataset with common structures (scaffolds). To obtain a robust model one should have a diverse distribution of substitutes. Also, it worth to mention that Free-Wilson method is linear and can not regard non-linear relations in molecular structures. Free-Wilson approach was a milestone in QSAR/QSPR history Kubinyi [1988]. Next important landmark in fragment-based QSAR/QSPR was a method for algorithmic generation of custom molecular fragments on a dataset Adamson and Bawden [1975]. Scince that time, Fragment-based descriptors (with some variations) have been implemented in many software applications: ISIDA (Ruggiu et al. [2010]), MultiCASE (Klopman [1992]), OCHEM (Sushko et al. [2011]). Fragment-based descriptors demonstrate excellent performance for many tasks, especially for the prediction of the physical-chemical properties of organic compounds. Clear chemical meaning and interpretability is another advantage of these descriptors. But computational complexity and redundancy are major drawbacks to analyze large chemical datasets. Another problem is the non-universality: an applicability domain of a model is technically limited only to subgraphs that occur in the training set.

Fragment-based descriptors are implemented, among others, in aggregated software (programs that calculate many different descriptors of different nature): Dragon,

PyDescriptors, CDK descriptors.

2D structural features of organic compounds are limited to the prediction of biological activity. This activity is based on 3D ligand-protein interactions, which are surely related to 3D conformations of ligands. To overcome this limitation, Cramer et al. [1988] proposed 3D Comparative Molecular Field Analysis (3D CoMFA). This method is alignment-dependent and can be applied only to compounds with a common scaffold. Aligned structures are placed in a 3D grid, and then, the sum of interactions of each molecule with a probe atom is calculated. After that, Partial least squares (PLS) method is used for the prediction of biological activity. This method was widely used for the prognosis of biological activities Melo-Filho et al. [2014]. Further, many other 3D QSAR methods have been developed. Comparative molecular similarity indices analysis (CoMSIA) that is a modified version of 3D CoMFA was proposed by Klebe and Abraham [1999]. The authors replaced Lennard-Jones and Colomb potentials used in 3D CoMFA, with five different indexes, responsible for ligand binding. They claimed that this improvement could help to build easily interpretable models. It was a significant step because the possible interpretability is the advantage of 3D molecular fields. Baskin and Zhokhova [2019] proposed a method of Continuous Molecular Fields. This method is based on the decomposition of interaction pattern into two spatial functions: $X(r)$ – a molecular filed constructed from Gaussian functions placed on atoms and a continuous function $C(r)$ – which plays the role of the regression coefficients. The resulting biological activity is calculated as an overlap integral between these functions. Continuous Molecular Fields are easily interpretable and can be nicely visualized, which helps chemists to understand the nature of activity cliffs in particular targets. In our research, we used 3D descriptors calculated by Corina software.

Common tools and frameworks for molecular descriptors calculation

For more than 40 years, many programs and tools have been created to calculate molecular descriptors. Here, we will give links to the most popular chemoinformatic software with such functionality. This software can be roughly divided into three categories:

- standalone programs (Dragon, PaDEL)
- descriptor modules implemented in chemoinformatics frameworks (CDK descriptors, RDkit descriptors)
- programming libraries (mordred, e3fp)

Among standalone programs, it is worth to mention Dragon 7. It is one of the most reputable proprietary software for chemical descriptors calculations. The last version can calculate 5,270 descriptors and some classes of molecular fingerprints. E-DRAGON (a special edition of Dragon program) implemented in OCHEM has been used in our study. PaDEL-Descriptor Yap [2010] – is another program for calculation molecular descriptors. It is mostly based on Chemistry development kit descriptors, and somehow can be regarded as GUI and command-line interface for CDK descriptors. ChemAxon JChem provides several descriptors on the top of popular chemical platform JChem. Chemistry development kit is a popular Java library for chemoinformatics Willighagen et al. [2017]. There are many chemical descriptors of different nature implemented in this library. We utilized CDK descriptors from OCHEM in our research. We should also mention a popular framework RDkit, that provides many different molecular descriptors. RDkit implementation of extended connectivity fingerprints (Morgan fingerprints) is a popular choice for describing large chemical databases; due to quick calculations.

There are some libraries for the calculation of chemical descriptors. These libraries provide Application programming interface (API) for developers. The authors of *mordred* python package were motivated to create a universal and reproducible framework Moriwaki et al. [2018]. They addressed problems of software bugs, low update frequencies, licensing issues of the existing solutions. They focused on the quality and reproducibility of descriptors between different versions of the library. Since the publication, this library has gained popularity in the community rapidly, and now it can be a first-choice option. In our experiments *mordred* descriptors often demonstrate better performance than extended connectivity fingerprints; however, the processing speed is much lower, which restricts the ability to use *mordred* descriptors for processing large chemical databases.

2.2 Machine Learning Methods in QSAR/QSPR Studies

Machine learning methods are a critical thing for QSAR/QSPR modeling. Commonly, they are divided into two groups: *supervised* and *unsupervised learning*. *Supervised learning* is based on the fitting of parameters of a model by demonstrating a set of *training examples* – are pair of an object (a molecule) and a *supervisory signal*. A supervisory signal can be a class (i.e., toxic/nontoxic) or a continuous value, for example, a boiling point of a molecule. The supervised learning algorithm infers hidden relations in the training dataset to "learn" from labeled data. *Unsupervised learning* is a common name for a group of algorithms, which do not rely on pre-existing *supervisory signals*, but extract this information directly from data. Typical examples of *unsupervised learning* are clustering algorithms and dimensionality reduction methods. *Semi-supervised* learning is a name for hybrid algorithms; they have features in common with *supervised* and *unsupervised learning*. Below, we provide a brief overview of ML methods that we used in this research. Because of importance for our study, artificial neural networks along with deep learning are described in greater details in a separate section 2.2.

Decision trees

The concept of decision trees is essential for further discussion because they are the basis for many efficient algorithms and tools. A decision tree can be represented in a flowchart-like manner, where each node corresponds to a constrain, and a leaf corresponds to an outcome. The significant advantage of this approach is the clarity: a decision tree is fully transparent. However, the discrimination ability of a single decision tree is low, and usually, trees are grouped into ensembles to boost the performance. There is a substantial bias-variance trade-off associated with this method, and some regularisation techniques are required. The problem is the construction of decision trees in a fully automatic way. We will discuss it on the example of CART algorithm. To compute a split, one should understand how to split a node in a tree onto the right and left branches. There are two criteria for that: entropy-based

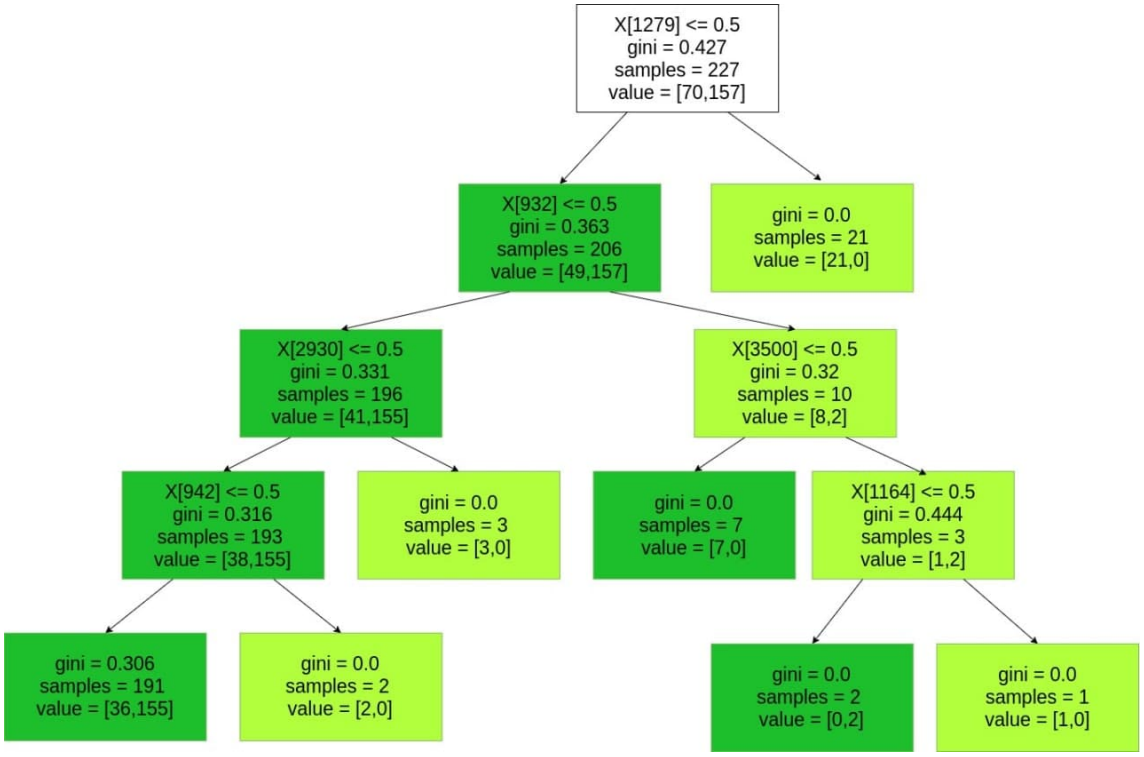


Figure 2-2: An example of a very simple decision tree for the binary classification of the developmental toxicity of molecules. $X[i]$ is a binary flag of the presence of certain functional groups in a molecule. This example was made on the real data, but only for the demonstration, and does not correspond to any model mentioned in this thesis.

information gain and Gini index [1]. For simplicity, let's discuss only the last one:

$$Gini = 1 - \sum_{i=1}^{N} (\Delta p_i)^2 \tag{2.6}$$

where Δp_i is the ratio of a class in a set. This index demonstrates the impurity of a set. Given a set with only one class (a perfect case), the index is 0. If the distribution over classes is equal (the worst case), the index has its highest possible value of 0.5 Given this metric, one can perform a search over possible splits to maximize the function:

$$\frac{1}{L} \sum_{i=1}^{L} (\Delta p_i^{left})^2 + \frac{1}{R} \sum_{i=1}^{R} (\Delta p_i^{right})^2 \longrightarrow max \tag{2.7}$$

[1] Despite the similar name, that is not the same Gini index that economists use for the inequality estimation

where R and L are numbers of instances in the left and the right nodes. It is known that regression trees tend to overfit. To combat this problem *pruning* is used. As we mentioned before, decision trees are not a production method, but it is a basis for many well-established algorithms. We will discuss two of them: Gradient boosting trees (on the example of XGBoost) and Random forest because we used these techniques extensively in our research.

Gradient boosting trees

Boosting is a meta-algorithm that demonstrates good performance in many different problems. The main idea of boosting is a composition of weak classifiers in such a manner that the next added classifier corrects the mistakes of previous classifiers (or follow the gradient of a loss function). Adaptive boosting (AdaBoost) was one of the first implementations of this idea Schapire [1999]. The main idea of AdaBoost is the increasing weights for the samples that were misclassified on the previous iterations. He et al. [2004] demonstrated that AdaBoost can be applied in chemistry for the classification of organic compounds by their biological and structural properties. Friedman [2002] introduced Stochastic Gradient Boosting as an universal framework for building boosting models over different types of classifiers and regressors, in particular decision trees. Svetnik et al. [2005] demonstrated that gradient boosting can be comparable or slightly outperform Random Forest for compounds classification problems and QSAR/QSPR modeling.

XGboost is one of the most prominent approaches in data mining Chen and Guestrin [2016]. This algorithm frequently becomes the leader in the Kaggle data science competition. It was shown that XGBoost could be very efficient for processing large chemical datasets in terms of accuracy and speed of computation Sheridan et al. [2016]. On each boosting iteration, a new decision tree is constructed to fit the residuals of the model obtained at the previous stage. It is worth to mention that XGboost can perform GPU calculations that enhance the performance on several orders of magnitude.

K-nearest neighbors

K-nearest neighbors, perhaps, is the simplest possible non-parametric method. Even more surprising is the unreasonable effectiveness of this method in QSAR/QSPR modeling Mitchell [2014]. In this method the prediction is calculated as a mean (or weighted sum) of N compounds that are the closest ones to the compound under investigation in some descriptor space. For the classification the example is classified in accordance with the dominant class among k-nearest neighbors. The idea is close to chemical paradigm that similar compounds have lookalike properties. This method is frequently used in chemical modeling especially for small datasets Kauffman and Jurs [2001], Gunturi et al. [2008].

Random Forest

This method uses the set (forest) of the simple classifiers or regressors, namely decision trees Breiman [2001]. This method has been heavily used in chemoinformatics for the last decade before the rise of deep learning because it has many advantages, particularly the performance of modelling, the speed of computation, and the ability to use default parameters or parameters with minimal tuning. It should be mentioned that this method has a long history of usage for toxicity prediction Cao et al.. Svetnik et al. [2003]

Artificial neural networks and Deep Learning

The history of Artificial neural network (ANN)s rooted in early attempts to emulate the activity of the animal's neural tissue. McCulloch and Pitts [1943] proposed a Heaviside step function as a mathematical model of a neuron and did a description of neural activity as logical computations. In their work, they theoretically demonstrated that nets which are based on these types of neurons can perform Boolean operations and can operate as a formal logic machine. Despite the naivety of the model, their theoretical findings played an important role in further AI researches. However, there was a gap between theory and implementation. Rosenblatt was the first researcher who closed this gap by implementing the first neural computer: *Mark*

I Perceptron. He designed this machine to demonstrate the feasibility of optical image recognition, and one can regard this computer as a precursor for further neural optical image recognition engines. The next milestone was the design of the backpropagation algorithm for the layered update of the coefficients of neural networks Rumelhart et al. [1986]. Cognitron and Neocognitron are unique neural architectures that have been designed by Fukushima, especially for optical pattern recognition. The perceptual elements of Neocognitron (S-cells) emulated the elements from the cat's receptive fields Fukushima [1980]. However, these architectures were hard to implement. Convolutional neural network (CNN) were proposed by Lecun et al. [1998] as a simple and efficient method for optical image recognition. The real power of CNNs has been revealed since 2011, within the era of deep learning.

Recurrent neural network (RNN)s are the unique neural architecture to process time series. However, this type of neural net had a major drawback: vanishing and exploding gradients. These problems restricted the abilities of RNN. To combat with these limitations Hochreiter and Schmidhuber [1997] proposed Long Short-Term Memory (LSTM). This model demonstrated excellent performance in many tasks: speech recognition, time series forecasting, text generation. But, as for CNNs, this architecture gained exceptional popularity only in the deep learning era. Before the last decade, Artificial neural networks' popularity was comparable to other machine learning methods. One could hope to obtain just slightly better quality but at a price of lower explainability and higher complexity. Choosing an architecture for an ANNs was a trickily process without any defined rules and was based mostly on intuition. All these issues restricted the applicability of ANNs in science and technology. The situation changed in last decade. First, GPU-based computations demonstrated the speedup of traditional backpropagation for two orders of magnitude. It motivated researchers to pay close attention to deep, multilayered artificial neural networks. The experiments have become feasible in terms of time. Second, the establishment of automatic differentiation frameworks: *Theano*, *Tensorflow*, *Chainer*, *Torch* and *PyTorch*, etc. Given automatic differentiation, one can describe only forward computations while gradients are calculated automatically. This feature allowed researchers to spend much less time on the designing of new

architectures. Interestingly, all ingredients were ready before. But the synergy effect of these techniques makes something that we know now as deep learning only in the last decade. Krizhevsky et al. [2012] proposed AlexNet network, and with this architecture, the team won the ImageNet 2012 challenge with a large margin. This paper was a milestone in deep learning history; One might even say that deep learning originated from this research. Since 2012 the solutions based on Deep neural networks became state-of-the-art in computer vision, speech recognition, natural language processing, and machine translation. The last notable success (from the author's point of view) was Transformer Vaswani et al. [2017]. Google designed it for neural translation, and this model demonstrated an overwhelming performance. Transformer was successfully used to predict the outcomes of chemical reactions in IBM's team research Schwaller et al. [2020]. Karpov et al. [2020] proposed Transformer-CNN – a hybrid model which calculates SMILES-embeddings from the encoder part of Transformer and uses these embeddings to build QSAR/QSPR models.

An artificial neural network is a function that translates objects from one representation to another. Or a function G that transfers the input object I into the output object O. It is worth mentioning that O can be of any nature. For example, Graph U-Net Gao and Ji [2019] maps graphs to graphs. Commonly, most neural networks operate with tensors; however, there is a branch of machine learning that studies ANNs on non-euclidean domains – geometric deep learning Bronstein et al. [2016]. In this case, an artificial neural network is a function that maps points from the input space to the output space. Still, the majority of neural networks operate in the traditional euclidean domain.

Deep ANNs commonly consists of several layers where each layer represents linear vector transformation

$$output_i = \sigma(W^T x + b) \tag{2.8}$$

where W – is a matrix of tunable weights, b – is a bias vector, followed by a non-linear transformation function σ (i.e. sigmoid, relu). The scheme of layers connections is a *computational graph* of a neural network.

Modern frameworks use the idea of automatic differentiation for the computation

of direct outputs and gradients with respect to each parameter of the network. The computational graph is a convenient theoretical framework and a practical tool in some deep learning frameworks, for example, in *Theano* and *Tensorflow*. The *computational graph* is a directed graph, where the nodes of the graph represent some computational units. These units can be *parametric* or *non-parametric*. There are two functions inside: *forward* and *backward*, where the first one describes how to calculate the unit's forward projection. The last one describes how to calculate gradient with respect to the parameters of the units. If all functions in this graph are continuous and differentiable, one can calculate a partial derivative of a final value with respect to any computational unit parameter. A deep learning framework calculates the derivatives using *chain rule*:

$$\begin{aligned} L &= f(g(x)) \\ \frac{dL}{dg} &= \frac{dL}{df}\frac{df}{dg} \end{aligned} \tag{2.9}$$

Given a loss function E, one can regard a problem of neural network training as the problem of minimization of the loss function.

$$E = \frac{1}{2}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{2.10}$$

where $\hat{y}_i$ – is a predicted value. To train our network, we need to have partial derivatives for each weight w_{ij} with respect to E. One can expect that the minimization of E on the training set can result in reasonably good predictions on the test set[2]. The minimization of E corresponds to the optimization of the weights of a neural network. One can do it by a gradient update:

$$\Delta w_i = -\gamma\frac{\partial E}{\partial w_i} \tag{2.11}$$

where γ is a *learning rate* – a small constant determining the size of a step in a counter-gradient way. In practice, for the optimization of neural networks, stochastic gradient descent is used, along with methods that regard the adaptive moment for

[2]And the whole building of machine learning is based on this simple assumption

the gradient, for example, Adam Kingma and Ba [2014]. Commonly, the training procedures use several techniques, such batch normalization Ioffe and Szegedy [2015] and dropout Srivastava et al. [2014], which help to achieve faster convergence and prevent overfitting. Deep neural networks are also a good choice for Multitask learning (MTL).

This was a brief historical background of ANNs and deep learning. To those who want to find out more about the history of artificial neural networks,We recommend an extended review, written by Jürgen Schmidhuber (Schmidhuber [2015]). It's worth mentioning a review prepared by Yann LeCun, Yoshua Bengio, and Geoffrey Hinton (LeCun et al. [2015]): three scientists who were among the founders of deep learning.

2.3 Models Validation and Performance Measurement

Bias-variance trade-off

In machine learning, the bias-variance trade-off is a problem of balancing between a bias of a model – the generalization ability, and a variance – the ability of the model to fit fluctuations in the data. We can illustrate this problem seeing bias-variance decomposition to estimate expected generalization error: let's assume that we have a dataset (X_i, y_i) where X_i – are input values, y_i – are corresponding output values. A function $y = F(x) + \epsilon$ where ϵ is the noise with zero mean and variance σ^2. $\hat{F}(x)$ – is our approximation of $F(x)$

$$\mathbb{E}_d[y - \hat{F}(x)] = (Bias[(y - \hat{F}(x)])^2 + Var[\hat{F}(x)] + \sigma^2 \tag{2.12}$$

The derivation of this equation can be found in Hastie et al. [2009]. There are three parts of the error: the bias, the variance, and the irreducible error σ^2. Because we can do nothing with the irreducible error, let's ignore it in further discussion. This form can be regarded as a lower bound of the expected error. So, one can see that the expected error for unseen examples consists of the bias term: that describes

"a price" of simplifying a model regarding the proper process, and the variance term: that indicates the deviation of a model around the mean. High bias low variance cases are associated with *underfitting* – a model is not trained enough (or has a small number of parameters). Conversely, when a model has low bias and high variance – it is *overfitting* – model predicts well for samples in the training set but fails on the test set. It is easy to demonstrate the bias-variance trade-off on the example of k-nearest neighbors regression:

$$\mathbb{E}_d[y - \hat{F}(x)] = (f(x) - \frac{1}{k}\sum f(N_i(x)))^2 + \frac{\sigma^2}{k} + \sigma^2 \tag{2.13}$$

Here, the bias increases with k but the variance decreases. Thus, one can not improve the quality of a model only by increasing the number of neighbors. It is a common problem for other machine learning algorithms too. Increase the number of parameters leads to a drop in performance due to overfitting. However, it does not mean that it is impossible to improve the quality of models. One can reduce the bias and variance simultaneously by regularization techniques, fitting the hyperparameters of an algorithm, using meta-algorithms on the top of ordinary ML algorithms. In Hastie et al. [2009] book there is a demonstration that overfitting leads to the overestimation of the quality of a model, which is evidently undesirable for production. Overfitting is a current problem in chemoinformatics. Chemistry is an experimental science and obtaining and collecting experimental data is an expensive procedure. That is the reason why the size of datasets in chemistry so far was remarkably low. It is easy to overestimate if the model is built on a small dataset and the number of parameters is somehow comparable to the size of the dataset. That is why the studies about the correct estimation of the performance of modeling get attention. It is worth to mention that one of the most cited paper in chemoinformatics ("beware of q^2" Golbraikh and Tropsha [2002] was cited more than 3000 times) is devoted to assessing model quality correctly.

Regression and Classification Model Accuracy Metrics

Typical metrics used for evaluating the performance of a regression model are Root Mean Square Error (RMSE):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{T}(\hat{y}_i - y_i)^2}{T}} \tag{2.14}$$

Mean Absolute Error (MAE):

$$MAE = \frac{\sum_{i=1}^{T}|\hat{y}_i - y_i|}{T} \tag{2.15}$$

and R^2:

$$R^2 = 1 - \frac{\sum_{i=1}^{T}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{T}(y_i - \overline{y})^2} \tag{2.16}$$

where $\hat{y}_i$ is a predicted value, y_i is a real value, $\overline{y}$ is a mean value over all samples, and T is the number of samples. R^2 – indicates the variance correctly described by a model. A simple explanation is how well the model relative to the most straightforward model – mean over y_i. Commonly, in chemoinformatics, researchers are used to q^2 – is a cross-validated R^2. Golbraikh and Tropsha [2002] reported that q^2 itself is a poor measure of the quality of models and additional conditions should be satisfied for honestly assessment of models. There are many ways of estimation of the performance of binary classifiers. The binary classification problem is common in medical diagnosis to estimate whether a person has a disease or not. There are four types of outcomes in binary classifying: True positive (TP) – the true samples that were correctly classified, True negative (TN) – the false samples that were correctly classified, False positive (FP) – the false samples that were incorrectly classified as true, and False negative (FN) – the true samples that were incorrectly classified as false. In statistics, there are two types of errors: type I and type II. Type I error is the rejection of a correct null hypothesis, and Type II error is the acceptance of a false null hypothesis. One can see that a type I error is equivalent to a false positive, and a type II error is equal to a false negative.

A number of classification metrics are used for the estimation of performance of

classifiers:

$$Accuracy\ (ACC) = \frac{TP + TN}{TP + TN + FP = FN} \tag{2.17}$$

$$Sensitivity\ (SE),\ recall,\ true\ positive\ rate\ (TPR) = \frac{TP}{TP + FN} \tag{2.18}$$

$$Specificity\ (SP),\ true\ negative\ rate\ (TNR) = \frac{TP}{TP + FN} \tag{2.19}$$

$$Precision\ (Pr) = \frac{TP}{TP + FP} \tag{2.20}$$

$$Balanced\ Accuracy\ (BA) = \frac{TPR + TNR}{2} \tag{2.21}$$

$$F_1\ score = \frac{2TP}{2TP + FP + FN} \tag{2.22}$$

Commonly, a binary classifier returns not a predicted class, but a probability distribution over two classes (True/False). The particular problem is to set a threshold. It depends on the technical requirements for the classifier and user's preferences. Fitting the cut-off value, one can increase one parameter, but decrease others. For example, in medicine, there is a sensitivity-specificity trade-off.

For the exploration of classifier behavior, one can use dependency curves. The Receiver operation curve (ROC) is one of the most used tools for this. Sensitivity (TPR) is plotted on the Y-axis and 1-specificity (FPR) on the X-axis. Given the ROC curve, one can set a cut-off value following the desired balance. Commonly Area under ROC curve (AUC ROC) is regarded as a measure of the performance of a binary classifier.

Despite the popularity of this metric, the usage of AUC ROC values for imbalanced datasets requites caution. In the prevalence of negative samples, AUC ROC can overestimate the performance of a classifier. Researchers proposed a number of other metrics: concentrated ROC (CROC) Swamidass et al. [2010], Cost Curves

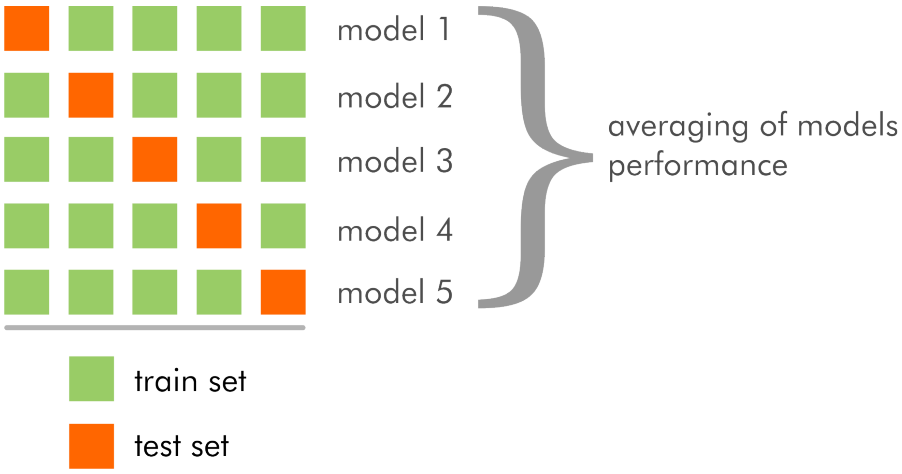


Figure 2-3: The scheme of *5-fold* cross-validation procedure. On each fold $\frac{4}{5}$ of a dataset becomes a training set and $\frac{1}{5}$ becomes a test set, sliding over folds. The cross-validation is done based on molecules and thus all toxicity values for the same molecules are within the same set always.

Drummond and Holte [2000], and Precision-Recall Curves. Saito and Rehmsmeier [2015] reviewed alternatives to ROC analysis and concluded that for imbalanced datasets Precision-Recall analysis is strongly recommended. Davis and Goadrich [2006] prepared a theoretical explanation to support Precision-Recall analysis over ROC analysis.

Cross-validation

Overfitting is a well-known problem resulting in inadequate performance estimations Golbraikh and Tropsha [2002] of ML models. Cross-validation routines are used to combat this problem and estimate the statistical performance of models in a robust way. A graphical explanation of a cross-validation procedure is given in Figure 2-3. A reader can find out the historical background of validation protocols used in QSAR/QSPR studies in the editorial note Gramatica and Sangion [2016].

A *5-fold* cross-validation has been carried out for all models in this thesis unless otherwise specified.

2.4 3D Reference Interaction Site Model (3D-RISM)

Calculation of an equilibrium distribution of solvent around an arbitrary molecule is a challenging problem in computational molecular science Ratkova et al. [2015]. It can be done by molecular dynamics simulations, but extremely long simulation times are needed to obtain smooth solvent distributions Luchko et al. [2010]. Theoretical methods of statistical mechanics like MDFT method can be applied to this problem as well as the 3D-RISM theory Chandler et al. [1986], Ratkova et al. [2015], Kovalenko and Hirata [1999, 2000b], Beglov and Roux [1997], Kovalenko [2003] which is used in this research.

As a result of the 3D-RISM calculation, one obtains *density distribution functions* (local densities) $\rho_\gamma(r)$ of every of the solvent sites γ around the solute molecule. These density distributions can be regarded as a variant of molecular fields. Notice that the densities obtained from RISM calculations are not exact Hirata [2003], Misin [2017], but can be successfully used to predict a variety of physical properties using either empirical or semi-empirical corrections Truchon et al. [2014], Misin et al. [2015, 2016a,b,a] or QSPR approaches Palmer et al. [2015].

The 3D-RISM main equation can be written as Hirata [2003], Misin [2017]:

$$h_\gamma(\boldsymbol{r}) = \sum_{\alpha=1}^{n_s} (\chi_{\alpha\gamma} * c_\alpha)(\boldsymbol{r}) \tag{2.23}$$

where $*$ denotes convolution, n_s stands for the number of solvent sites, $\rho_\gamma(\boldsymbol{r})$ is a density of site γ at point r, ρ_γ is a bulk density, and $h\gamma(\boldsymbol{r}) = \rho_\gamma(\boldsymbol{r})/\rho_\gamma - 1$ is usually referred to as the total correlation function. One should note that $\chi_{\gamma,\alpha}$ is obtained from a homogeneous solvent, while h_γ, c_α are solute-solvent correlation functions that describe a system with a fixed solute molecule, surrounded by solvent. $c(\boldsymbol{r})$ is a direct correlation function Kovalenko [2003]. Finally, $\chi_{\alpha\gamma}(r)$ is a site-site susceptibility function that can be obtained from a bulk solvent radial distribution functions. More conveniently, $\chi_{\alpha\gamma}$ can be calculated from a separate 1D-RISM calculation Perkyns and Pettitt [1992], Ratkova et al. [2015].

The above equation is coupled with a separate closure relation that provides

another connection between $h_\gamma(\boldsymbol{r})$ and $c_\alpha(\boldsymbol{r})$. One of the most popular and computationally robust is Kovalenko-Hirata (KH) Kovalenko and Hirata [2000a] closure:

$$h_\gamma(\boldsymbol{r}) + 1 = \begin{cases} \exp\left[-\beta u_\gamma(\boldsymbol{r}) + h_\gamma(\boldsymbol{r}) - c_\gamma(\boldsymbol{r})\right], & \text{if } h(\boldsymbol{r}) \leq 0 \\ 1 - \beta u_\gamma(\boldsymbol{r}) + h_\gamma(\boldsymbol{r}) - c_\gamma(\boldsymbol{r}), & \text{if } h(\boldsymbol{r}) > 0 \end{cases} \tag{2.24}$$

where $\beta = 1/(kT)$ and $u_\gamma(\boldsymbol{r})$ is a potential energy between the solvent site γ and the solute molecule. Together the above systems of equations are usually iteratively solved until both $h_\gamma(\boldsymbol{r})$ and $c_\alpha(\boldsymbol{r})$ achieve a predefined convergence criteria.

2.5 Multitask Learning for Chemical Data Analysis

In the modern era, the number of chemical data generated is increasing exponentially Tetko et al. [2016b]. New data requires new techniques for processing. Machine learning and especially deep learning play an important role in handling the chemical and biological data, providing predictive models for various properties. Even more, several years ago, there was a question whether "Big Data" really exists in chemistry Tetko et al. [2016a]. Nowadays, there is a consensus opinion about the necessity of new methods for processing such amounts of data. However, data management is a difficult task, requires much time and effort, and one should find a way of the best usage of all available data. But it is known that biological information is often interrelated, which can be used to increase the quality of modeling. For example, there are known empirical relations between the melting point of an organic compound and the logarithm of solubility of the compound in water Ran and Yalkowsky [2001]. Thus learning several ADME-Tox properties together can result in better models. The measurement's costs for different properties vary notably. For example, kinetic water solubility, which is the concentration of a compound in solution when an induced precipitate first appears, can be measured in High Throughput Screening (HTS) settings. Combining much “cheap” data with a few “expensive” ones is a promising way for improving the quality of multitask models.

These multi-learning approaches belong to so-called transfer learning, Pan and

Yang [2009] a technique where knowledge gained in one or several (source) tasks is used to improve the target task. The transfer learning approaches differ with respect to whether the source and/or target tasks have labelled data. Thus, they can be classified as semi-supervised or "self-taught" learning (no labelled data in the source domain), transductive learning (labelled data are only in the source domain), unsupervised transfer learning (no labelled data are available) Pan and Yang [2009] as well as methods which use labelled data for both source and target tasks, which include multi-learning approaches.

The ability to infer relevant knowledge is very important for intelligence. For example, humans, who can draw on vast amounts of previously-learned information, can be trained on a new task with a relatively tiny number of examples. In contrast, traditional machine learning algorithms, which usually learn from scratch, and require large example sets to do so. Therefore, there is active development and interest in machine learning to design new methods having the same speed and accuracy as humans. Early examples of such types of learning have been successfully reported since the mid-1990s, e. g. the use of neural network weights trained with one task as a starting point for new ones to increase the development speed and the accuracy of models Caruana [1998]. Associative Neural Networks Tetko [2008] are another example, which applied on-the fly correction of predictions for new data by using the errors of the nearest neighbours of the target sample Tetko and Poda [2004]. Transfer of information was also done by developing models for individual properties, and then using those model predictions as additional descriptors for the target property, known as the feature net approach Varnek et al. [2009]. In the case that the target and source properties are very similar or identical (e. g., measured for different species or at different conditions), one can encode different targets by using additional descriptors (e. g., conditions of experiments) and model all properties simultaneously.

Multitask learning (MTL) is a technique which aims to improve Machine learning (ML) efficacy by simultaneously co-modelling multiple properties within a single model. A lot of developments in this field were done in in 1990s by Rich Caruana [1998] who investigated how to improve related task performance by leveraging

domain-specific information, and inductively transferring it between the tasks. In comparison to the other transfer learning approaches, which use labelled data for both source and target tasks, the aim of MTL is to improve the performance of all tasks with no task prioritised.

MTL trains tasks in parallel, sharing their representation internally. As a result, the training data from the extra tasks serve as an inductive bias, acting in effect as constraints for the others, improving general accuracy and the speed of learning. Caruana [1998] noted how MTL may show improvement over Single-task learning (STL):

- amplification of statistical data;
- attention focusing (finding a better signal in noisy data)
- eavesdropping (learning "hints" from simpler tasks)
- representation bias and feature selection
- regularisation (less overfitting)

As MTL implies sharing information between all tasks, it is possible to define three main types of MTL based on the type of data sharing: feature, instance and parameter-based. Zhang and Yang [2017] Feature-based MTL models learn a common feature representation among all the tasks by assuming that such a representation can increase the performance of the algorithm vs. single-tasks. Parameter-based approaches explore the similarity between target properties and include task clustering, learning of task relationships, as well as multilevel hierarchical approaches.

Feature Based Approaches

Neural networks are the primary platform for multi-learning. Rich Caruana was one of the first to develop multi-task learning using backpropagated neural networks. He found that four separate neural networks performing only one task can be reduced to one network with multiple outputs that performs the tasks simultaneously. As a result, he created a multi-task neural network to perform parallel learning. One

should also mention the earlier work of Suddarth and Kergosien Suddarth and Kergosien [1990] who used an additional layer to inject rule hints and to guide the neural network as to what should be learned.

The network forms a set of features on the hidden layer(s), which can fit several tasks simultaneously. Moreover, the activation patterns of neurons in neural networks with several hidden layers contribute to the formation of features, which are known to be important for the analysed type of properties, e.g. toxicophores for the prediction of toxicological end-points Mayr et al. [2016].

One of the first successful applications of MTL in chemoinformatics was done by Varnek et al. [2009] who demonstrated that learning several tissue/air partitioning coefficients by using Associative Neural Networks provided models with statistically-significantly higher accuracy compared to the respective single task models. The neural network models analysed by this team were examples of so-called "shallow" neural networks since they included only one hidden layer. The appearance of new training algorithms and in particular GPU-accelerated computing has brought the renaissance of Deep Neural Networks Baskin et al. [2016] which incorporate multiple hidden layers with much larger numbers of neurons. This greater flexibility of DNN networks allows them to learn more complex relationships and patterns in the data.

Regarding multi-learning one can distinguish two primary architectures with respect to the sharing of parameters: hard and soft. "Hard" parameter sharing is similar to that of shallow neural networks and implies the sharing of hidden layers between all tasks, except some task-specific output layers. "Soft" parameter sharing gives each task its own model with its own parameters, where these model parameters have a regularized distance to facilitate the sharing of learning Ruder [2017].

Ma et al. [2015a] performed several experiments on STL and MTL neural networks. They found out that in some cases multi-task learning deep neural networks (MTL DNNs) are better than single task learning deep neural networks (STL DNNs). The authors suggested that better performance of MTL is based mainly on the size of data sets: STLs are useful for small and mixed (small and large) datasets and. MTLs are good for large data sets.

MTL provided the best model according to the ROC AUC metric for the Tox21 challenge Mayr et al. [2016]. The authors showed that such networks learned on their hidden layers chemical features resembling toxicophores identified by human experts. The networks used these features to classify active and inactive (toxic and nontoxic) compounds. It is also of note that the second best approach was based on "shallow" STL associative neural networks Abdelaziz et al. [2016].

Xu et al. [2017b] investigated why an MTL DNN can outperform separate STL DNNs and under what scenarios the multi-task approach is advantageous. The result of this study lead to two main findings regarding the efficacy of multi-task deep neural networks:

- Similar molecules may have correlated properties which will boost the predictive performance of the DNN, and likewise uncorrelated properties will degrade performance.
- Structurally dissimilar molecules have no influence on the predictive performance of the MTL DNN, regardless of whether or not tasks are correlated.

Their conclusions are important for strategies for designing datasets for MTL learning.

MTL can be used to simultaneously learn both regression and classification in one model, as was demonstrated by Xu et al. [2017a] for the prediction of acute oral toxicity. The authors used convolutional neural networks and reported that their model provided higher accuracy compared to conventional methods.

Human cytochrome P450 inhibition for 5 kinases were predicted using a pre-trained autoencoder-based DNN Li et al. [2018]. On the pre-training stage, the first layers were trained to reconstruct the original input layer on the whole database. The authors proved that an autoencoder-based DNN can achieve better quality than other popular methods of machine learning for cytochrome P450 inhibition prediction, and a multi-target DNN approach can significantly outperform single-target DNNs. The flexibility of neural networks allows to use them not only with descriptors derived from chemical structures in the traditional way, but to directly

analyse chemical structures represented as SMILES or chemical graphs Karpov et al. [2020].

Other approaches to Multi-task feature learning

The problem of feature selection has an exact mathematical formulation and an analytical solution for linear methods. For example, Varnek et al. [2009] compared the performance of neural networks with Partial least squares (PLS). PLS could also provide multi-task learning by identifying common internal representations, so called latent variables, for several analysed properties simultaneously. In addition to the PLS method, there are other approaches for identifying sparse features or to perform multi-feature selection as comprehensively analysed in a recent review Zhang and Yang [2017]. These methods can be used directly with linear or kernel methods, or to provide features for training other methods.

One such method is Macau Simm et al. [2015]. It is based on Bayesian Probabilistic Matrix Factorisation (BPMF). BPMF was used to win the Netflix prize for predicting film recommendation, the interest in this method notably increased. One of the problems during multi-learning are missing values; frequently not all measurements are available for all targets. For some other tasks the matrix of responses can be extremely sparse, for example only 1.2% of all users-combinations were available for the Netflix competition. Some methods, such as neural networks, can naturally work with missing values by ignoring the error contribution from missing values when calculating the loss for backpropagation. The BPMF allows imputing missing values in the matrix thus enabling the application of standard techniques, such as singular value decomposition and principal component analysis. In contrast to classical algorithms of matrix factorization, Macau is able to handle side relations i.e. fingerprints of chemical compounds or phylogenetic distance between protein targets. Another useful feature of Macau is the ability to work with multi-dimensional data and perform tensor decomposition. The capacity to deal with multi-dimensional biological sparse data was studied by de la Vega de León et al. [2018] who applied this technique to inhibition activities of 15073 compounds for 346 targets extracted from ChEMBL. The authors showed that Macau provided performance similar to

that of neural networks methods but did not require GPU-accelerated computing.

Chapter 3

A hybrid 3D method for the prediction of bioconcentration of organic compounds

As we mentioned in the previous chapter, there is an ultimate interest in developing new 3D based descriptors. These descriptors, on the one hand, should be accurate enough to provide the possibility for building the predictive models, and on the other hand, should be universal to cover more extensive parts of chemical space. One possible way is to use molecular theories of liquids for the generation of numerical description of solvation, and the usage of this description as 3D molecular descriptors.

Molecular theories such as three-dimensional reference interaction site model (3D-RISM) Beglov and Roux [1997], Hirata [2003], Ratkova et al. [2015], ER-theory Matubayasi and Nakahara [2000] or Molecular density functional theory (MDFT) Jeanmairet et al. [2013], Ramirez and Borgis [2005], Gendre et al. [2009] rely on approximations derived from rigorous statistical mechanics to estimate the equilibrium distribution of solvent around solvated molecules. In turn, these distributions can be related to many physical-chemical properties of a solvated molecular system Hansen and McDonald [2013], Ben-Naim [2006]. Examples of such properties include solvation free energy Du et al. [2000], Palmer et al. [2010], Misin et al. [2015], partial molar volume Ratkova et al. [2015], Misin [2017], salting-out constants Misin

et al. [2016b] and binding free energies Genheden et al. [2010], Güssregen et al. [2017], Sugita and Hirata [2016]. However, using a purely theoretical approach, it is difficult to relate these distributions to the substance's biological effects which are a result of a large number of complex interrelated phenomena, such as toxicity or bioaccumulation.

The above does not mean that the solvation structure is not useful for the understanding of the influence of chemical compounds on living organisms. On the contrary, the information encoded in the solvation shell can be used to understand whether a given compound is hydrophobic or hydrophilic Lum et al. [1999] which in turn can provide a reasonable guess whether it will be able to pass certain membrane channels Roux and Karplus [1991]. In the case of a solution that contains ions, the solvation structure can provide an estimation for the solute affinity towards them Misin et al. [2016b]. All this information is directly *related* to the compound's biological effects but can not be expressed *explicitly* using equations. On the other hand, machine learning methods are usually quite good at finding and quantifying such 'hidden' relations Myint et al. [2012], Ajmani and Viswanadhan [2013], Ma et al. [2015b].

We utilize a 3D Convolutional neural networks to develop a prediction model that can estimate the bioaccumulation propensity of a compound characterized by the Bioconcentration factor (BCF) for a number of different organic molecules. As an input, we use three-dimensional distributions of water around these molecules, obtained by 3D-RISM with Kovalenko-Hirata closure (KH) Kovalenko and Hirata [1999]. Artificial neural network (ANN)s have been previously used for predicting the biological effects of organic molecules Myint et al. [2012], Ajmani and Viswanadhan [2013], Ma et al. [2015b]. However, they were combined with a very broad set of descriptors that have diverse physical meanings. Here we focus on a single descriptor; solvation shell structure in an attempt to show that this can be a universal descriptor for the prediction of properties of molecules that are difficult to formalize by a theory. To determine whether the CNN-based machine learning setup is necessary, we also tested linear and Extreme Gradient Boosting (XGBoost) models and compared them with the 3D CNN approach.

3.1 Materials and Methods

Bioconcentration factor

This factor is the ratio between the concentration of an organic compound in biota and in water: Arnot and Gobas [2003]

$$BCF = \frac{C_{biota}(\text{compound})}{C_{water}(\text{compound})}, \tag{3.1}$$

This factor is an important parameter for estimating the potential danger of an organic compound. It is one of the parameters that determine the labeling of the compound under Registration, Evaluation, Authorisation and Restriction of Chemicals (REACH) program. The ability of a compound to penetrate and remain in an organism may influence the toxicity and mutagenicity, and so may reveal potential environmental risks. Generally, if a chemical has BCF value of more than 5000 (or $\log_{10}$ BCF $>$ 3.67), it is regarded as potentially dangerous. There are several methods to measure and estimate the confidence of the BCF data, described in details in Arnot and Gobas [2006]. It should be emphasized that determining of BCF in in-vivo experiments is a very expensive procedure.

where C represents concentration. It should be noted that BCF is regarded as a consolidated property of a chemical compound; thus, the definition involves some common concepts like "biota" and "stationary concentration in vivo". However, there is OECD 305 guideline OECD [2012], which provides the basic requirements for the methods that should be used for Bioconcentration factor estimation to obtain high quality and comparable data. The typical way to estimate BCF is a measurement of the concentration of a compound in fishes and water after reaching of stationarity of concentrations, usually by exposing the chemical during the pre-defined long period. Strictly speaking, there are many types of BCF which definitions depends on the concentrations of compounds, species of animals, times of expositions, and other factors of the experiments. However, OECD 305 guideline (Bioaccumulation in Fish: Aqueous and Dietary Exposure) allows comparing measurements even for different fish spices under certain circumstances.

Over the years, several models for the BCF prediction have been published. Arnot and Gobas have proposed a linear model that predicted BCF as a function of the uptake and elimination of an organic compound by an aquatic organism. Since BCF is related to logP and water solubility Arnot and Gobas [2006], some authors proposed models that utilised these descriptors Papa et al. [2007]. These linear models work satisfactory only for moderately hydrophobic compounds, but fail to address strongly hydrophobic chemicals Gramatica and Papa [2005]. Additionally, *LogP* and solubility must be measured separately and this may be problematic. Another notable model has been produced by Zhao et al. [2008] using a hybrid of a number of machine learning methods. Their model managed to produce an impressive accuracy ($R^2 = 0.8, \mathrm{RMSE} = 0.59$), albeit on a somewhat curated dataset.

We used the dataset collected by USA Environmental Protection Agency (EPA) for their Toxicity Estimation Software Tool (TEST) QSPR platform for risk estimation EPA [2016]. US EPA collected the database from several sources Dimitrov et al. [2005], Database, Zhao et al. [2008]. This dataset contains BCF measured values for several fish species: european carp and salmonids. As it has been discussed above combination of BCF values from cross-species experiments is allowable. We did not do any changes (modifications, additions, filtration) in the dataset. This dataset has been split into training and test subsets in the same manner as it was done by US EPA, and statistical values on the test set are published. We used them as a baseline for our model. There are 541 molecules in the training set and 135 molecules in the test set. We used RDKit to perform basic molecular routines and to estimate the geometries of molecules.

Conformers Generations: We regard conformers generation as a part of the data augmentation process. For that point, we need conformers that, on the one hand, have low energy (to assure physical meaning for such conformations); on the other hand, these conformers should be diverse to provide more information for our 3D CNN. For deep neural networks, a high amount of diverse data is a key factor to success. Our approach to conformer generation and selection is similar to the article Jean-Paul et al. [2012] and is briefly described below. At the first stage of the algorithm, we generate a number of conformers by rotating a molecule's bonds

in a stochastic manner. This is followed by an energy minimization step, consisting of 5000 iterations and performed using the universal force field (UFF) Rappe et al. [1992]. Because after the optimization, some molecules fall into very similar conformations, we have somehow to restrict the number of "valuable" conformers. To achieve that, we do Butina clustering. Then, on each iteration, we select one compound from each cluster until we reach the number of conformers we need. But when we select a candidate, we calculate a set of RMSD values overall non-hydrogen atoms (heavy atoms) between the candidate and all conformers we have already chosen. If there is a low RMSD value, we exclude this candidate and continue iteration. This procedure prevents the final conformers from being similar to each other. Although we use RDKit machinery for the conformers generation, the process of conformer selection is not a part of the standard RDKit pipeline. Our experiments, and experiments of other researchers Hemmerich et al. [2020], demonstrated that the number of conformers directly affects the performance of modeling; however, we believe that similar conformers bring no useful information for the neural network. We note that the prediction output for every molecule is an average over the whole ensemble of corresponding conformers. We believe that this procedure can also mitigate potential issues related to rotations of molecules.

3D-RISM Calculations: We used AmberTools16 Case et al. [2016] package to calculate the partial charges of each molecule using AM1-BCC Jakalian et al. [2002] semi-empirical model. At this stage, for some molecules the calculations have not converged, and these molecules were eliminated. These partial charges were used for further 3D-RISM calculations. All 3D-RISM computations were performed using *rism3d.snglpnt* program Kovalenko and Hirata [1999, 2000b], Kovalenko [2003], Luchko et al. [2010] from AmberTools16 Case et al. [2016] package. Site-site susceptibility functions of bulk water $\chi_{\alpha\gamma}(\boldsymbol{r})$ were calculated using DRISM method by *drism* program from the same package. The water temperature was set to 298 K. For 3D-RISM we used a 35 Å × 35 Å × 35 Å grid with 0.5 Å step size. The resulting oxygen and hydrogen density distributions were saved as HDF5 binary files. We ran a separate 3D-RISM calculation for each conformer. If more than 50% of 3D-RISM calculations did not converge, such molecule was eliminated from the dataset.

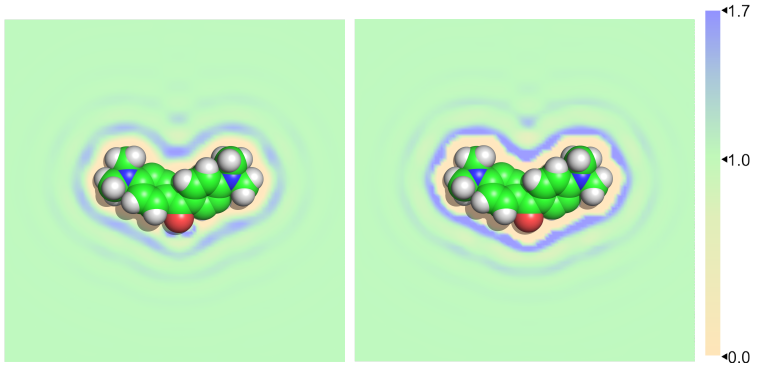


Figure 3-1: An example of the visualization of the scalar fields for a molecule as 2D slices taken by the principal axis (Left – a visualization of hydrogens density. Right – a visualization of oxygen density. Light yellow color – lower values, pale green color – bulk values, blue color – higher values))

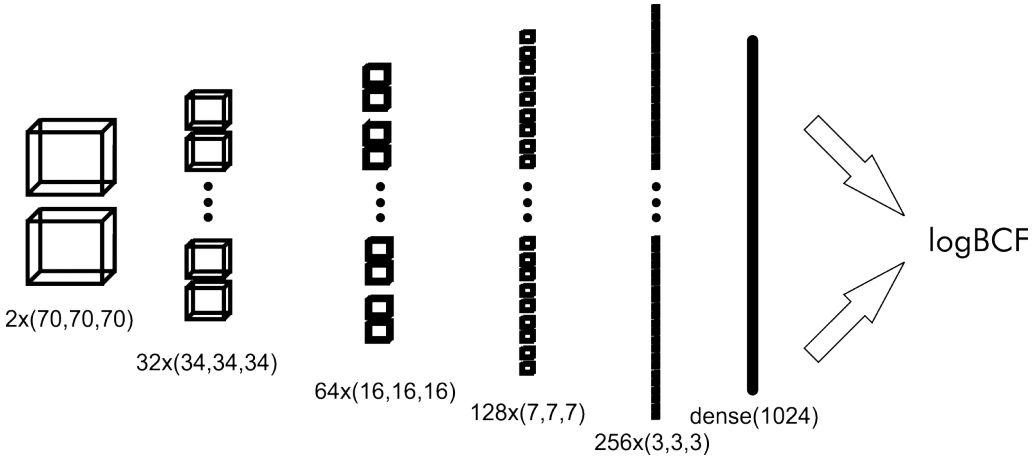


Figure 3-2: A general representation of *ActiveNet4* 3D Convolutional ANN

3D Convolutional Neural Networks Modeling Procedure: We used framework Chainer Tokui et al. [2015a] to build networks for processing 3D data. The architecture of the network is schematically presented in Supplementary Information in Figure 3-3. This architecture was optimal in terms of speed and the quality of the training models. This model has been called *ActivNet4*, with four indicating the number of convolutional layers used. A *pooling* layer is introduced in the structure of the CNNs which reduces to a minimum the potential effects of translation, rotation and shifting of molecules on the final output of the algorithm. We trained this network using both oxygen and hydrogen density distributions, obtained from 3D-RISM calculations.

Parametric Rectified Linear Units He et al. [2015] were used as activation func-

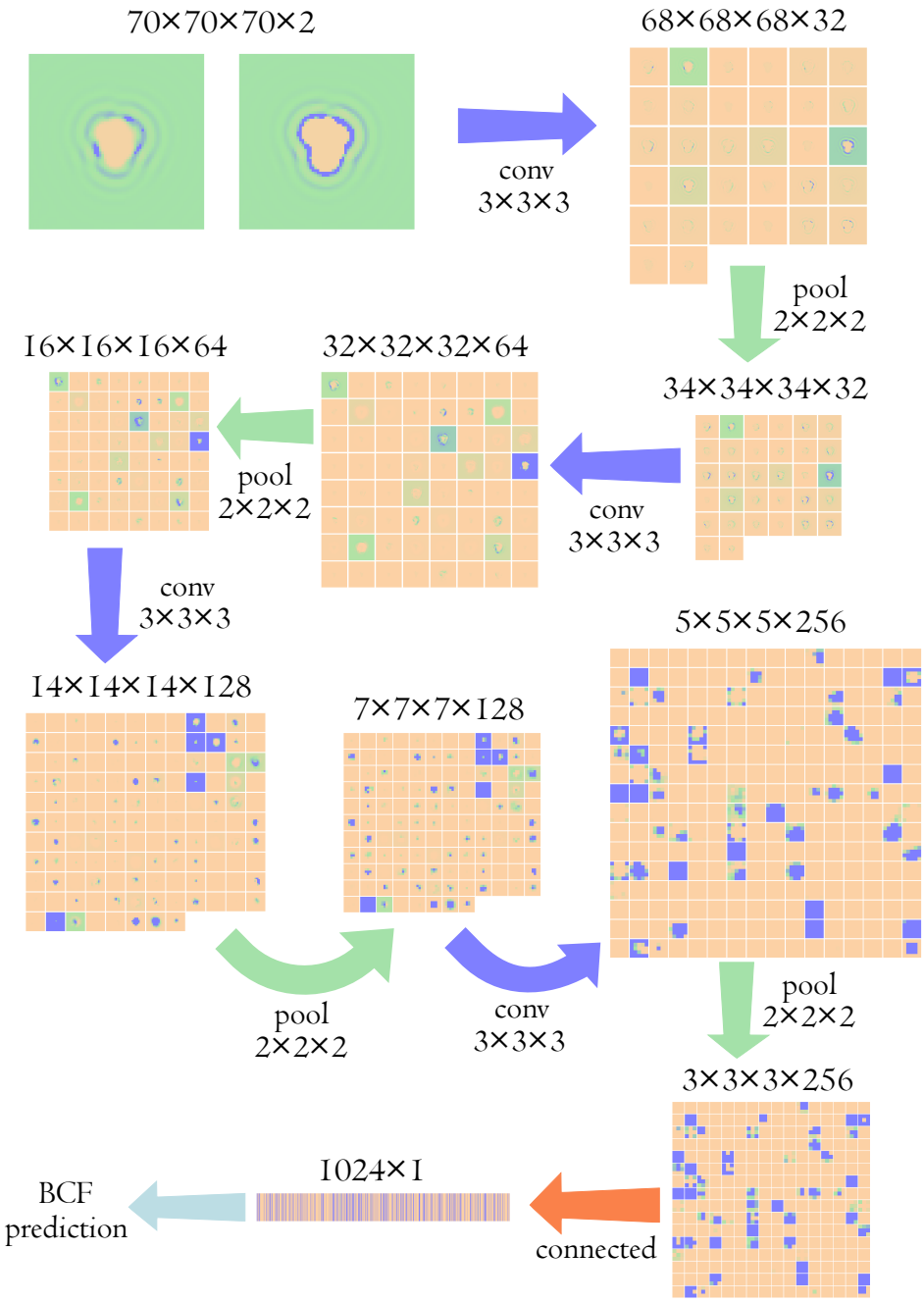


Figure 3-3: A schematic representation of *ActivNet4* architecture with visualized 2D slices of feature maps on a trained network. Feature maps are colored using the same color scheme as in Figure 3-1. Blue arrows labeled conv N × N × N denote a 3D convolution layer, green arrows labeled pool N × N × N denote 3D max-pulling layer, and red arrow labeled "connected" denotes a fully-connected layer. The figure is based on Figure 4 from Ref. 59

tions for the model since they showed small improvement in the prediction quality, although, it is possible to replace them with the commonly used ReLu activation function without a noticeable lack of performance. To train *ActivNet4*, we experimented with several optimizers: Stochastic gradient descent with momentum, Adam Kingma and Ba [2014], RMSprop Tieleman and Hinton [2012], and SMORMS3 Funk [2015]. The best and the most stable convergence has been provided by SMORMS3 method. RMSprop and Adam have a good convergence ability, but the training process was less stable. Stochastic gradient descent has converged noticeably more slowly for the network. To train our networks we used the parameters of optimizers: for Adam optimizer $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ $\epsilon = e^{-8}$. For RMSProp we used learning rate = 0.01, $\alpha = 0.001$, $\epsilon = e^{-8}$. For Stochastic gradient descent we used learning rate = 0.01, momentum = 0.9. All other parameters were set to default. The training and test procedures slightly differed. At the training stage, each conformer of the molecule has been regarded independently from the other conformers. At the test stage, the prediction value for each conformer of the molecule has been calculated and the final result was the mean value for all conformers of the molecule. The performance of the model was estimated on the same test set that has been used in the original work to compare our model with the baseline. Additionally, we used a 5-fold cross-validation technique for the whole dataset to measure the quality of the model in a more reliable way. The Neural networks have been trained using Nvidia K80 graphics cards and Nvidia GTX 1080 cards. Training of one model requires approximately 5 hours on Nvidia GTX 1080 and up to 4 times longer on Nvidia K80 graphics cards.

Extreme Gradient Boosting modeling (3D Fields): To compare our 3D convolutional network with other machine learning approaches we built a model using Extreme Gradient Boosting (XGBoost implementation Chen and Guestrin [2016]) algorithm. This method has been proposed for use in cheminformatics Sheridan et al. [2016] and can process very large datasets. In this experiment, initially, we had to decrease the volume of each 3D cube from *70x70x70* to *17x17x17* by performing the average pooling operation with a kernel *(4,4,4)*. Then, both oxygen and hydrogen channels have been flattened and stacked forming a vector of *9826* values.

These vectors served as the inputs for XGBoost algorithm. The application of the method to the test set has been performed in the same manner as in the neural network experiment. We used the maximal number of trees = 100 and maximal depth of each tree = 6 to train the models, the other parameters have been set to default.

Graph Convolution modeling: It was shown recently that in some cases graph convolution methods can overperform traditional QSAR/QSPR approaches which are based on the molecular descriptors Kearnes et al. [2016], Duvenaud et al. [2015]. We used DeepChem Ramsundar et al. [2019] framework included in Online Chemical Modeling Environment Sushko et al. [2011] to build graph convolutions models. For graph convolution model we used the hyperparamethers: epochs 100, learning rate 0.001, dropout rate 0.25, dense layer size 128 neurons, the size of convolution layers was (64,64) the other parameters have been set to default.

Linear model: Finally, we also built a linear model for BCF prediction using the following relation:

$$\log_{10} \mathrm{BCF} = a_1 \Delta G + a_2 \bar{V} + a_3, \tag{3.2}$$

where ΔG is molecule's hydration free energy, obtained with 3D-RISM PC+ method Sergiievskyi et al. [2015], Misin [2017], $\bar{V}$ is partial molar volume, and a_i are parameters adjusted in the process of regression. The optimal results were obtained with $a_1 = 0.10634 \frac{mol}{kkal}$, $a_2 = 0.003\,57\,\text{Å}^{-3}$, and $a_3 = 1.64677$.

3.2 Results and Discussion

Our main goal was to predict biological property using a combination of solvation structure and machine learning. For this, we used EPA database which has 676 molecules with known BCF. 670 molecules were successfully processed, while 6 molecules failed at the partial charges calculations stage or at the 3D RISM stage. The database were split into a training (537 molecules) and test (133 molecules) sets. For each molecule we then generated a diverse set of conformers, using a procedure described earlier. The distribution of a number of conformers for both training and test sets is shown in Figure 3-4. As one can see, about a quarter of the molecules in

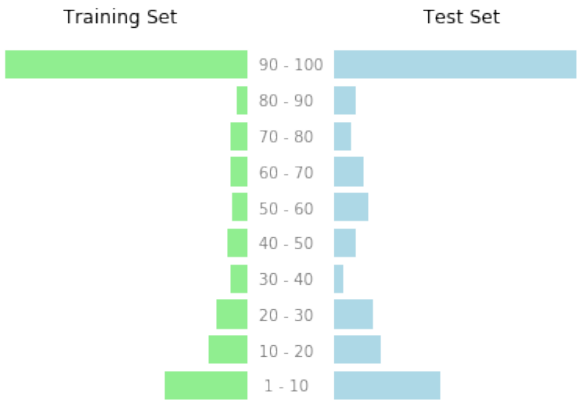


Figure 3-4: The distributions of the number of conformers for each molecule in the training and test sets

the training and test sets have less than 10 conformers (quite inflexible), while the remaining molecules are highly flexible with 90-100 conformers. The distribution of the conformers is similar in both sets.

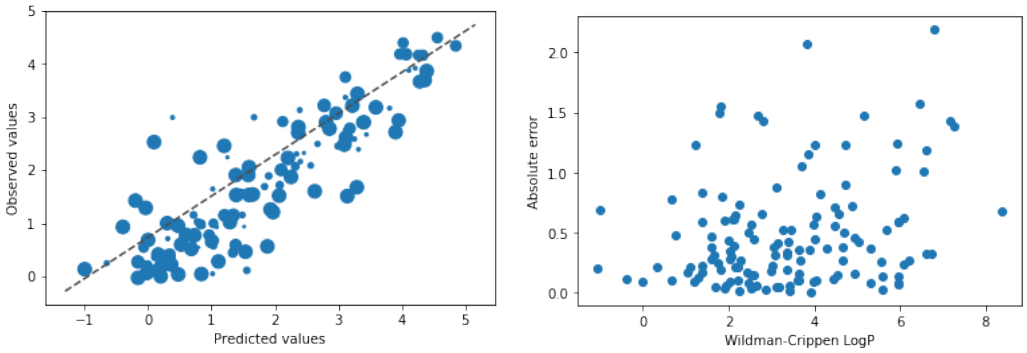
The main results are summarized in Table 3.1. ActivNet4 model is achieving accuracy comparable to the "consensus" model provided by the US EPA EPA [2016]. This result is noteworthy due to the fact that our model was based only on the 3D distribution of water molecules while the EPA's models used a large set of different descriptors . The comparison of the two models demonstrates that the analysis of the solvent density distribution using neural networks may be useful for predicting biological properties. Surprisingly, graph convolution model showed notably worse result than baseline model, this effect can be explained by the relatively small dataset.

To validate the necessity of using 3D convolutional neural networks we created Extreme Gradient Boosting (XGBoost) and linear models on the basis of 3D-RISM results. Both alternatives demonstrated poorer accuracy compared to the original method, highlighting that deep learning is more appropriate to achieve accurate results.

To explore the correlation between the hydrophobicity of compounds and the absolute error of predictions we calculated Wildman-Crippen LogP (n-octanol/water) values for the test set. The results is presented on Figure 3-5b. One can see that there is no strict correlation between these factors. Our approach was designed to

Table 3.1: Accuracies of $\log_{10}$ BCF predictions by different models. RMSE stands for root mean square error, MAE stands for mean absolute error and R denotes Pearson's correlation coefficient. For cross-validadated models the standard deviations have been calculated.

Model		**RMSE**	**MAE**	**R^2**
US EPA (baseline)	consensus model	0.66	0.51	0.76
	single model	0.68	0.64	0.74
ActivNet4 (3D data)	training/test	0.66	**0.48**	**0.77**
	5-fold CV	0.65 ±0.04	0.48 ±0.01	0.77 ±0.03
XGBoost (3D data)	training/test	0.85	0.70	0.61
	5-fold CV	0.91 ±0.02	0.72 ±0.02	0.54 ±0.04
Graph Convolution	training/test	0.85	0.67	0.61
	5-fold CV	0.84 ±0.03	0.66 ±0.02	0.62 ±0.02
Linear Regression (ΔG and $\bar{V}$)	training/test	1.11	0.92	0.32



(a) Correlation between observed and predicted values of $\log_{10}$ BCF. The size of the marker depends on the number of conformers of the molecule.

(b) The correlation between the Wildman-Crippen LogP and the absolute error of predictions

provide a full computational pipeline from only structural information to BCF predictions. Despite we use the 3D fields that have physical meaning, we do not believe that the usage of experimental data is feasible. First, this data is sparse and hard to standardise to provide a bias for joining values of different experiments. Second, experimental methods for the determination of solvation are expensive, and if one has the compounds physically the direct measurement of BCF is preferable.

3.3 Conclusions

The aim of this part of my research was to predict the bioaccumulation factor using an approximate solvent density obtained using 3D-RISM method of integral equation theories. After training, the *ActivNet4* (4-layer convolutional neural network) predicted $\log_{10}$BCF from water density distribution with RMSE=0.66. We demonstrate that average solvent distribution in the neighbourhood of solutes can be combined with machine learning algorithms to predict biological properties. Although the model used relatively simple 3D descriptors, it was enough to achieve prediction accuracies comparable to the state of the art models.

Chapter 4

Multitask learning for acute toxicity modelling

Toxicity is defined as the potential for a chemical compound to cause injury Katzung and Trevor [2014]. Accurate prediction of toxicity of organic compounds is one of the most challenging tasks in medicinal chemistry and pharmacology. According to a study Wong et al. [2018], nearly 30% of drug candidates fail in the first stage of clinical trials due to a presence of non-desired side effects, which results in a cost increase for pharmaceutical industry. This fact emphasizes that current methods for 'in-silico' toxicity estimation have serious shortcomings and that development of the new methods is of the utmost interest.

Toxicity estimation can be performed in two main ways: *in-vivo* using rodent models and *in-vitro* using cell-based bioassays. The former approach allows for the estimation of the toxic effect, at organism level, producing comprehensive results, and is widely used in preclinical tests. It should be noted that rodent models are not fully representative of humans and their use can thus result in unexpected side effects, which can be observed during clinical trials or even after drug approval Alden et al. [2011]. The fact that *in-vitro* tests are relatively inexpensive facilitates automation and makes their use possible in high-throughput screening (HTS) Inglese et al. [2006]. The different types of toxicity mechanisms can be detected by using

different assay types. Currently, there is great demand for development of new reliable relevant assays for, e.g., nephrotoxicity. Huang et al. [2014] However, the *in vitro* tests do not also always consistent with the *in vivo* toxicity because human cell-based data used in *in vivo* testing may not take into account the general systemic toxicity for the whole organism. At the same time, the in-vivo based rodent models do not always correctly represent human toxicity. Thus there is a great interest in the development of computational techniques to reliably predict toxicity Thomas et al. [2018] .

Currently, a large amount of information has been accumulated and kept in commercial and open source databases. Some examples of the open source databases are the TOXNET database Institute of Medicine (US) Committee on Internet Access to the National Library of Medicine's Toxicology and Environmental Health Databases [1998] and DSSTox, Richard and Williams [2002] which includes Tox21 high throughput data and ChEMBL Bento et al. [2014] database containing approximately 15 million of bioactivities. Among the proprietary databases, the Registry of Toxic Effects of Chemical Substances (RTECS) database is the most valuable, and it contains information about 187 000 chemical substances. It has *in-vivo* data for acute toxicity, skin irritation, tumorogenic properties and other effects measured for different organisms such as rodents, rabbits, and many others.

The open access to bioactivity data in these and other databases prompted the development of high quality prognostic models created using various machine learning methods. For example, the PASS software (and web-service) Pogodin et al. [2015] based on the Naive Bayes approach and trained using ChEMBL, demonstrates good reliability when performing the classification task on a set of more than 2500 protein targets. The EMolTox web-service Ji et al. predicts different types of toxicity using random forests and conformational prediction as measure of confidence and simultaneously visualizes the ToxAlert substructures on the molecular graph. The ProTox web-server is another tool for prediction of acute toxicity and other types of toxicity Drwal et al. [2014], which utilizes a nearest neighbor approach combined with fingerprint similarity assessment. There is a number of models constructed for a narrow class of chemical compound Asadollahi-Baboli [2012], Auerbach et al.

[2010], Liu et al. [2013] or the certain model organism Wang et al. [2010], Li et al. [2017], however, the applicability domain of such models is limited. The toxicity of chemical compounds is estimated using different types of biological assays which describe various toxic effects (neurotoxicity, cardiotoxicity, etc), model organisms (rodents, dogs, monkeys), or the toxicity outcome (LD50, LD100). Only a few compounds are investigated in several assays and unavailability of experimental data in all assays may prevent detection of their toxicity. However, since the toxicity datasets are correlated, we can expect that such correlations can be used to develop models with higher predictivity for each datapoint by modeling such datasets simultaneously (multi-task learning). The previously mentioned RTECS dataset, which contains data for different species and endpoints, is more suitable for such a study. This dataset was not widely used for the development of predictive models. We are only aware that part of it was used for mapping and chemical space visualization of the IDDB dataset von Korff and Sander [2006]. In this study we have addressed this question by using multi-task learning Unterthiner et al. [2015], Dahl et al. [2014], Sosnin et al. [2018c] with state of the art machine learning methods

4.1 Materials and Methods

RTECS dataset

We extracted acute toxicity data from Registry of Toxic Effects of Chemical Substances (RTECS) – is a database which collect the information about various toxic effects of chemical substances. We used RTECS database version 2018.1 to extract organic compounds with acute toxicity records available. Since the structures of organic compounds are not presented in the database, we extracted them from PubChem Kim et al. [2016] using Chemical Abstracts Service (CAS) Registry Number. The non-organic compounds, plant extracts, parts of biological compounds, and compounds containing elements other than (C, H, O, N, P, S, F, Cl, Br, I) were ignored.

The goal of this study was to examinate the toxic effects of the organic compounds. However, many compounds were reported in the database as salts or as

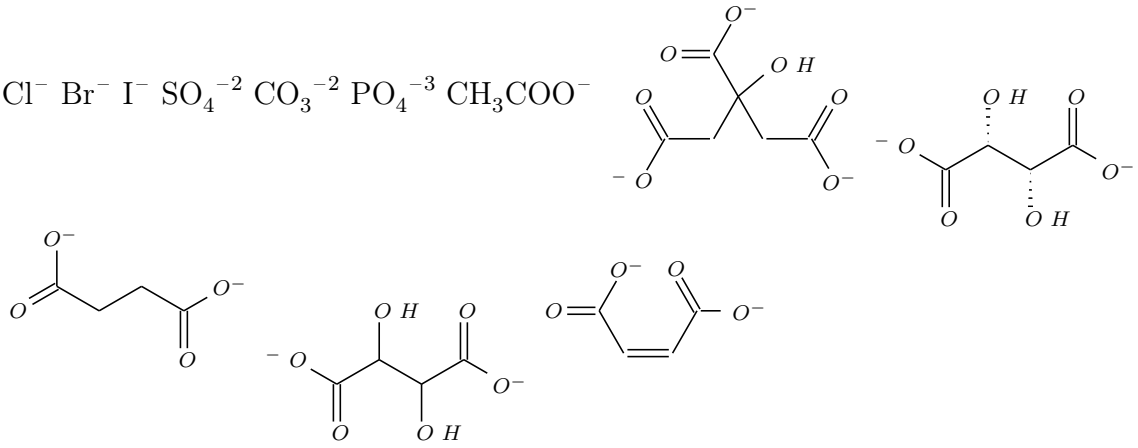


Figure 4-1: Ions considered to be nontoxic

mixtures, and some of the counterions are toxic themselves, e.g. methylsulphate ion ($CH_3OSO_4^-$). Their toxicity could interfere with the interpretation of the toxicity of the organic part. Therefore, only compounds with non-toxic counterions listed in Figure 4-1 were kept in the database. The compounds with other counterions and compounds with mixtures were eliminated. We also eliminated all polymeric substances. For the salts which were kept in the database, only the organic part was used to generate descriptors.

After the preprocessing stage, all compounds were grouped for the same toxicity type by two parameters: the route of administration and the animal species used for the experiment. We removed all records that had less than 300 reported measurements for each group to reduce the dimensionality of the output. As the result, a database with 129,142 toxicity measurements was created. It consists of 87,064 unique molecular structures and 29 unique endpoints. The sparsity (the percentage of the filled values) of the data matrix is 5.12%. The information on the endpoints is summarized in Table 8.5 in Supplementary Material.

Molecular descriptors

Different descriptor sets may have different performance in the modelling of toxicityFeng et al. [2003], Baskin [2018]. The testing of different sets of descriptors for the performance of single and multi-task models could help to better understand whether the performance of models depends on the used descriptor sets. There-

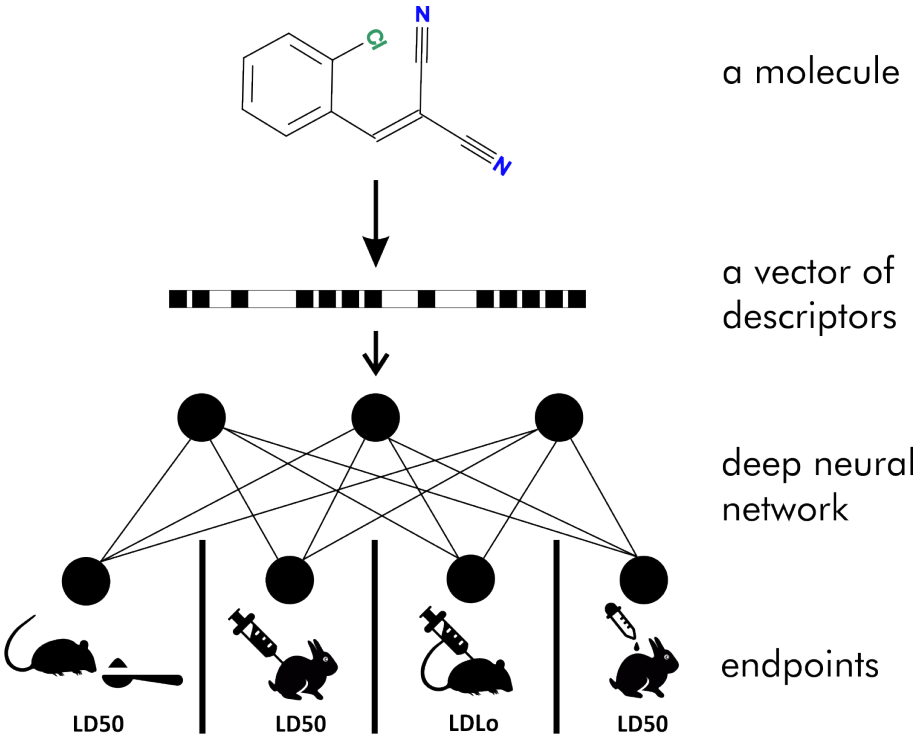


Figure 4-2: Representation of endpoints as outputs of a deep neural network.

fore, we calculated a number of molecular descriptor sets which are provided by the OCHEM platform. A short description of the descriptors used is given in *Table 4.1*.

It should be noted that OCHEM developed a new model on each validation step without using any information about the test compounds, which are only predicted following model developments. This provides correct validation (identical to the use of so-called "external sets") since no information about the test molecules is used to guide model development.

We implemented our DNN in Chainer Tokui et al. [2015b] framework and included it into the OCHEM Sushko et al. [2011] platform.

4.2 The description of RTECS chemical space

For the description of the whole dataset, we took the highest value across all endpoints for each molecule. For the generation of the 2D chemical space representation the calculated RDKit circular fingerprints (4096 bit vectors) based on the standard-

Table 4.1: The descriptors used in our experiment. Several descriptor blocks that are indicated by "(3D)" required 3D representation of molecules, which was calculated by using 2D to 3D structure conversion using *Corina* program.

Descriptor	Short description
PyDescriptor (3D) [129]	A PyMOL-based plugin for calculations different groups of descriptors
Dragon6 (3D) [208]	Descriptors provided by Dragon 6 program
SIRMS [116]	Calculates simplexes, which are n-atoms fragments of a xed composition, structure, chirality and symmetry
StructuralAlerts [192]	Presence of certain sub-fragments in molecular graphs which are believed to be related to toxicity of organic compounds
QNPR [205]	Uses substrings of SMILES as a representation of molecules
Spectrophores (3D) [203]	Spectrophores are one-dimensional descriptors that describe the three-dimensional molecular fields surrounding a molecule
Adriana (3D)	The descriptors provided by Adriana.CODE program
Inductive (3D) [30]	Descriptors based on inductive and steric effects of atoms
Chemaxon (3D)	A subset of descriptors calculated by Chemaxon (www.chemaxon.com) module in OCHEM
Mera and Mesry (3D) [159]	3D descriptors of molecules
GSFrag [159]	Descriptors calculated by GSFRAG program (the occurrence numbers of certain special fragments on k=2,...,10 vertices in a molecular graph)
Fragmentor [215]	Molecular fragments which contains from 2 to 4 atoms genereted by ISIDA module in OCHEM
ALogPS [200, 196], OEstate [67]	Prediction of logP by ALogPS2.1 program in combination with OEstate descriptors which are based on electrostatic properties of atoms and bonds
CDK2 (3D) [188]	Chemistry development kit descriptors, version 2.0
Morgan fingerprints [168]	Morgan (circular) fingerprints of radius two (which corresponds to ECFP4 [168]) calculated by RDKit

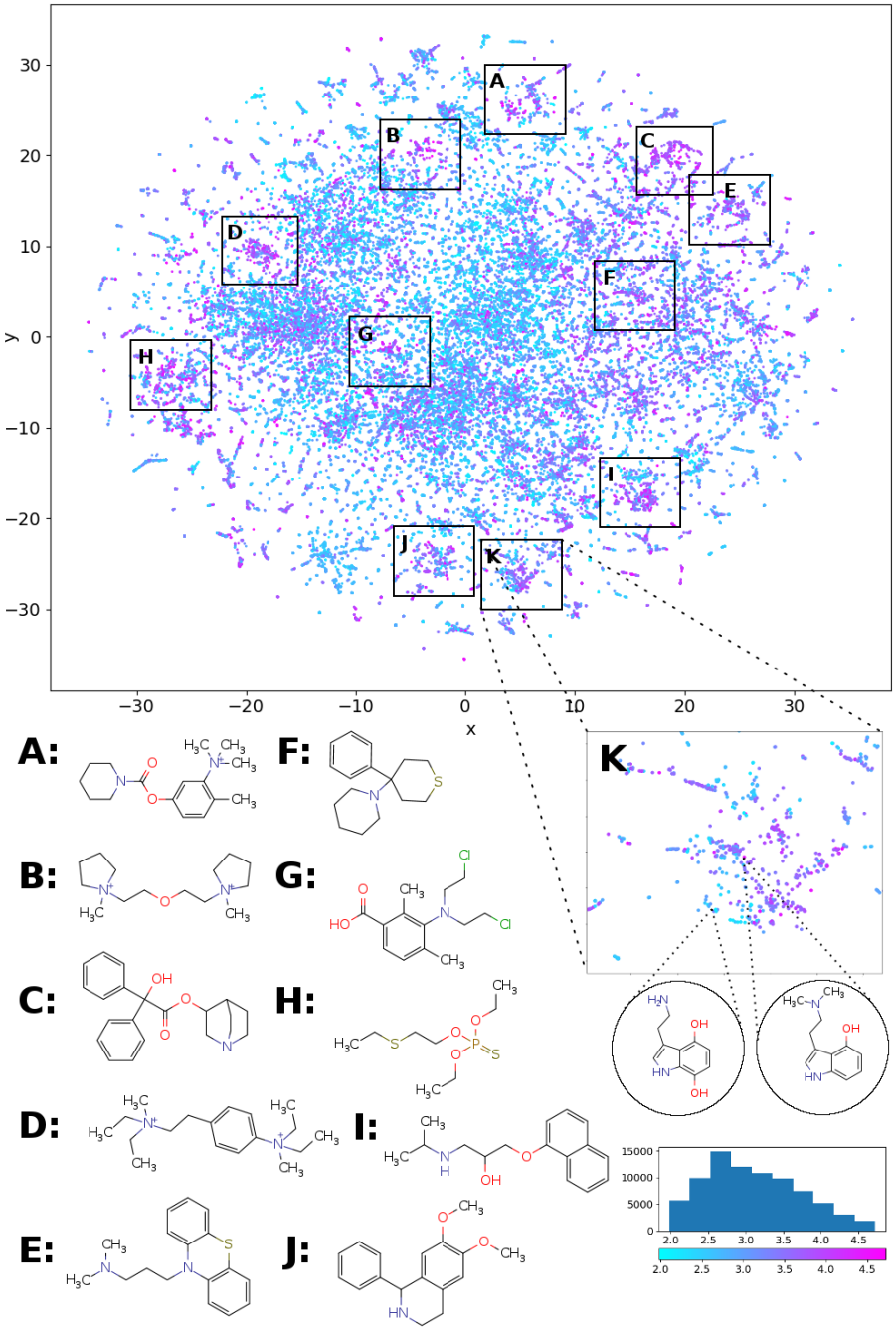


Figure 4-3: The RTECS chemical space visualization. Each point stands for the one molecular structure and its color indicates the acute toxicity values in log(mol/kg).

Table 4.2: The architecture of *dense7* neural network

Hidden Layer	Neurons	Batch Normalization	Dropout ratio
1	512	Yes	0.5
2	256	Yes	0.5
3	128	Yes	0.5
4	64	Yes	0.5
5	32	Yes	0.25
6	32	No	0.1

ized SMILES molecular representation (molvs python package) were embedded into the 2D space using the t-SNE method van der Maaten and Hinton [2008]. A pairwise distance matrix was calculated using the Dice metric, and the default values were chosen for parameters of the algorithm. Figure 4-3 shows the results of the chemical space embedded in the 2D space. Each point corresponds to a chemical structure and the color denotes the toxicity values according to the palette. Some of the toxic clusters are highlighted by the rectangular shapes and their representative members are visualized in Figure 4-3. We provide the description of several clusters composed from the relatively toxic molecules. The enlarged image of cluster **K** is given for clarity and demonstrates its composition from the hydroxytriptamine derivatives. Arylcarbamate (neostigmine derivative is shown as a representative cluster member) derivatives are embedded into cluster **A** and their toxic effects may be explained by the cholinesterase inhibition. Cluster **B** is composed of possible nicotinic acetylcholine receptor ligands. The derivatives of the 3-quinuclidinyl benzilate which is a potent muscarinic anticholinergic agent are the major members of cluster **C**. Cluster **D**, similarly to cluster **B**, is composed of compounds based on the two quarternary amine groups connected by a linker. Phenotiazine derivatives acting on a number of different targets and widely used as antipsychotic agents earlier are the major components of cluster **E**. Phencyclidine derivatives (NMDA-receptor channel blocker) are included in cluster **F**. Possible alkylating agents and organophosphorus compounds were grouped in clusters **G** and **H**, respectively. Cluster **I** is composed of the adrenoreceptor ligands and the propranolol structure is shown for example in Figure 4-3. And isoquinoline derivatives belong to cluster **J**. This result shows that toxic compounds are grouped by similar structural features and neighbor compounds

tend to have similar toxicity.

4.3 Correlation Analysis of Endpoints

Previous studies Xu et al. [2017b] pointed out that the efficiency of multi-task modelling depends on correlations between targets. To examine it, a correlation analysis of endpoints was performed. Pearson correlation coefficients between each pair of endpoints were calculated. Mutual correlations as heatmaps are presented on Figure 4-4. For the objective evaluation of correlations, we set a number of thresholds. If the corresponding endpoints have the number of simultaneous measurements less then a threshold, the color on the heatmap is absent. The successful application of multi-task modelling can be seen from the high correlations between endpoints. The high correlations between endpoints also reflect the good quality of the data presented in RTECS on the assumption that the provided measurements were independent.

4.4 Comparison of Models

Our main goal was to compare models of toxicity prediction built for different endpoints. In this study we defined each endpoint according to the conditions of the experiments. For example the LD50 toxicities measured when using intraperitoneal administration to mouse belong to the same endpoint. As a counterexample LD50 records with oral and intraperitoneal admission belong to different endpoints. However, due to hidden relations between endpoints we can expect that the multi-task (multi-endpoint) models should achieve better quality than single-task models. To prove the hypothesis we built multi-task DNN models *(MT_DNN)*, single target DNN models *(ST_DNN)*, and several models with other aforementioned machine learning algorithms, namely: *XGBoost*, *Random Forest*, *k nearest neighbors* . In order to show that the observed relationships are not specific to a single set of descriptors, we used all sets of descriptors reported in Table 4.1. The performance of different models is given in Figure 4-5.

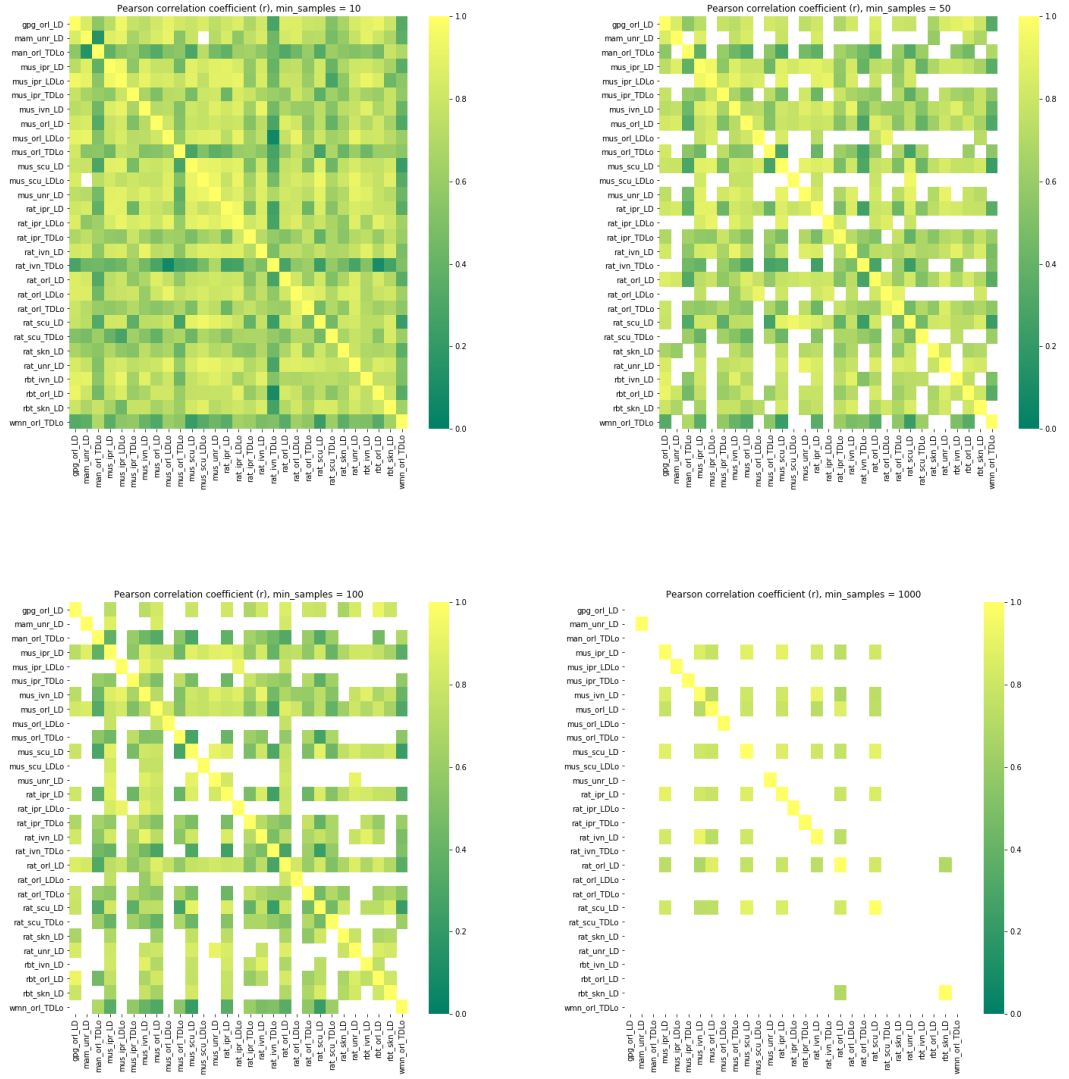


Figure 4-4: Matrices of correlations for endpoints with various thresholds (*min_samples*) values. The toxicity endpoints demonstrate their correlation notwithstanding the number of compared samples.

The *MT_DNN* models outperformed both *ST_DNN* models and all other methods used for all analyzed sets of descriptors. Models, which are based on *ALogPS* combined with *OEstate* descriptors achieved the best average performances across all studied methods. The red dashed line on Figure 4-5 corresponds to average RMSE $= 0.71 \pm 0.01$, which was calculated using *MT_DNN* for several sets of descriptors, namely *Fragmenter*, *CDK*, *Dragon* and ALogPS, OEstate. The performances of *ST_DNN* models were comparable with *XGBoost* and *Random Forest* models. This result is not surprising, and consistent with previous studies Sheridan et al. [2016], Zhang et al. [2017] where the efficiency of these methods was similar. The *Random Forest* method achieved a better average performance compared to the *XGBoost* method. This can be related to the robustness of this method in comparison to that of *XGBoost*. One should carefully select the *XGBoost* parameters to achieve close to optimal solution, while Random Forest usually provides high quality results for the models "out-of-the-box". We also experimented with other ANN types. Associative neural networks (ASNN)s Tetko [2002 May-Jun] required long computational time, because they used CPU and not GPU computing. This algorithm, which was based on a so-called "shallow neural network" with one hidden layer, provided a lower accuracy presumably due to the absence of latent representation of the molecules (in deep neural networks latent representation is commonly regarded as the outputs of second-to-last layer).

Endpoints modelling

We also compared the quality of models for each individual endpoint. To do that, a consensus model which averages of the outcomes of the top 5 descriptor models was created. There were 29 endpoints which represent 4 animal species: mouse, rat, rabbit, guinea pig, one unspecified class, and two classes of human based on gender: man and woman, several types of administration and 3 outcomes: Lethal Dose Fifty (**LD50**), Toxic Dose Low (**TDLo**), Lethal Dose Low (**LDLo**). The numbers of records for each endpoint are given in *Table 4.1*. Our automatic data extraction procedure keeps the extracted endpoints unchanged, that is why the human toxicity is reported separately for man and woman and an "unspecified" animal class

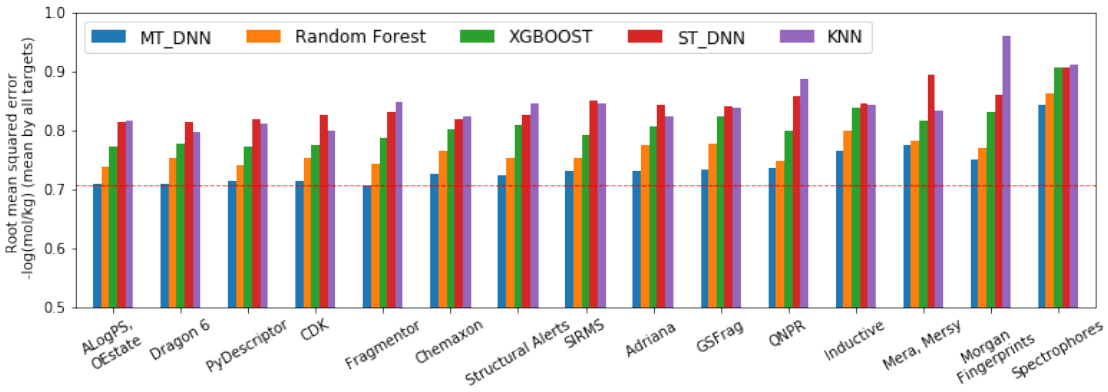


Figure 4-5: Average RMSE of predictions of toxicity for all endpoints in -log(mol/kg) units by different methods and descriptor sets. Descriptors were arranged in accordance with mean values of predictions by all methods (the best are on the left). Methods are ordered by the mean RMSE over all descriptors (*MT_DNN* and *KNN* demonstrated highest and lowest overall performances, respectively).

is also present. There is the significant gap in the quality of prediction of toxicity for different endpoints. **LD50** values were predicted with relatively good quality for several species and several types of administration: for **mouse intravenous**, **oral**, **subcutaneous** The **LD50** type of toxicity gave the value of $R^2 \geq 0.65$ for corresponding models. The same model quality is observed for **rat** and **rabbit intravenous LD50** toxicity. It should be noted that **LDLo** was predicted with lower accuracy than **LD50** toxicity for all species and administration types. For **TDLo** the prediction accuracy is inferior: R^2 values for those targets are in the range 0.26-0.43 which is fairly low. The low accuracy of the prediction of these endpoints can be explained by the limited amount of data for these types of toxicity. Moreover, **TDLo** and **LDLo** measurements are less reliable due to disproportionately inaccurate experimental conditions (e.g. could be contributed by other sources of toxicity not directly related to the analyzed compounds) the instrumental errors during measurements were higher for these endpoints, since both of these toxicities have lower values compared to LD50. The target with the lowest error is **rat**, **intravenous**, **LD50** with $R^2 = 0.71$ and $RMSE = 0.54$. Toxicity for humans is represented only by **TDLo** values and the quality of prediction of models for this target is unsatisfactory. This is related to the factors mentioned above and it should inspire new developments because of the extreme importance of such modelling to drug

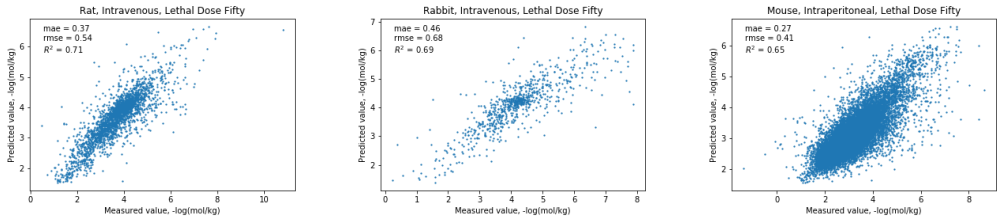


Figure 4-6: Prediction charts for a number of selected endpoints

development. On *Figure 4-6* we demonstrate some representative prediction charts, a full set of prediction charts (for each endpoint) can be found in Supplementary Information.

4.5 Attributed Modeling

Multi-task modeling can be approximated as single-task where the endpoint tags are provided to the input of the model as attributes. For example in our case animal species as soon as a type of administration and type of toxicity can be encoded by one-hot encoding and concatenated with a vector of chemical descriptors. The scheme of the attributed modeling is given in Figure 4-7. The advantage of the attributed modeling is the possibility to use any machine-learning algorithm without additional modifications of a loss function. We compared the performance of consensus XGBoost attributed model with consensus multi-task DNN model and consensus single-target DNN model. XGBoost method has been chosen due to both quickness and its ability to achieve the good quality among single-target models. Our experiments revealed that there is no significant discrepancy between the performance of the multi-task DNN and the attributed XGBoost model. The statistical performance of different modelling schemes is given in Table 4.3.

Feature Net approach

The Feature Net approach has been proposed as a variant of multi-task learning. The main idea of this approach is to use a predictions of one (or a group) model as additional descriptors for the resulting ST models. It was shown that the Feature

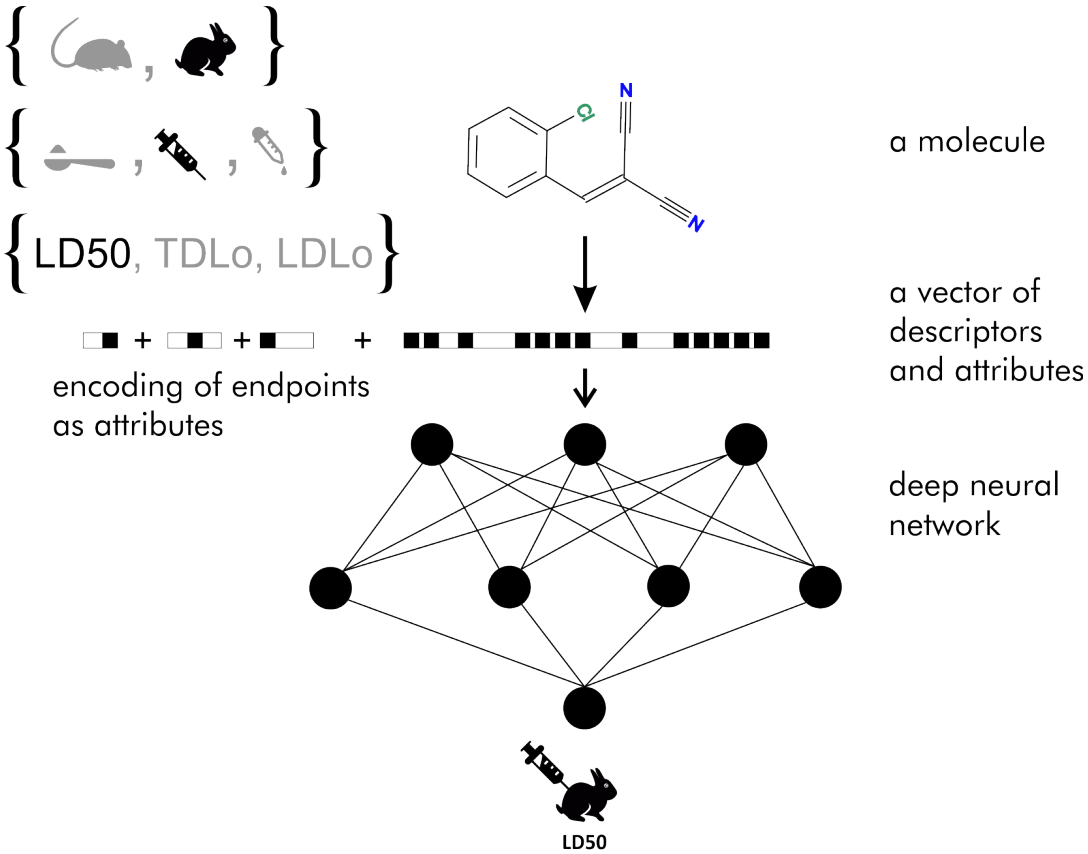


Figure 4-7: Representation of endpoints as attributes in STL modelling. The encoding of endpoints as input descriptors allows their simultaneous prediction using neural network with one output neuron.

Table 4.3: The comparison of quality of two consensus attributed models with consensus multi-task model (averaged over all endpoints)

Model	MAE	R^2	RMSE
DNN (attributed)	0.49	0.54	0.69
XGBoost (attributed)	0.49	0.55	0.68
DNN (multi-task)	0.49	0.55	0.68

Table 4.4: The comparison of RMSE for models based on Feature Net approach with multi and single task models (averaged over all endpoints)

Descriptors	Feature Net	ST DNN	MT DNN
Dragon 6	0.77	0.85	**0.74**
ALogPS, OEstate	0.75	0.86	**0.74**
Fragmentor	0.77	0.88	**0.74**
PyDescriptor	0.76	0.85	**0.74**

Net approach can achieve better accuracy than single-task learning Varnek et al. [2009] and can provide models with similar accuracy to MT models. We used results of *ST_DNN* as the feature nets to train the models and after that we used these predictions as additional descriptors to develop final models. The statistical performance of these models are given in Table 4.4.

We observed that for all descriptors the general trend remains the same. The accuracy of Feature Net models is between that of single-task models and multi-task models. We believe that Feature Net models partially regard latent correlations in the data; however, the multi-task models have significantly better performance. Taking into account that fact that the Feature Net approach requires significantly more time compared to MT models the feasibility of usage of this approach is questionable.

Censored data modeling

Toxicity datasets frequently include a significant number of records reported as intervals e.g., "$>$" (greater than), in cases where the exact value of toxicity has not been measured. This frequently refers for non or low toxic compounds or for compounds for which high concentrations can not be achieved due to solubility or availability. This problem is known in literature as censored data modelling. To solve this problem a number of statistically-based approaches were proposedGijbels [2010]. This large number of the records without exact toxicity values is a special problem in automatic data analysis. The most common approach in this case is to set the maximal toxicity dose observed in the whole dataset for these types of

records, considering them to be nontoxic. But in case of particularly heterogeneous data this approach is not optimal due to the large variations in the toxicity values for different endpoints. We propose a modification of a loss function which allows the correct processing of such records; the formula for a RMSE loss function over a batch which regards intervals is given below:

$$L(y,\hat{y}) = \begin{cases} \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - max(\hat{y}_i, y_i))^2 & if \quad > \\ \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - min(\hat{y}_i, y_i))^2 & if \quad < \end{cases}$$

where $\hat{y}_i$ – is a predicted value, y_i – is a real value, n – total number of samples in a batch.

To estimate if the training on ranged data can improve the quality of models or we trained two models: one with modified loss function and one with the standard RMSE loss and applied them to compounds with exact values of toxicity. No significant difference between models trained with modified loss functions and with RMSE loss was observed . This experiment demonstrated this method is not efficient for the dataset under study and the standard loss function i.e. RMSE or MAE during training is preferable. Nonetheless, the correct and efficient processing of ranged data, especially for large diverse datasets, might be crucial for other applications and should be kept in mind.

Latent representation of compounds

Neural networks generate a hidden representation of data on their hidden layers by processing the data. We visualized this process directly by performing projection of the latent representations of the compounds onto the 2D plane by the t-SNE method. The neuron's activation on the last-to-last ANN layer for the molecule was used as their hidden representations. The visualization of this latent space on *Figure 4-8* shows that ANN on the last hidden layer achieves good separation of toxic and non-toxic compounds but generally does not group structurally similar compounds together. One can notice three areas containing the most toxic compounds and each of these groups are composed of different compounds: organophosphorus compounds, sterane derivatives, etc.

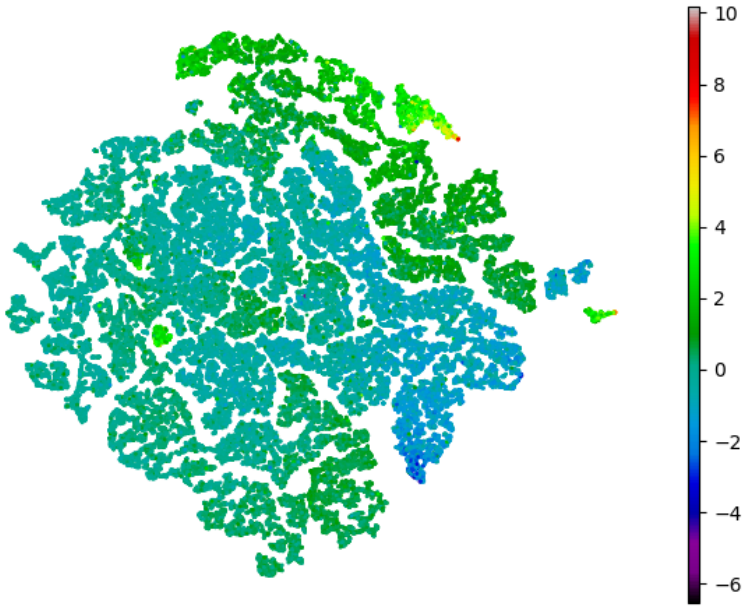


Figure 4-8: The results of the application of the t-SNE method to deep features generated by the multitask DNN, values are minus logarithms of maximal endpoint (greater values correspond to larger toxicity). Several clusters with high toxicity are observed.

Regulations in the light of multi-task learning

Recent progress in QSAR/QSPR modelling raises questions about the correspondence of newer methods to guidelines established and approved by authorities. In this section we would like to put forward for discussion the OECD principles for the validation, for regulatory purposes of QSAR models. *"Guidance Document on the Validation of (Quantitative) Structure-Activity Relationship [(Q)SAR] Model"* summarized the collective opinion of OECD specialists to QSAR modelling. In this document a peculiar attention is given to the *Principe N 1* – a defined endpoint. Despite of an uncertainly of formalizing defined endpoint, the authors of the guideline warned researchers from usage of endpoints which are not clearly defined. We agreed with the authors that for a QSAR model the endpoint should be clearly defined, but we believe that the current description of the defined endpoint is insufficient. For example Item 68. states that *"4. The chemical endpoint of the (Q)SAR should be contained within the chemical endpoint of the test protocol. 5. The endpoint being predicted by a (Q)SAR should be the same as the endpoint measured by a defined test protocol that is relevant for the purposes of the chemical assessment."* The interpretation of this formulation may prohibit the usage of multi-task learning. In the same time, we are at the beginning of a "big data" time Tetko et al. [2016b] in chemistry and biology The appearance of these data promotes development of powerful multi-task models that could significantly increase quality of models for individual end-points. But these methods can break the paradigm "one accurate dataset" $\longrightarrow$ "one model for narrow endpoint". It should be mentioned that the *Feature Net* approach, in principle, can still allow us to use the OECD principles by treating predictions of STL models as additional descriptors. However, as we have shown in our studies this approach many not allow us to use the full advantages of multi-task modeling.

4.6 Conclusions

In this study the efficiency of several methods of machine learning and several types of descriptors was estimated on a large multi-task dataset. The statistical analysis

of the data extracted from the largest toxicity dataset the Registry of Toxic Effects of Chemical Substances (RTECS) was performed. We demonstrate that multi-task deep neural networks can significantly improve prediction of toxicity by comparing them to investigated single-output types of models including: single-task deep neural network, *XGBoost*, *Random Forest*, *K-nearest neighbors*. The models with highest prediction abilities were obtained for *rabbit* and *rat* species.

Interestingly, the attributed models (target endpoints are encoded with additional descriptors), and multi-task models (each endpoint corresponded to one output) demonstrated similar accuracy. While the *Feature Net* approach contributed better models than single-task models, it performed worse than the multi-task models. Our results demonstrate that multi-task approach can be beneficial for toxicity prediction due to its ability to process a heterogeneous dataset containing different endpoints.

Chapter 5

Chemical space visualization guided by deep learning

Visual representation of the chemical space is growing in popularity and medicinal chemists use it to have the better understanding of chemical data Osolodkin et al. [2015]. Technically, it is an information-losing projection from multi-dimensional molecular space (commonly described by molecular descriptors, so-called descriptor space) into two- or three-dimensional space, in which humans can operate easily. The majority of chemical space visualization methods use two discrete procedures:

- calculation of molecular descriptors
- performing a projection from descriptor space into a 2D plane or 3D volume by one of several known techniques Sorzano et al. [2014]

There is the option to combine different descriptors with different dimensionality reduction algorithms, however, sometimes authors of a visualization method propose a suitable combination of molecular descriptors and algorithms for better performance, e.g. GTM Bishop et al. [1998] may be successfully combined with ISIDA descriptors Baskin et al. [2017].

The type and the length of the descriptor vector influences the details of the chemical representation, and the choice of the feature set is driven by the expected depth of description. Molecular quantum number (MQN) Reymond et al. [2010] is an example of a simple molecular descriptor set consisting of atomic and bond

counts and some other topological descriptors. Despite the fact that the size of the descriptor set is relatively short (42 descriptors), this method performed very well in the identification of the novel nAChR allosteric modulators Bürgi et al. [2014]. Alternatively one can use a fingerprint description of the molecular structure, which is a bit string where each bit indicates the existence of predefined substructure (MACCS Structural Keys; Symyx Software: San Ramon, CA, 2002.) or the certain atom types in the predefined atomic environment (ECFP fingerprints) Glen et al. [2006].

Here we describe an application of deep feed-forward neural networks as a t-SNE mapper to the bioactivity data taken from A Database of Useful Decoys (DUDe)Mysinger et al. [2012] and the Trace Amine Associated Receptor 1 (TAAR1) ligands visualization task. The workflow consists of three main stages. First, we trained a set of the mapper functions varying the perplexity level in the loss function with the overfitting controlled by the external test set (Figure 5-1). Second, since the dimensionality reduction techniques lead to information loss, we trained a set of classifiers on the mapped 2D data and compared the resulting accuracy. Third, we provide the visualization and analysis of the TAAR1 data set taken from PubChem.

5.1 Materials and Methods

A number of dimensionality reduction techniques were utilized for the processing of molecular databases and here we will briefly review the most important of them, commenting on their relative strengths and drawbacks.

The algorithm of Principal component analysis (PCA) performs an iterative search of directions with the highest variation in a multidimensional data space. Usually the first two components are easily interpretable and explain 60-80% of the whole variation in the data Osolodkin et al. [2015]. PCA-based mapping is fast, deterministic, and new compounds may be easily mapped using the principal components of an existing data set, but this method omits non-linear feature interactions Rose et al. [1991] and some map regions become overloaded with data Blum et al. [2011]. The method of Self-Organizing Maps (SOM) Kohonen [1982], Awale and Rey-

mond [2016] usually treats non-linearities in a better way, mapping the feature space to the low dimensional visualizable space. The Generative Topographic Mapping Bishop et al. [1998] approach represents a probabilistic alternative to SOM Kireeva et al. [2012]. This approach was applied to large data set collections identifying desirable chemical space regions for drug design Gaspar et al. [2015] and was successfully used for large-scale Structure–Activity Relationship (SAR) exploration Kayastha et al. [2017]. It is worth mentioning non-coordinate based approaches developed by the group of Jürgen Bajorath, which transform multidimensional chemical space to a graph with nodes representing chemical compounds, and edges connecting compounds within a specified similarity cut-off de la Vega de León and Bajorath [2016]. The other approach, so-called Scaffold Trees, treat the chemical space as a tree where leaves represents individual chemical compounds and the intermediate nodes represents scaffolds and subscaffolds Schuffenhauer et al. [2007].

A number of useful tools combining a variety of visualization approaches were created in the recent years. Stardrop (Optibrium Ltd., Cambridge, UK) and DataWarrior (openmolecules.org) combine a variety of visualization approaches with chemoinformatic data analysis. The *CheS-Mapper* Gütlein et al. [2012] tool, which is used for the visualization of chemical data sets in 3D space, provides both a number of chemical descriptors and several projection algorithms i.e. PCA, t-SNE, and also gives users the possibility to combine them.

5.1.1 Datasets

ChEMBL: Molecular structures for training were extracted from ChEMBL Gaulton et al. [2017] v.23. Only SMILES strings with lengths between 10 and 150 characters have been selected, yielding a data set containing 1564049 unlabeled items. Obtained SMILES representations were standardized using the *molvs* Python package and subsequently used for the computation of ECFP6 fingerprints comprising 2048 bit length. Then the data set was randomly split into training (90 %) and test (10 %) samples which were subsequently used for training and mapper quality estimation.
DUDe: In order to assess visualization performance we used data sets collected from DUDe Mysinger et al. [2012] which is successfully used for the assessment of

molecular docking performance. Two subsets containing GPCR and nuclear receptors' ligands and having relatively high similarity inside each group were selected for analysis. It should be noted that GPCR (contains 5 classes) and nuclear receptors' (contains 11 classes) data sets contain information about 1480 and 2995 chemical compounds, respectively.

TAAR1 ligands: 415 Trace Amine Associated Receptor 1 agonists with annotated EC50 values were taken from PubChem Kim et al. [2016]

5.1.2 Parametric t-SNE

Stochastic Neighborhood Embedding initially was proposed by Geoffrey Hinton as a method for the dimensionality reduction. The t-SNE approach, proposed by L. van der Maaten, has gained tremendous popularity in data visualization, however, it has two notable drawbacks:

- it can not be directly applied to new data (in other words when a new portion of data is obtained the whole data set must be reevaluated again)
- the computational complexity of the distance calculation is quadratic which requires the usage of approximations (i.e. Barnes-Hut approach) for the analysis of large databases

In practice, even with the Barnes-Hut approximation, applying t-SNE to more than 10^5 compounds on modern computers is computationally unfeasible. To overcome these problems we focused on the *Parametric t-SNE* algorithm that was proposed by the same author van der Maaten [2009]. In *Parametric t-SNE*, a function which performs a mapping from the high-dimensional descriptor space to a low-dimensional space (2D or 3D) $f : \boldsymbol{X} -> \boldsymbol{Y}$ is a normal feed-forward neural network with trainable weights. It should be noted that in the original paper the authors used Restricted Boltzmann Machines as their mapping function because they provide a good speed of computation, however, nowadays feed-forward neural networks trained on GPUs can be feasibly used as an alternative. At the first stage of the algorithm a distance matrix should be computed using a task-relevant distance metric. Then each row of the distance matrix is transformed into the probability distribution:

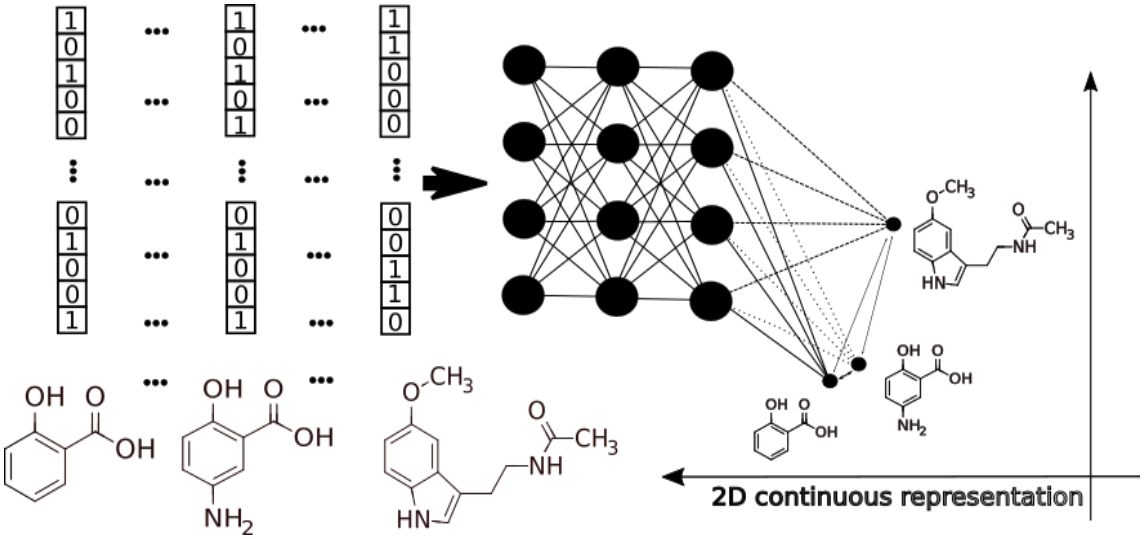


Figure 5-1: The schematic workflow of the pT-SNE mapping procedure

$$p_{ij} = \frac{e^{-\beta_i d_{ij}^2}}{\sum_{k \neq i} e^{-\beta_i d_{ik}^2}} \tag{5.1}$$

Where the parameter $\beta_i = \frac{1}{2\sigma_i^2}$. σ_i^2 is the bandwidth of the Gaussian kernels, β_i – is a tunable parameter, and it is tuned over batches to follow the pre-defined perplexity of data. The perplexity is:

$$Perp(P_i) = 2^{H(P_i)} \tag{5.2}$$

where $H(P_i)$ – is Shannon entropy:

$$H(P_i) = -\sum_{j=1}^{N} p_{ji} log_2(p_{ji}) \tag{5.3}$$

Perplexily is a predefined parameter of the algorithm. The parameter β_i is found by binary search to satisfy the predefined perplexity. When the described transformation is applied to each i row of the distance matrix we can observe that almost all elements of each row become zeros except some neighboring items to i item in terms of the used distance metric. This distribution defines the probability to pick j item (where $j \neq i, 0 < j \leq$ batch size) as a neighbor of i item among the whole batch. Our implementation allows us to perform this task on a GPU, increasing

the speed of training. The pairwise similarities in the latent space are computed using Student t-distribution to overcome the "crowding" problem van der Maaten and Hinton [2008] in the same way as in the high-dimensional descriptor space except the euclidean distances were chosen as a distance metric (2). The cost function is defined as Kullback-Leibler divergence Kullback and Leibler [1951] between joint probability distributions in high-dimensional space P and in low-dimensional space Q (3). α is the number of degrees of freedom, used in the definition of t-distribution.

$$q_{ij} = \frac{(1 + ||y_i - y_j||^2/\alpha)^{-(\alpha+1)/2}}{\sum_{j \neq k} (1 + ||y_i - y_k||^2/\alpha)^{-(\alpha+1)/2}} \tag{5.4}$$

$$L = KL(P||Q) = \sum_{i \neq j} p_{ij} log \frac{p_{ij}}{q_{ij}} \tag{5.5}$$

Where L is a loss function used for optimization of the weights of the neural network. Choosing of an optimal α value is an open problem, however L. van der Maaten in his original work van der Maaten [2009] defined some possible approaches. In our research, we start with α equal to one and along with updating weights in the mapping function we compute gradient and update alpha similarly.

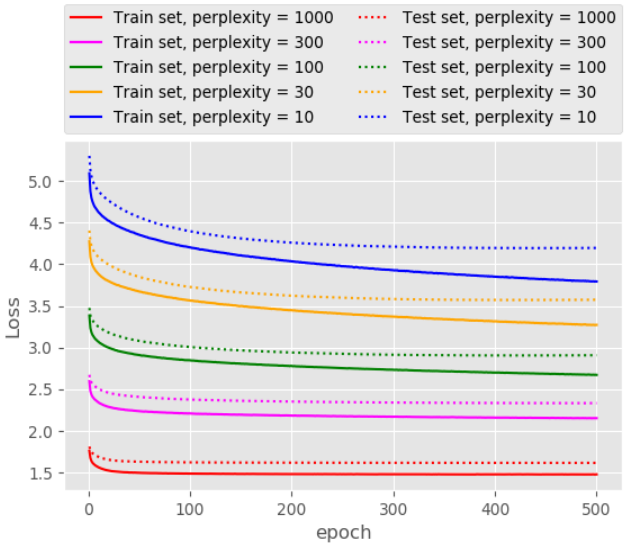


Figure 5-2: The learning curves obtained for different perplexity values

Artificial neural networks: We used deep artificial neural networks as a mapping function in our variant of *Parametric t-SNE* which projects the input space into 2D

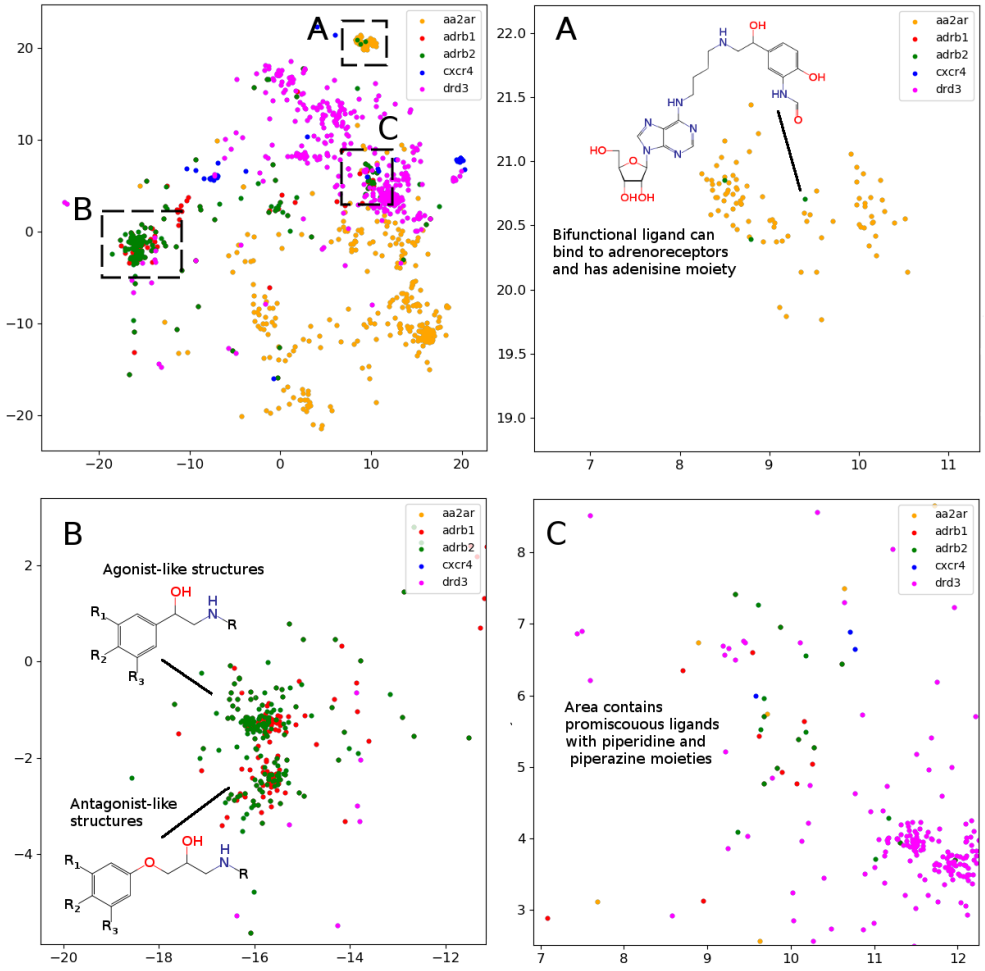


Figure 5-3: The results of the neural network mapping for a set of GPCR (A) ligands. **A** contains ligands of adenosine A2 (aa2ar), adrenoreceptors $\beta 1$ (adrb1) and $\beta 2$ (adrb2), chemokine CXCR4 (cxcr4) and dopamine DR3 (drd3). **B**, **C**, **D**, contains zoomed area from **A**. (Perplexity 100)

space. The architecture of the network and parameters of optimization are given in Supplementary Information. In our experiments we tested ECFP6 fingerprints (2048 bits). All fully-connected layers except the last one are followed with a batch normalization layer Ioffe and Szegedy [2015]. Rectified linear units (ReLU) were used as activation functions on the first three layers and the appropriate weight initialization was performed. Different perplexity values (10, 30, 100, 300, 1000) which can be understood as a mean number of neighbours taken into consideration were also tried at the training step. It should be noted that the resulting basis vectors of the output 2D space can not be easily interpreted in comparison with the results of PCA analysis.

We tried two different distance metrics: Euclidean and Jaccard distances. Due to the fingerprints' sparsity the common approach of cosine distance is inconvenient for this task and in our experiments Euclidean distance tended to overestimate similarity among small molecules. Because of the possibility of performing the training process in batch mode it is not necessary to compute the distance matrix for the whole data set, which reduced computational time and memory consumption and allows the processing of very large data sets.

5.1.3 Dimensionality Reduction Methods

Principal component analysis (PCA): is an orthogonal linear transformation which transforms the data into a new coordinate system where the first direction of the greatest variance become the new coordinate axis Bishop [2006]. This iterative approach allows the creation of new orthogonal basis sets and gives 2-3 components which usually explain the majority of data variance.

Multidimensional scaling (MDS): seeks the low-dimensional representation of high-dimensional data where distances in both representations are maximally close to each other Kruskal [1964].

5.1.4 Validation protocols

To control overfitting during training our mapper ANN we used 10% of the data as test set. Stratified Five-fold cross-validation was carried out to prevent overfitting and to compare the performance of the classification methods trained on the mapped data. For our multiclass classification models we calculated the accuracy of classification among all classes.

5.2 Results and Discussion

The main goal of visualization is to generate insight for the next step of the research. This is especially important for SAR exploration due to the fact that even small modifications of a scaffold may require additional synthetic efforts and one may

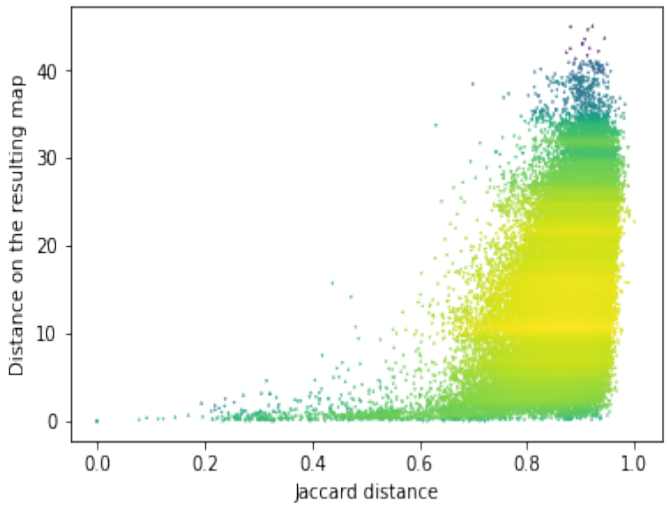


Figure 5-4: The dependence of the the resulting distance on the initial molecular similarity for the TAAR1 data set (Perplexity 100). Points' colors were set according to the density level: yellow means the highest density while magenta indicate the lowest one.

want to correctly prioritize further modifications to explore interesting regions of the chemical space Vogt [2018]. Let us clarify which regions of the chemical space are interesting. First, we should mention the areas of chemical space where the activity changed only slightly upon gradual structural changes which may be considered as activity plateaus and are useful for ADME tuning in the course of lead optimization. Second, the regions where small structural changes lead to strong changes in activity are called activity cliffs are associated with large SAR information content. The straightforward visualization and identification of such regions requires similarity preservation while mapping from high-dimensional descriptor space. Thus, the t-SNE objective perfectly meets this requirement. The learning curves are shown in Figure 5-2. The lowest and the highest loss values were obtained for perplexity values equal to 1000 and 10, respectively, as one may expect. Interestingly, the same trend was found for the loss decay during training: perplexity values of 1000 leads to a larger decrease in loss in comparison to perplexity values of 10. Also we tried to optimize the α value in the loss formulation which lead to significant loss decay as compared to the fixed $\alpha = 1.0$. Unfortunately, this parameter tended to zero during optimization on the ChEMBL data set. The decrease in this parameter means that the span of the map will increase allowing the map to occupy more area.

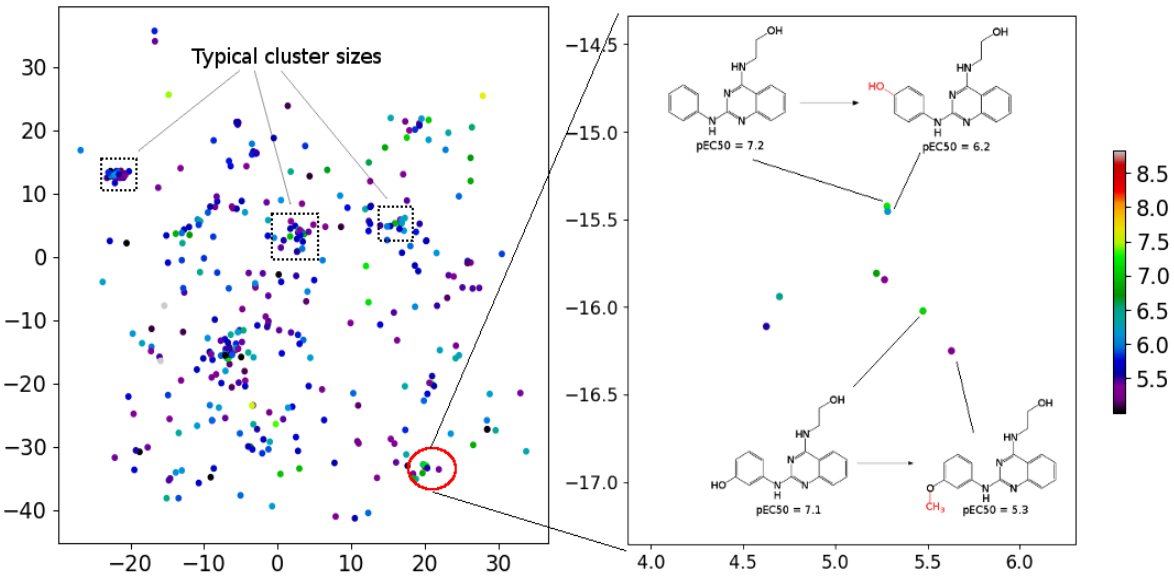


Figure 5-5: The mapping results for TAAR1 agonists data set (Perplexity 100). Points' colors were set according to the pEC50: yellow means the highest activity density while magenta indicate the lowest one

In order to assess visualization performance we used data sets collected from DUDe Mysinger et al. [2012] which has been successfully used for assessment of molecular docking performance. Two subsets containing GPCR and nuclear receptors' ligands and having relatively high similarity inside each group were selected for analysis. It should be noted that GPCR and nuclear receptors' data sets are highly balanced in terms of class composition and contain information about 1480 and 2995 chemical compounds, respectively. Figure 5-3 demonstrates the results of the neural network mapping for the GPCR ligand subset. The subgraph in the upper-left corner shows the overall view of the 2D representation. It should be noted that GPCR ligands used for analysis turned out to be highly separable and the overlap between classes is observed for highly similar receptors: $\beta 1$ and $\beta 2$ adrenergic receptors. Unfortunately, DUDe does not contain any information about the promiscuity of the active compounds but the cluster overlap may indicate such properties. Figure 5-3 (**B**) demonstrates the separation of the two clusters of β adrenergic receptors ligands: agonists and antagonists. Figure 5-3 (**A**) demonstrates the existence of the of the $\beta 2$ adrenergic ligand (green) in adenosine A2 ligand cluster. Interestingly, all these ligands contain an adenosine moiety which explains the mapping results.

Table 5.1: The results of application of the machine learning methods to the initial ECFP6 fingerprints and to the 2D mapped space (multiclass classification)

Descriptor	ML method	Accuracy	
		GPCR ligands	NR ligands
ECFP6 descriptors	kNN	0.829	0.526
	SVM	0.821	0.549
	XGBoost	0.821	0.540
	Random forest	0.788	0.537
pTSNE mapping (2D space)	kNN	0.763	0.383
	SVM	0.704	0.336
	XGBoost	0.764	0.394
	Random forest	0.745	0.360
PCA mapping (2 components)	kNN	0.739	0.296
	SVM	0.735	0.345
	XGBoost	0.743	0.360
	Random forest	0.735	0.349
MDS mapping (2D space)	kNN	0.725	0.326
	SVM	0.543	0.250
	XGBoost	0.712	0.333
	Random forest	0.707	0.328

Area **C** (Figure 5-3 (**C**)) shows the mixture of promiscuous ligands based on piperazine and piperidine scaffolds which can be found in different GPCR ligands (opioid, dopamine, serotonin receptors, etc.)

All dimensionality reduction techniques are often performed to get rid of noise in data but at the same time some information loss should be expected. Thus, we carried out the estimation of classification accuracy for two DUDe subsets containing GPCR and nuclear receptor ligands using widely known machine learning methods. The dimensionality of the data sets was reduced with PCA, MDS (Jaccard dissimilarity was used to construct the distance matrix) and pT-SNE trained as discussed above. The results of the performance estimation are shown in Table 5.1. First, it should be noted that the best achieved accuracy differs between the used data sets probably due to the fact that the GPCR subset contains fewer classes. For all constructed models the best accuracy was achieved for the initial descriptors (ECFP6 fingerprints) as was expected, and the pT-SNE dimensionality reduction technique significantly outperformed the other ones. The search for the optimal parameter set resulted in highly converged accuracies for methods on untransformed finger-

prints. For example, the difference in accuracy is observed only in the third decimal place when applying kNN, SVM and XGBoost on the GPCR data set, implying near-optimal models prior to mapping. The parameter sets yielding the highest accuracies were relatively similar for different dimensionality reduction techniques and appeared to be quite different for the both data sets. For example, the number of neighbours to achieve the highest accuracy for kNN was 24 for the GPCR and 9 for NR data sets. Interestingly, the SVM method demonstrated good performance for the initial fingerprints and the results of PCA, while the application of non-linear dimensionality reduction techniques (pT-SNE and MDS) yielded relatively worse performance. The XGBoost hyperparameter optimization resulted in a relatively similar set with variation only in the L2 regularization term, while the tree depth and the learning rate practically did not differ. It was found that the best value of the perplexity parameter is data set specific: 30 resulted in highest accuracy for the GPCR set after pT-SNE dimensionality reduction while 100 was the best for nuclear receptor ligands. These results are consistent with the fact that a perplexity value of 30 is a good starting point for visualization and usually recommended.

In order to assess the performance of the trained neural network to analyze the activity landscapes we used the TAAR1 receptor agonists' database collected from ChEMBL with measured activity in pEC50 and containing information about 376 chemical compounds. Let us compare the distance distribution in this data set in the original space and in the 2D mapped space (Figure 5-4). First, the distribution practically does not depend on the perplexity level. Second, similar compounds (Jaccard distances within 0.1 - 0.5) are very close together and dissimilar compounds (Jaccard distances more than 0.6) can be at any distance on the map. We estimated the uncertainty of the mapping performing the forward pass of the network using weights obtained during the last 100 epochs of training and found that in average the point position remained within 0.5 for both axes. As one can notice from Figure 5-5 (left) the typical cluster size lies within 2.0 - 3.0 and the compounds' distributions within the clusters remain relatively stable upon small perturbations in network weights near the local minimum. This is why one can easily analyze the activity landscapes. Unfortunately, the mapping does not guarantee that "very-very" similar

compounds will be closer together than just "very" similar compounds as one can notice from Figure 5-5 (right).

5.3 Conclusions

Understanding the internal relations in the chemical database is a key feature for the exploration of the chemical space to develop new substances with predefined properties. Visualization of the target chemical space by mapping from multidimensional descriptor space into space convenient to perceive is still a challenging task for chemoinformatics and computational medicinal chemistry. Stochastic Neighbour Embedding (SNE) and its modification t-SNE which preserves the points' positions in the target space to be t-distributed are not widely used for chemoinformatics tasks due to a number of problems: the high dimensionality of the initial descriptor space required to correctly describe chemical structure, computational cost, and non-deterministic results due to the stochastic nature of mapping etc. We show that parametric t-SNE approach can yield a neural-network-based function to map new portions of data. The speed of computation is comparable with other fast and widely used methods (PCA, MDS, etc.) while it preserves more information. This approach could be further explored for the interpretation of structurally-conditioned biological properties of chemical compounds.

Chapter 6

Legogram: Optimized molecular grammars for structures generation

Exploration of chemical space by a direct generation of molecules with desired properties is a challenging task. The main problem is the lack of a method for the direct production of molecular structures. Until recently, there was no simple and efficient way for machine learning-aided molecules generation. The breakthrough research of Gómez-Bombarelli et al. [2018] demonstrated the possibility of automatic chemical design using Recurrent variational autoencoder (RVAE). The authors applied linear SMILES notation to represent structures of molecules and use generative RNN similar to that have been used in Natural language processing (NLP). SMILES strings were tokenized at the character level. This approach is now known as a Character-based variational autoencoder (CVAE). Generation of molecules using SMILES notation was also used for the prediction of outcomes of organic reactions Lee et al. [2019], Schwaller et al. [2017].

However, there is a fundamental problem with SMILES notation that molecules generated can be incorrect not only chemically but syntactically. Due to this reason the percentage of sampled SMILES by Character-based variational autoencoder was below 1%. To tackle this problem Kusner et al. [2017] proposed the Grammar variational autoencoder (GVAE) . This model is based on formal grammar of SMILES language and allows researchers to generate only SMILES sequences that satisfy the grammar. This approach notably reduces the number of invalid molecules up

Level	Typical examples		Remedies
Syntax	CC(CC	Unmatched parenthesis	GVAE, SD-VAE
	C=#CC	Two bonds	
Semantic	C1CC	Non-closed ring	Graph grammars, JT-VAE
	CBrC	Wrong valency	
Chemical	C1=CC=C1	Energy >0	Graph grammars*

Figure 6-1: Different layers of SMILES invalidity * – graph grammar can solve the chemical invalidity partially due to the restriction of possible rules

to around 7%, however many chemically incorrect molecules occur and common patterns are mismatched rings numbers and wrong valency of atoms. To solve this problem several approaches were proposed. Dai et al. [2018] proposed Syntax-directed variational autoencoder (SD-VAE): with is based on Grammar variational autoencoder adding stochastic lazy attributes. This approach allows checking the syntax validity of a molecule during the generation process. Jin et al. [2018] proposed Junction-tree variational autoencoder (JT-VAE) an approach under which only the junction tree of a molecule is constructed, and the molecule is generated from the nodes of the junction tree by another neural network. The main advantage of this approach is the possibility to generate 100% of valid molecules; however, it possesses quite complicated architecture. Kajino [2018] proposed a Hyperegde replacement grammars (HRG), which are based on hypergraphs, for molecular generation. His implementation demonstrated 100% validity of generated molecules, however, the concepts of hypergraphs and hyperedge replacement grammars are hard to work with.

A fresh idea was proposed in the work OBoyle and Dalke [2018]. The authors of DeepSMILES revised the SMILES notation and implemented two features. First, they are excluded open parentheses to solve the problem of unbalanced parentheses. Second, they use only one number to denote rings. These differences solve the typical syntax SMILES generation problems: unbalanced parentheses and mismatching ring numbers.[1].

[1]Everyone who ran the training of a model on SMILES knows that, when the model is under-

Motivation

Above, we discussed the problems for graph generation by neural networks. Mostly it is related to the fact that the NLP approaches are not convenient in the chemical domain. Formulas of organic compounds are more structured than text, but Recurrent neural networks do not have a mechanism to implement these structural restrictions. From our perspective, other approaches that we mentioned before either provide low performance or over-engineered and extremely hard to follow. A convenient and straightforward process is required to make a step towards the efficient sampling of chemical compounds. The similar approaches are in Natural language processing. There are two types of language models – character-level language models – that generate text char by char and word-level models that produce text from a pre-defined dictionary. Character-level language models can create any word, even unknown. Still, the training of these models is a challenge because they have to learn the vocabulary and syntax of a language implicitly. Word-level language models provide better results but are not able to generate words that are out of a dictionary[2]. SMILES language is similar to character-level notation. Our graph grammar approach resembles word-level modeling, where "words" are chemical subgraphs. But, in contrast to Natural language processing models, our model guarantees the syntactic validity of molecules.

6.1 Formal Definition of Molecular Grammars

In chemoinformatics, organic molecules can be represented as colored weighted undirected graphs. The color of a node represents a set of chemical properties of the node (atom symbol, hybridization state, valency, etc.), and the weights of the edges correspond to chemical bonds. Given this formalism, one can regard the process of generating a molecule as a sequential application of graph rewriting operations. Graph rewriting (or graph transformations) are commonly used in computer sci-

trained, it generates a long list of unbalanced parentheses. DeepSMILES notation looks quite the same! From the author's view, DeepSMILES ideally fit the well-known paradigm *if you can't beat them, join them!*

[2]We consider only vanilla character-level and word-level language models. In practice, there are hybrid alternatives that solve this issue

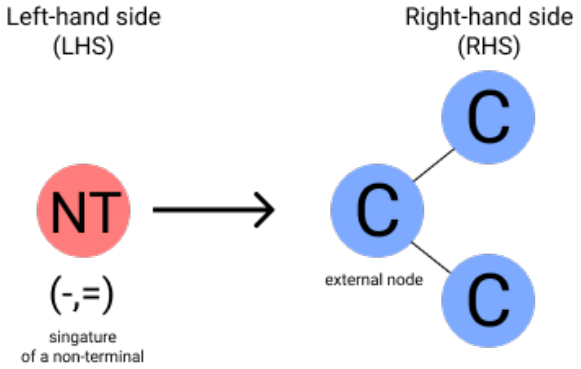


Figure 6-2: Here is the representation of a production rule. Each rule consists of two parts: Left-hand side (LHS) part with only one non-terminal and Left-hand side (LHS) with only one external node. In RHS a non-terminal also can occur, and even more, it can be an external node.

ence. Heckel [2006]. *A graph grammar* is a generalization of grammar from formal language theory. Commonly, graph grammars are divided into two categories: Hyperegde replacement grammars (HRG), which operate on hypergraphs instead of graphs, and Node-label controlled graph grammars (NLC grammars). In HRG rules define how one can replace a hyperedge by a graph. NLC grammars provide a formalism for replacing nodes by graphs. Because of the ambiguousness of the replacement procedure, there is more than one variant of node replacement grammars. Rewriting rules in these grammars are represented as $N \longrightarrow S/E$ where N is a non-terminal S is a subgraph to replace and E is an *embedding rule*. The necessity of embedding rules occurs because, contrary to strings, there are many possible ways of connecting the subgraph S to the rest of the graph. These rules are, in fact, the instructions on how to perform these operations. NLC grammars are the simplest case of graph grammars. In NLC grammars, the replacement process is completely local, and the embedding rules describe only the mechanism of connecting a specific node in a S graph to the neighborhood of the non-terminal N. Our definition of molecular grammar is given below. A molecular grammar is a tuple (N, Σ, P, S^g) where are:

- a finite set N_s of non-terminals, defined as Non-terminal label with a signature $\in S^g$
- a finite set Σ of terminal graphs, with one and only one external node. An

external node has a signature $\in S^g$

- a finite set P of production rules
- a finite set S^g of signatures.

We do not define *embedding rules* explicitly. Our embedding rule is just a replacement of a non-terminal by an external node from another rule (gluing two rules together). S^g solely determines which non-terminal can be replaced. A signature S^g is a set of chemical bonds. This set provides the possibility of replacing a non-terminal with an external node without losing the chemical valency and violating the chemical laws. Using the analogy from formal language theory, our grammar is a "context-free" grammar, which means that there is one and only one non-terminal on the left-hand side of a rule.

6.2 Implementation of Molecular Grammars

In section 6.1, we gave a formal definition of our molecular grammar and inference process. However, our technical implementation of molecular grammar uses a simplified representation of rules. Because LHS of any rule consists of only one non-terminal, we can ignore this part in our implementation. We represent rules as graphs (we use igraph python package), where a node can be either terminal (atom) or Non-terminal (a node that can be replaced). A rule without non-terminals represents a molecule and can be converted into a RDkit molecule or SMILES string. Two compatible rules can be combined, forming a new rule. We have three node types in our framework:

- – an external node
- – a non-terminal node
- – a common node

The replacement process can be explained as a key-lock analogy. Each rule has one and only one *external* node (a key). A *singature* S^g of a rule is a set of incoming connected bonds.

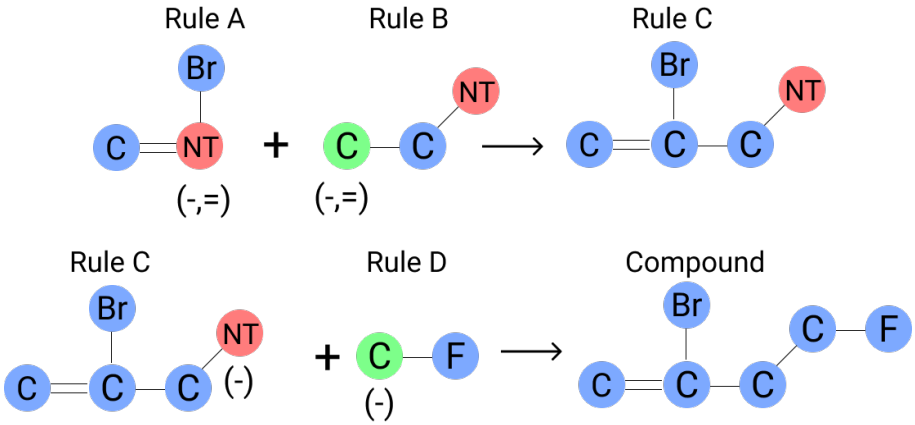


Figure 6-3: The scheme of inference process: (-,=) – is a common signature of non-terminal (NT,rule A) and external node, (carbon atom, rule B). (-) – is a common signature for Rules C and D, respectively.
● – is an external node, ● – is a non-terminal node, ● – is an ordinary node

For example: a signature **(=,-)** means that an external atom can replace a non-terminal (a lock) with the same signature. The combination of rules (A + B) gives a new rule (C) Figure 6-3. Further combination of rules C and D results in a molecular graph without non-terminals. At this stage, the inference is finished and we have a valid molecular structure. Given a set of rules, one can decompose a molecule in a sequence of rules i.e.: **[Rule A, Rule B, Rule C, Rule D]**. It can be regarded as a *linear notation* for molecules. One obtains a molecule at the final stage of folding rules.

Dataset

Gómez-Bombarelli et al. [2018] collected a dataset for training and validation of generative models Kusner et al. [2017], Dai et al. [2018], Jin et al. [2018]. Because it consist about 250 000 molecules we refer it as *250k* dataset for short. We used this dataset for the experiments with Legogram.

Molecules encoding

First, a molecule is decomposed into a sequence of rules. This operation is deterministic and can be done for each molecule individually (the result of initial decomposition does not depend on any other molecules). Using the analogy of fragment-based methods, this stage resembles the decomposition of a molecule to fragments. At

this stage the algorithm transforms a molecule from *rkdit* object to *igraph* object, where nodes are atoms and edges are bonds. Then we rank atoms by RDkit atom ranking. One can use any ranking scheme, but it is convenient to utilize canonical RDkit ranking. The algorithm traverse atoms in order from low-rank to high. In the beginning, a stack E is initialized, and one puts the first atom into the stack. The algorithm loops until the stack is not empty. There are two possible processing routes: working with trees and processing of cycles. We denote an atom that is currently been processed as A^{curr}, the atom that was processed at the previous stage as A^{prev} and the neighbor lowest-rank atom which would be processed at the next step as A^{next}. Processing the A^{curr}, the algorithm forms a new rule R^{curr}, adds A^{curr} to R^{curr}, and denotes this atom as "external" with the signature[3] S^A_{prev} of a non-terminal node from a rule R^{prev}. After that, the algorithm pushes into the stack E all neighbor atoms, goes to A_{next}, and adds to the rule R^{curr} a non-terminal node with the signature S^A_{next}. So, the rule R^{curr} has one external atom with the signature that matches with the non-terminal from R^{prev} and a non-terminal node which matches with the external atom A^{next} from the rule R^{next}. The algorithm will create R^{next} at the next stage. One can regard this process as Breadth-First Search (BFS) which cut molecules atomic-wise but keeping chemical bonds as signatures in non-terminal and external nodes. This process repeats until the algorithm reaches a cycle. Then, the algorithm creates a rule with a maximal ring representing this cycle (for example in the naphthalene system the algorithm takes the largest 10-atomic cycle), and processes this cycle scaffold as a new rule. In this rule, atoms are replaced with non-terminals. Then, the algorithm for the trees processing runs on the cycle skeleton. This procedure provides the ability to encode any organic structure. After encoding of all molecules in a dataset, a folding operation runs. The same rules grouped together and obtain an ID in the rules database. So the list of IDs represent a molecule in Legogram representation. At this stage, one can optionally perform the grammar compression procedure (described in the paragraph "Grammar Compression").

[3]We remind that a signature is a connectivity pattern. For example, an atom with one single and one double bond has the signature (-,=), an atom with two single bonds: (-,-), etc.

Molecules decoding

Decoding is a transformation of a list of rules into a molecular graph. First, we put the first rule in a variable G which corresponds to the growing molecular graph. Then we match the signature of the external node in the following rule (G_i) with one of the non-terminals in the G. If there are more than one non-terminals in G we do it in accordance with the order (as we mentioned in subsection 6.2 our non-terminals are ordered). After that, we add rule G_i into G and replace corresponding non-terminal with the external node of former G_i After that we recalculate S for a new G and process to the rest of the list. After processing the last rule G_n we will have a graph G without non-terminals, thus it can be regarded as a molecule (and converted back to an internal RDKit object or to a SMILES string). The process can be regarded as folding over a list of rules. Figure 6-3 gives an idea of how the inference process works for a simple case (if we assume that "rule C" is G).

Restricted Stochastic decoding

In Subsection "Molecules decoding" we explained the basic decoding procedure. However, with neural networks, we use a *restricted stochastic decoding* technique. The basic idea of our approach is based on the fact that neural networks provide probabilities for each grammar rule at t generation step – p_t. Because we know in advance which rules are compatible, we can mask invalid rules using the multiplication of the logits to the mask. The masks restrict the generation of invalid molecules. One can see the Algorithm 1. The algorithm requires a function $F(previous\_rule)$. Typically, this function is an RNN network that obtains a previous token and generates logits for the next one. We use multidimensional sampling to sample rules in a batch. *calc_ mask* is a function that calculates which rules from a set are comparable

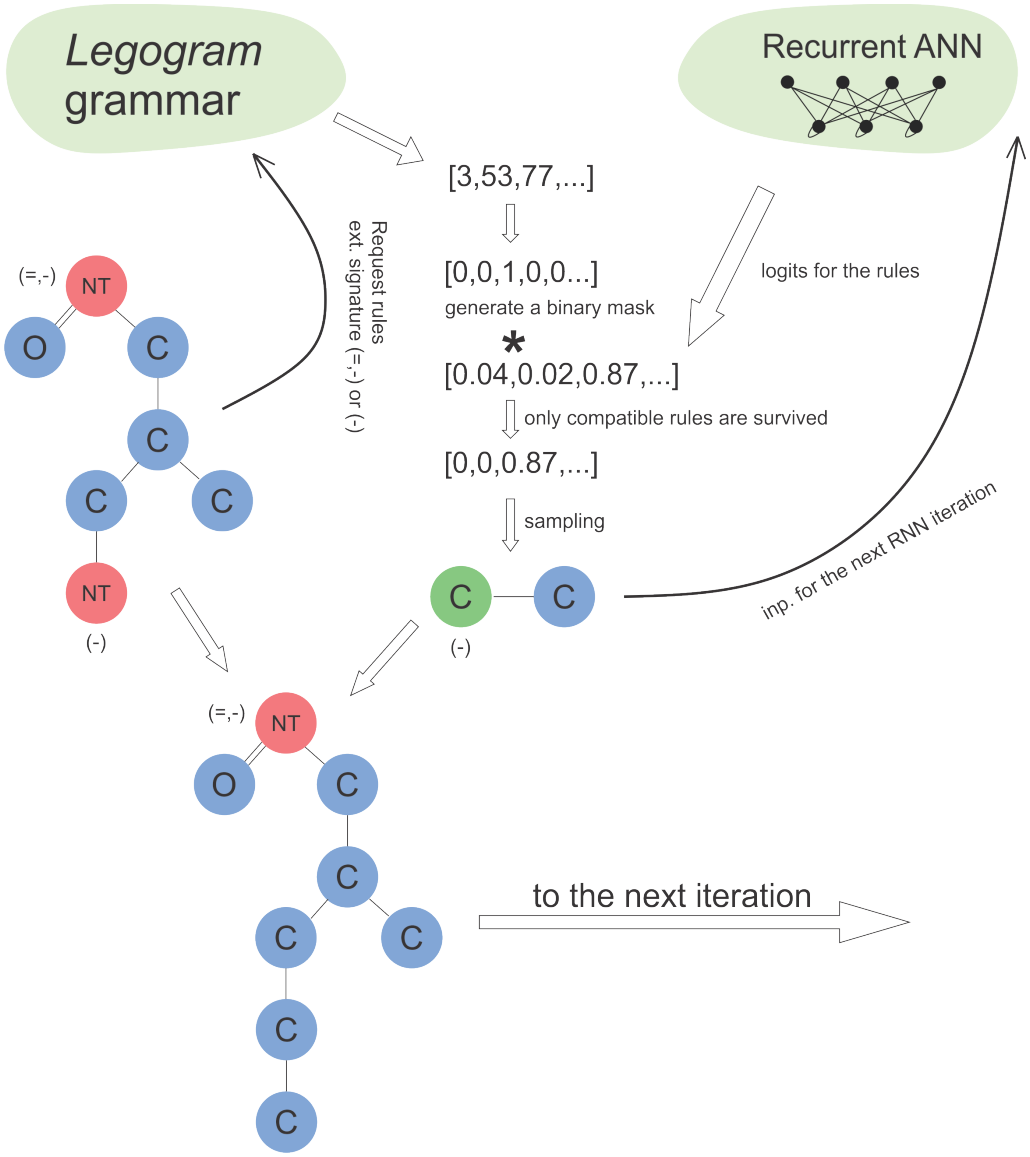


Figure 6-4: A scheme of restricted stochastic decoding

with the rule under consideration.

Algorithm 1: Restricted stochastic decoding

Result: a sampled molecule

Input : a function $F(previous\_rule)$ that returns logits for the next stage, G – is a molecular grammar, T is a temperature for sampling

Output: a molecular graph M

M = Graph($\varnothing$) //*Init. Empty graph*

$\text{logits}_{t0} = F(\varnothing)$

first_rule = sample(logits_{t0}, T)

M.add(first_rule);

while *(get_nonterminals(M)* $\notin \varnothing$) **do**

$mask = calc\_mask(M, \forall\ G.rules)$

$masked\_logits = logits_{t_i} * mask$

$next\_rule = sample(masked\_logits,\ T)$;

$M.add(next\_rule)$;

$logits_{t+1} = F(next\_rule)$

end while

return M

6.3 Validation of the Algorithm

We used a dataset of organic compounds from Gómez-Bombarelli et al. [2018] that consist of about 250k of structures extracted from ZINC database. Authors of this dataset choose compounds only with organic atoms, the second condition was been correctly processed by *RDkit*. The following rule should be satisfied for all molecules in a dataset to check the correctness of the implementation:

$$m = Decode^G(Encode^G(m)) \tag{6.1}$$

where M is a set of molecules been used for the construction of the grammar G. m – is a molecule (or molecular graph). *Encode* – is the encoding function, *Decode* – is the decoding function. This formula represents a simple idea that each struc-

ture, after encoding and decoding, should be the same. We performed this test for our dataset, and it has successfully passed for all molecules in the dataset. It should be noted here that the current version of Legogram can not guarantee the correct reconstruction of stereoisomers (however the generation of stereo-compounds is possible).

6.4 Grammar Compression

As it was mentioned before, one of our motivations was to create a molecular representation with the reduced mean length. SMILES notation was designed as a balance between computational processing and humans readability. That is why SMILES notation is redundant. To compress the molecular representation more, we implemented an optimization algorithm. The idea of the algorithm is based on the fact that one can combine rules to form new rules. At the preparation step, our algorithm searches the frequent substrings in encoded representations of molecules and groups them into new rules. On encoding stage, the algorithm searches these substrings and replace them into additional rules. This idea resembles popular text compression algorithms, but regards the compatibility of rules. Because, not every linear rule sequence can be grouped due to possible violations, the algorithm analyse these sequences before grouping. This analysis is time costly, and we put the sequences that have already been analysed in a cache.

Commonly Legogram uses a substantially larger dictionary, so the grammar rules describe more general chemical fragments, rather than atomic symbols in SMILES. Due to this fact, Legogram notations one can expect for Legogram notations to be more compressed. To explore it, we performed an experiment. We calculated the distributions of lengths for two types of SMILES notations, for the unoptimized Legogram grammar and for the optimized Legogram grammar. One should mention that there are two possible types of SMILES tokenization: *character based* and *regular expression* based. In character-based tokenization each character in SMILES string is represented as a token. It is the simplest way, however, it leads to chemically related issuers: for example, a chlorine atom *Cl* is represented by two tokens: *C*

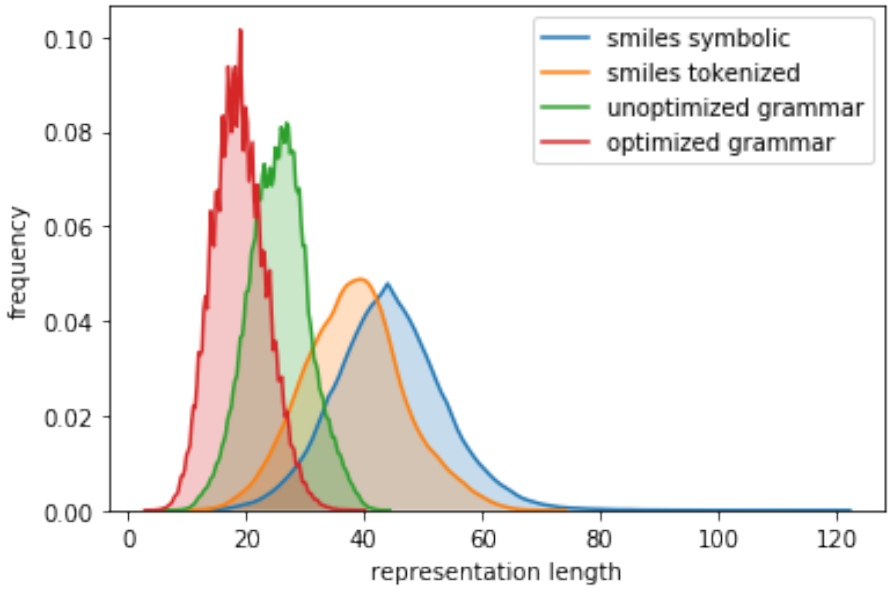


Figure 6-5: The distribution of different molecular representations. One can see that even unoptimized grammars can notably reduce the mean length of the dataset. Optimization of the grammar leads to better results.

and l, where the first one means carbon atom in SMILES and the second one has no meaning. To overcome this issue the *regular expression* tokenization can be used. A SMILES string is parsed by a regular expression. In this case, atoms like chlorine and common groups like [NH2+] are processed as one token.

In Figure 6-5 one can see that even unoptimized grammars can notably reduce the mean length of the dataset. Further optimization of this grammar can improve the compression substantially. As we discussed earlier, the smaller length can combat a "one symbol failure". The probability of the correct generation of the entire compound is a multiplication of probabilities for each symbol. Let's assume that the probability for 1 token is 0.99 The median for SMILES is around 40 tokens, for optimized grammar is around 19. The probability of the correct generation of the entire sequence would be: for SMILES $0.99^{40} = 0.66$ and for grammars $0.99^{19} = 0.82$ This simple example demonstrates the importance of proper representation for the quality of structure generation. We can also speculate, that molecular grammars can be a feasible tool for storing extra-large chemical databases. General lossless compression algorithms, such as Lempel–Ziv–Welch (LZW) Welch [1984] operate with byte strings and do not regard the internal structure of compressed objects. In con-

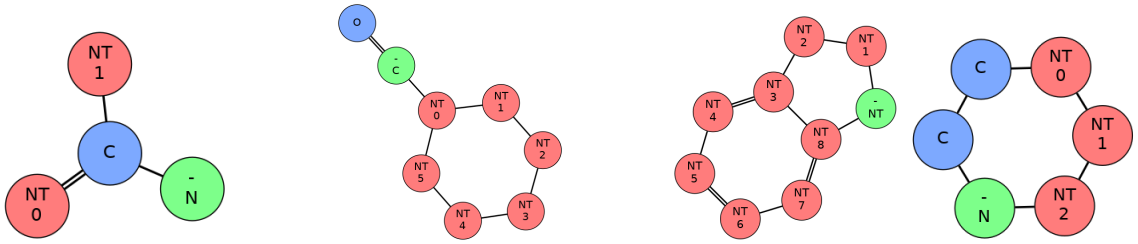


Figure 6-6: Frequent representative rules from the optimized grammar. ● – is an external node (a signature is on the top of atom symbol) , ● – is a non-terminal node (a number is the rank), ● – is an ordinary node

trast, our algorithm provides compression with respect to the chemical structures. We hope that the co-use of lossless compression algorithms to structures encoded by molecular grammars can strongly compress chemical data. But additional experiments are needed to support this statement.

We analyzed rules generated after grammar optimization and draw some representative ones in Figure 6-6. One can note that these rules have chemical meaning. For example, some groups correspond to precursors for well-known chemical groups, i.e., amino-group, cyclo-aldehyde derivatives, heterocyclic compounds. It is worth mentioning that original non-cyclic rules can have only one atom. That means that the algorithm can successfully combine several rules to derive a chemically valid (and well-known for chemists) fragment as a new rule. So, we can speculate that this algorithm can infer chemical knowledge directly from the data.

We would like to stress that molecular grammars provide the considerable reduction of representation of structures; theoretically, it can help to achieve better performance in structure generation problems.

6.5 Generative Models

One of the most important applications for Legogram is the *de-novo* generation of chemical compounds with desired properties. We followed the approach to generative Reinforcement learning models from Olivecrona et al. [2017]. Under this approach, two neural networks with the same architecture are used. The first one is a policy network. This network is trained to generate structures that resembles the training set (*Prior* network). The second one is the *Agent* network. This network

generates molecular structures biased towards the desired properties. A used-defined scoring function $S(A)$ is used to control the generation. Following the traditional Reinforcement learning notation, a sequence of grammar rules (in original parer a set of SMILES tokens) is a sequence of actions $A = a_0 a_1 ...$ The likelihood of a generated sequence is represented by the Equation 6.2.

$$P(A) = \prod_{t=1}^{T} \pi(a_t | s_t) \tag{6.2}$$

The authors of REINVENT proposed an augmented likelihood:

$$\text{LogP}(A)_U = \text{LogP}(A)_{Prior} + \sigma S(A) \tag{6.3}$$

Where σ is a scalar coefficient. This coefficient allows to balance between the quality of molecular structures and $S(A)$. In REINVENT the loss function is:

$$G(A) = -[\text{LogP}(A)_U - \text{LogP}(A)_A]^2 \tag{6.4}$$

where $\text{LogP}(A)_A]^2$ is a *Agent* likelihood. Optimizing of *Agent* network with this loss function leads to generating compounds with desired properties, as it was showed by Olivecrona et al. [2017].

6.5.1 Legogram-based Generative Modeling

To validate our idea of an faultless generation of chemical compounds, we trained a classical Recurrent neural network model and compared the ratio of correct molecules for grammar and SMILES models. The architecture and training parameters of the grammar model were the same as *Prior* model from REINVENT. We regarded a molecule to be valid if RDKit can process it. We sampled 128 structures, every 10 iterations. The result of this experiment is given in Figure 6-7. One can see that our grammar model can produce valid molecules after the first 20 iterations. For SMILES model, we used the original REINVENT implementation from GitHub. The quality of the SMILES model at the same iterations is much lower. Even to the

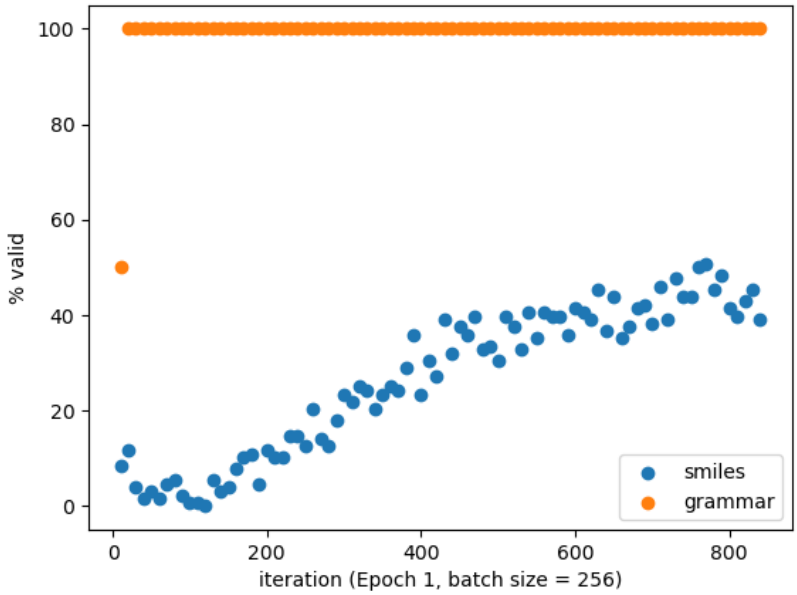
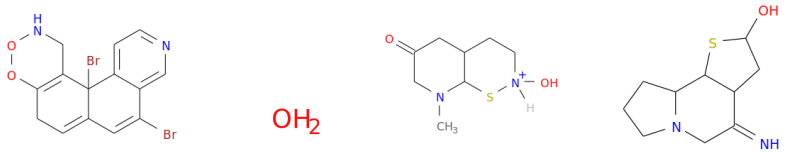


Figure 6-7: The comparison of grammar and SMILES models on the generation of chemical structures

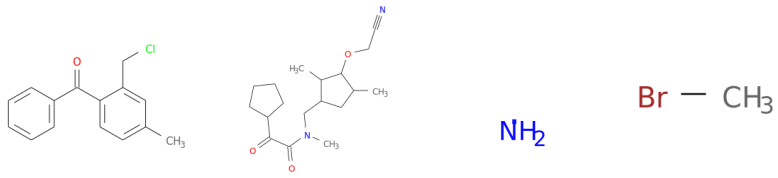
end of the first epoch, the SMILES model is capable of generating only 50 % of valid structures. However, we believe that the performance of SMILES generative models can be notably higher. The authors of ReLeaSE (Reinforcement Learning for Structural Evolution) Popova et al. [2018] claimed that about 95% of generated molecules are valid. The authors of REINVENT Olivecrona et al. [2017] noted that the *Prior* network can generate 94% of correct molecules. Both ReLeaSE and REINVENT networks were trained on the significantly large ChEMBL dataset, and, for example, for REINVENT, it requires several hours to finish *Prior* training. Nevertheless, the ability of Legogram to produce absolutely correct molecules, that looks quite "chemically" just after a few seconds of training is quite impressive.

6.5.2 Optimization of Drug-likeness

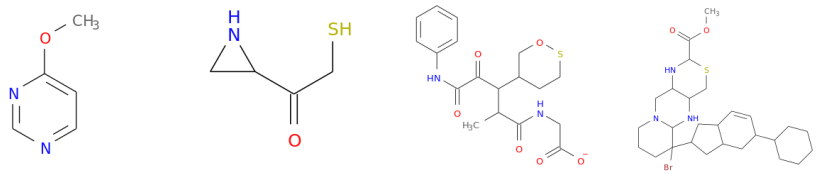
Because the generation of chemical structures, even with absolute validity, is not useful itself, we experimented the optimization of a molecular property. We decided to generate drug-like molecules. To achieve it, we used a well-known Quantitative Estimate of Druglikeness (QED) index, developed by Bickerton et al. [2012]. We



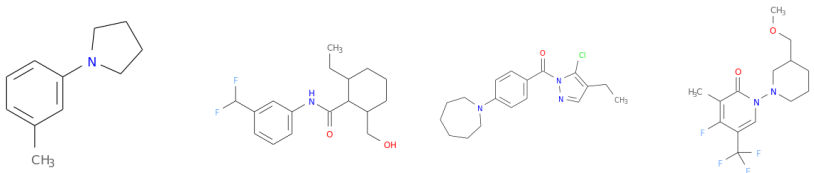
Iteration 20



Iteration 50



Iteration 100



Iteration 500

Figure 6-8: Examples of compounds that are generated at the first iterations.

utilized a QED implementation from RDKit. *Prior* network was trained on *250k kekulized* dataset for 5 epochs. We trained *Agent* network for 1000 iterations, generating 64 molecules for each iteration. We calculated statistical parameters: median and standard deviation of the distribution. We also built histograms of the distributions for each iteration. The number of unique generated structures during *Agent* training was: 99.98% The top-9 generated structures with maximal QED value are given on Figure 6-9. Our model reached the highest QED value of 0.95 One can see on Figure 6-10 that the QED distribution is biased towards high QED values at the end of the training, so the optimized *Agent* produces compounds with higher QED index.

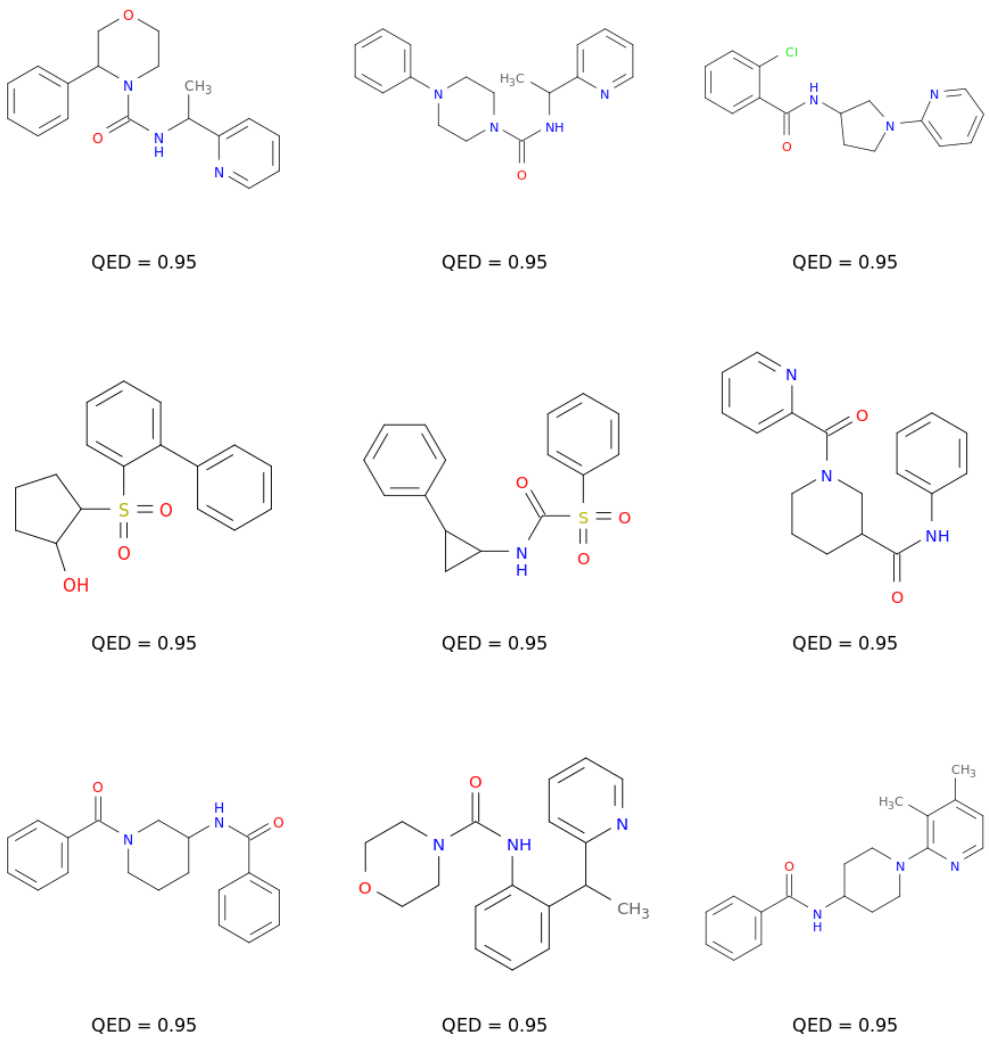


Figure 6-9: Top-9 QED molecules generated by *Agent* with Legogram *250k kekulized* model

6.5.3 Synthetic Accessibility

Compounds must be synthetically accessible to provide interest for chemists. There are some machine-learning methods for the quantitative assessment of synthetic complexity: SAscore Ertl and Schuffenhauer [2009], SCSCore Coley et al. [2018], and SYBA Voršilák et al. [2020]. To estimate the synthetic accessibility of the compounds generated by *Agent* network, we decided to use SYBA method. This approach predicts a compound to be either easy-to-synthesize (ES) or hard-to-synthesize (HS). If SYBA score is positive, a compound regarded as been easy-to-synthesize; otherwise,

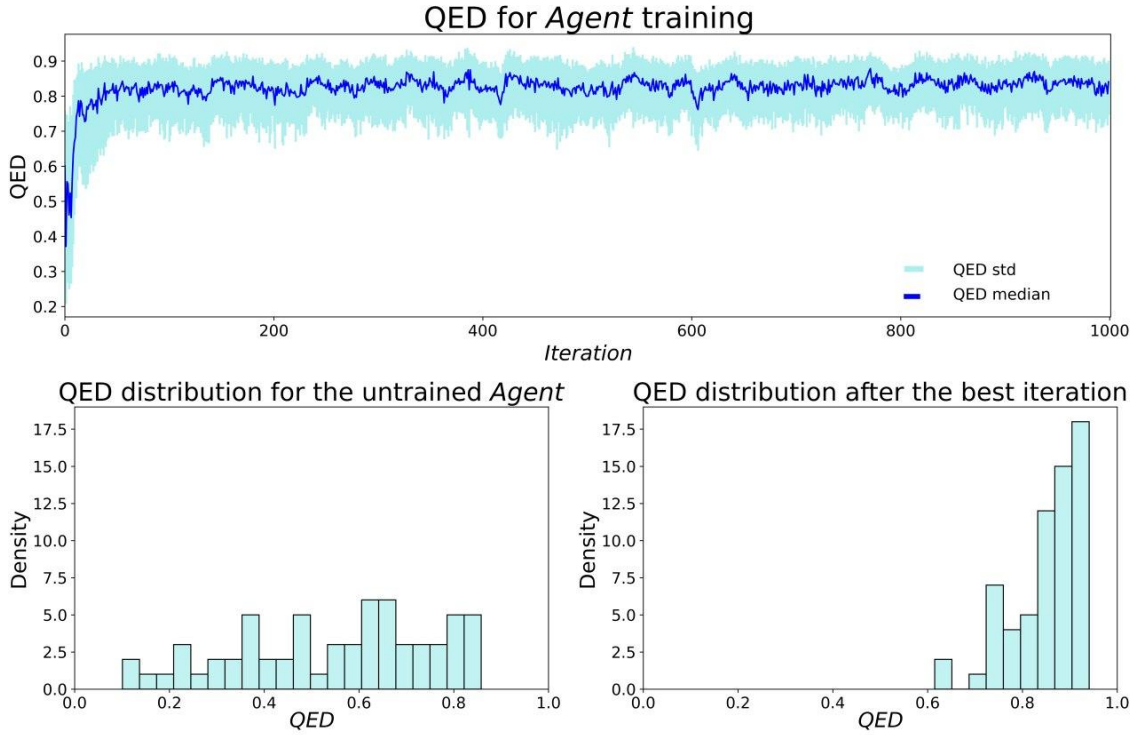


Figure 6-10: Generation performance for drug-like compounds by reinforcement learning and Legogram. *Top:* QED median and standard deviation during *Agent* training, *Bottom:* the shift of the distribution of generated molecules towards higher QED values after training.

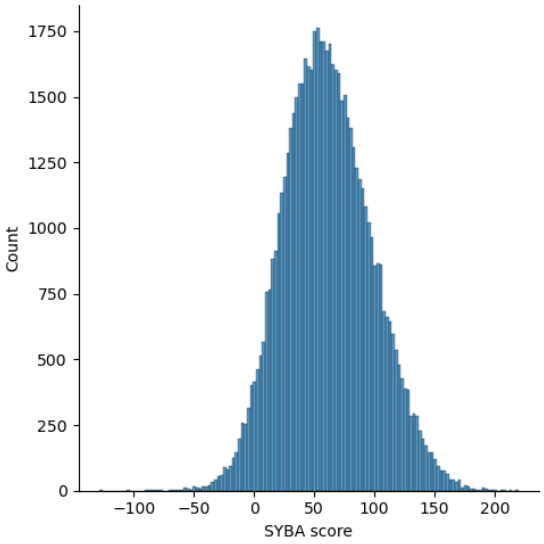


Figure 6-11: SYBA scores for the compounds generated during QED optimization

if the score is negative, a compound regarded as been hard-to-synthesize. This score varies from $-\infty$ to ∞; however, as the authors mentioned, it is within $(-100, 100)$ range for the majority of compounds. The authors consider the score as the confidence of prognosis. We calculated SYBA scores for all compounds generated by *Agent* network during QED optimization. This histogram is provided in Figure 6-11. One can see that the majority of compounds were considered easy-to-synthesize with a median value of 59.

6.6 Sampling Compounds from Chemical Space

Systematic exploration of some regions of chemical space is a typical problem in drug discovery. In Chapter 5, we proposed a method for the projecting of chemical compounds to 2D coordinates for the visualization of chemical space. Now, when we have a technique and a tool for the visual analysis of the chemical space, it's worth focusing on the sampling of chemical compounds directly from the desired regions of chemical space. The pipeline of our method is based on REINVENT approach, which we described above. Our solution allows generating both compounds that are drug-like and lies close to the regions of interest in the chemical space.

The scoring function definition

We use a scoring function $S(m) \in [0, 1]$ that describes the alignment of a molecule to our pre-defined conditions:

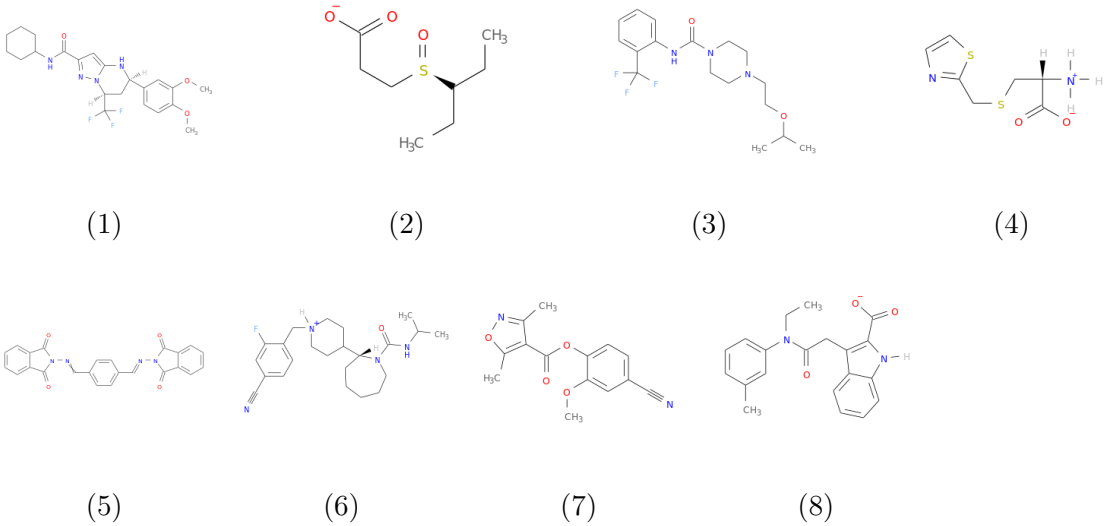
$$S(m) = \frac{M(m) + wG(m)}{N_m + w} \tag{6.5}$$

The scoring function $S(m)$ consists of two parts: drug-likeness $M(m)$ and a distance function $G(m)$. In this formula, N_m is several parameters of the drug-likeness function, and w is a weight of the distance function. In our case $N_m = 8$. The denominator of this equation is a normalizing factor. Fitting w, one can balance the drug-likeness of generating compounds *versus* and the proximity to the desired region of chemical space. The list of parameters of the drug-likeness function is given in Table 8.4.

Choosing a distance function $G(m)$ is quite tricky because there are many types of distances[4]. We propose two types of distance functions: a boxcar function and a flat-top Gaussian function. The idea of a boxcar function is to indicate whether the generated compound is inside a region surrounding a molecule (or point). It is the most straightforward way; however, this function is not smooth. The authors of REINVENT proved that this approach could potentially work with indicator functions. They showed the example of sampling new compounds avoiding sulfur successfully. But our preliminary experiments revealed that sometimes the training is unstable. To combat this problem, we proposed a flat-top Gaussian function: a boxcar function merged with Gaussian function at a distance R_{cut} The formal definition of boxcar function G^B is:

$$G^B = \begin{cases} 1 & \text{if } d(p_i, p_s) < R_{cut} \\ 0 & otherwise \end{cases} \tag{6.6}$$

[4]strictly speaking, the functions that we will discuss are not distances in the mathematical sense because they do not satisfy three laws of distance functions. Still, it is convenient to take it in common sense



(1) (2) (3) (4)

(5) (6) (7) (8)

Figure 6-12: Reference structures: 1-8

A flat-top Gaussian (G^{FT}) function is defined as:

$$G^{FT}(r, eps) = \begin{cases} 1 & \text{if } d(p_i, p_s) < R_{cut} \\ e^{-(d(p_i,p_s)/eps)^2} & otherwise \end{cases} \tag{6.7}$$

In these equations d is the euclidean distance between a point p_i and the starting point p_s.

Sampling of compounds with different distance functions

We selected 8 reference molecules from *250k* dataset to cover the local chemical space. Besides, we manually chose 3 points that do not correspond to molecules from the dataset (points 9-11).

We trained *Agent* models with different $G(m)$ and the corresponding parameters. Using these models, we sampled 64 molecules from each point and calculated the mean values of $S(m)$. The results of these experiments are given in Table 6.1. One can see that for all points, Flat-top Gaussian models provide good quality. It is not the case for a Boxcar model; for example, for points **(5)** and **(6)**, the mean score of generated molecules is much lower than for the original structure. It is noteworthy that for point **(2)**, the score of the original molecule is relatively low because the

molecule is small, polar, and do not have rings. Moreover, this point is far away from the main distribution. Nevertheless, the scores of generated compounds are higher than for the original molecule. The distribution of generated molecules is given in Figure 8-2. There is an unusual behavior of a model; when it is hard for the model to generate the compounds, it samples ones from the most drug-like region (see, points **(5,6,9,10,11)** at the first column). We named this type of failure as *breakdown*. As we will show later, *breakdowns* are common if the point is located in an uninhabited region of chemical space.

Grid search over the whole chemical space of *250k* dataset

To analyze the behavior of our models and their robustness on the full dataset, we performed a grid search over the whole space of *250k* dataset. We iterated over each dimension from point -45 to 45 by step=10. We used a Flat-top Gaussian with parameters: $R_{cut} = 3$, $eps = 20$, $w = 10$. The results are given in Table 6.2. From this data, it is clear that the quality of sampling is much better in the inhabited region of chemical space, that is located (in our case) close to the geometrical center of the space. The visualization of sampled compounds for each point is given in a voluminous Figure 6-14. Sampling from uninhabited regions at the edges of the space results in *breakdowns*. Models either sample from the drug-like regions, or sample from the nearest cluster. A possible explanation is based on the fact that t-SNE method, which is used as the back-end for our *Agent* does not perceive the global relations in the chemical space.

We should raise a question here: does our model follow a global chemical space? t-SNE is a method that keeps the local distances between points[5] and there are no chances for it to follow the global space. However, the parametric t-SNE uses Artificial neural network to map compounds, and one can expect that the global chemical space would be learned during the optimization of local chemical regions. Unfortunately, it was tricky to prove (or reject) this hypothesis without generative modeling, and due to this reason, we have not discussed this problem before. After the systematic study of this dataset, we can prove that our parametric t-SNE model

[5]It is evident even from the name of the method: t-distributed Stochastic **Neighbor** Embedding

does not follow the global chemical space. This fact restricts the ability for parametric t-SNE based generative models to create principally new compounds. The search for new methods for the projection of chemical space preserving its global structure would be the topic of our further research. At the same time, the current method demonstrates the feasibility of our approach, and one can create a useful tool for chemists for the exploration of local regions of chemical space on the base of this approach.

6.7 Conclusions

In the course of our work, we developed a molecular grammar framework that is based on graph grammars. We have shown experimentally that chemical structures can be encoded as graph grammars and decoded back (with exceptions to stereochemistry). We demonstrated that this representation is capable of reducing the mean length of structures in comparison to SMILES up to 35%. We designed a grammar compression algorithm that allows one to additionally decrease the mean size of compound representations up to 2 times from the original SMILES representation. Our main goal was to develop a method for the error-free generation of chemical structures, and we have shown that using the procedure of restricted stochastic decoding and molecular grammars, a recurrent neural network can reach 100% correctly generated molecules. As a showcase, we used Legogram to sample compounds from chemical space. On the base of our parametric t-SNE model, described in the previous chapter, we have illustrated that using Reinforcement Learning and Legogram one can generate molecular structures from local regions of chemical space. We demonstrated the possibility of a focused generation of organic compounds only for some specific cases; however, we believe that it can be used for many other applications, including the generation of compounds with the desired bioactivity. Our confidence is based on the fact that the original REINVENT approach was successfully used to generate DRD2 inhibitors. We think that our approach would also be effective in it. We believe that our Legogram library became a useful tool for generative modeling. The library is available on GitHub:

Table 6.1: Mean scores of 64 sampled compounds for each model. *Agent* network was trained with different functions $G(m)$. R_{cut} for all models is 3

Point	Compound	Score	**Boxcar** (G^B) $w = 5$	**Flat-top** (G^{FT}) $eps = 10,\ w = 5$	$eps = 20,\ w = 8$	$eps = 20,\ w = 10$
1	1	0.87	0.86	0.93	0.92	0.94
2	2	0.62	0.58	0.78	0.82	0.85
3	3	0.87	0.95	0.96	0.98	0.89
4	4	0.62	0.79	0.88	0.91	0.88
5	5	1.0	0.56	0.83	0.81	0.94
6	6	1.0	0.59	0.91	0.94	0.93
7	7	1.0	0.80	0.93	0.92	0.94
8	8	0.87	0.62	0.85	0.92	0.91
9	–	–	0.60	0.58	0.87	0.91
10	–	–	0.59	0.59	0.84	0.93
11	–	–	0.58	0.89	0.94	0.93

Table 6.2: Grid search over full *250k* dataset chemical space with step 10. The values that are > 0.85 are bold

Flat-top Gaussian, eps=20, w=5										
y	-45	-35	-25	-15	-5	5	15	25	35	45
x										
-45	0.43	0.58	0.77	**0.88**	0.84	0.79	0.83	0.84	0.73	0.43
-35	0.44	0.69	**0.90**	**0.95**	**0.93**	**0.88**	**0.87**	**0.87**	0.76	0.41
-25	0.53	0.77	**0.93**	**0.91**	**0.95**	**0.86**	**0.88**	0.80	0.68	0.42
-15	0.51	0.74	**0.9**	**0.95**	**0.91**	0.81	**0.89**	0.83	0.67	0.40
-5	0.75	0.82	**0.9**	**0.88**	0.84	**0.92**	**0.94**	**0.9**	0.86	0.35
5	0.67	0.79	**0.87**	**0.93**	**0.94**	**0.97**	**0.94**	**0.96**	**0.90**	0.42
15	0.61	0.64	0.82	**0.91**	**0.91**	**0.92**	**0.94**	**0.92**	0.84	0.4
25	0.43	0.52	0.67	0.82	**0.85**	**0.88**	**0.89**	**0.91**	0.77	0.41
35	0.41	0.42	0.44	0.80	0.82	**0.89**	0.72	0.71	0.41	0.40
45	0.41	0.38	0.41	0.40	0.42	0.40	0.42	0.42	0.42	0.40

https://github.com/sergsb/LegoGram

Figure 6-13: Modelling with different distance functions (**One can zoom this picture to see the details**)

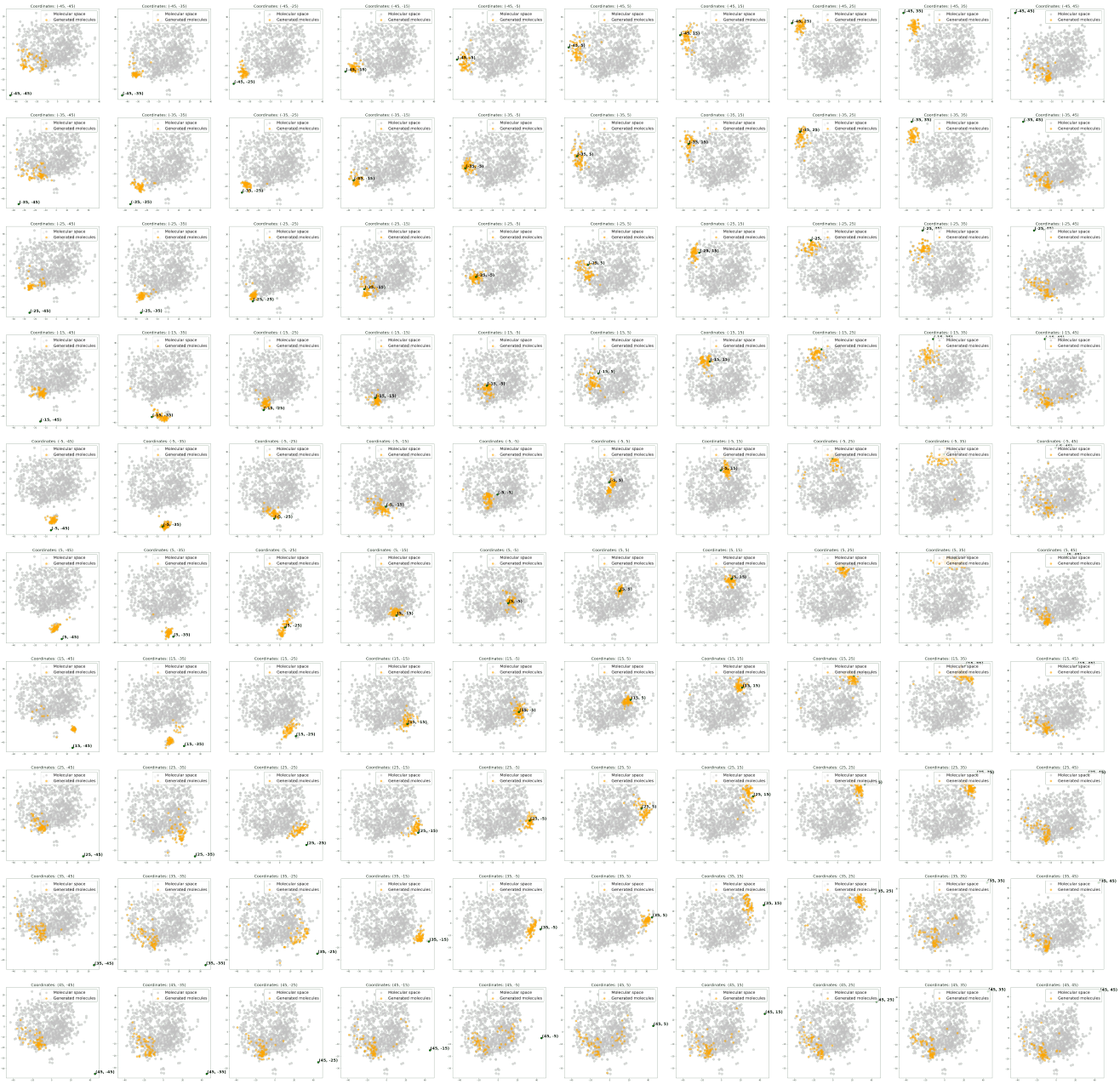


Figure 6-14: Visualisation of the grid search (**One can zoom this picture to see the details**)

Chapter 7

Conclusions

Been extremely large, the chemical space requires special methods and tools for analysis. In this research we developed a number of methods and tools for the exploration of chemical space. We showed that one can calculate the density of solvent sites surrounding a solute by 3D RISM method and use it as spatial descriptors. We proved the possibility of using 3D spatial descriptors and 3D Convolution neural networks for the prediction of properties of molecular structures on the example of Bioconcentration factor (BCF).

We studied the feasibility of Multitask learning (MTL) for the activity modelling on the example of acute toxicity on a broad chemical space. Our experiments revealed that Multitask learning (MTL) learning provides better performance in comparison with Single-task learning (STL) and other machine learning methods. We demonstrated that in the latent space of our models the distribution of compounds in the light of toxicity is not uniform, and there are clusters of toxic and non-toxic compounds.

Been motivated by the potential of visual analysis of chemical space, we studied the methods for visualization of chemical space. We have developed a technique for chemical space visualization guided by neural networks. Our visualization method highlights the well-known chemical rule that the structure of compounds related to their activities, and one can study chemical space for the search for new drugs by our tool. We proved that the distribution of chemical compounds on the 2D space is meaningful by building machine learning models on the top of this distribution

and comparing their performance.

The problem of direct sampling of compounds from chemical space was compounded by the lack of chemical notations suited to use with Recurrent neural networks (RNNs). We addressed this problem by creating a chemical-oriented graph grammar library Legogram. We experimentally proved that this library can represent chemical structures in a compressed way (comparing to SMILES) representation) and can guarantee the generation of correct chemical structures by RNN. We demonstrated that one could sample compounds directly from 2D projections of chemical space, and the generated structures are similar to typical ones in the regions of interest.

We believe that our methods and tools would be useful to speedup the drug-discovery process.

List of Figures

1-1 "Bottom-up" vs "Top-down" approaches 14

2-1 Common QSAR/QSPR pipeline. Molecular structures are converted to vectors of molecular descriptors. Stacked molecular descriptors form a matrix X, and activity/property vector y. Given X and y one can build a model $F(X) = y$ 19

2-2 An example of a very simple decision tree for the binary classification of the developmental toxicity of molecules. $X[i]$ is a binary flag of the presence of certain functional groups in a molecule. This example was made on the real data, but only for the demonstration, and does not correspond to any model mentioned in this thesis. 25

2-3 The scheme of *5-fold* cross-validation procedure. On each fold $\frac{4}{5}$ of a dataset becomes a training set and $\frac{1}{5}$ becomes a test set, sliding over folds. The cross-validation is done based on molecules and thus all toxicity values for the same molecules are within the same set always. 35

3-1 An example of the visualization of the scalar fields for a molecule as 2D slices taken by the principal axis (Left – a visualization of hydrogens density. Right – a visualization of oxygen density. Light yellow color – lower values, pale green color – bulk values, blue color – higher values)) 49

3-2 A general representation of *ActiveNet4* 3D Convolutional ANN 49

3-3 A schematic representation of *ActivNet4* architecture with visualized 2D slices of feature maps on a trained network. Feature maps are colored using the same color scheme as in Figure 3-1. Blue arrows labeled conv N × N × N denote a 3D convolution layer, green arrows labeled pool N × N × N denote 3D max-pulling layer, and red arrow labeled "connected" denotes a fully-connected layer. The figure is based on Figure 4 from Ref. 59 50

3-4 The distributions of the number of conformers for each molecule in the training and test sets 53

4-1 Ions considered to be nontoxic 59

4-2 Representation of endpoints as outputs of a deep neural network. . . 60

4-3 The RTECS chemical space visualization. Each point stands for the one molecular structure and its color indicates the acute toxicity values in log(mol/kg). 62

4-4 Matrices of correlations for endpoints with various thresholds (*min_ samples*) values. The toxicity endpoints demonstrate their correlation notwithstanding the number of compared samples. 65
4-5 Average RMSE of predictions of toxicity for all endpoints in -log(mol/kg) units by different methods and descriptor sets. Descriptors were arranged in accordance with mean values of predictions by all methods (the best are on the left). Methods are ordered by the mean RMSE over all descriptors (*MT_ DNN* and *KNN* demonstrated highest and lowest overall performances, respectively). 67
4-6 Prediction charts for a number of selected endpoints 68
4-7 Representation of endpoints as attributes in STL modelling. The encoding of endpoints as input descriptors allows their simultaneous prediction using neural network with one output neuron. 69
4-8 The results of the application of the t-SNE method to deep features generated by the multitask DNN, values are minus logarithms of maximal endpoint (greater values correspond to larger toxicity). Several clusters with high toxicity are observed. 72

5-1 The schematic workflow of the pT-SNE mapping procedure 79
5-2 The learning curves obtained for different perplexity values 80
5-3 The results of the neural network mapping for a set of GPCR (A) ligands. **A** contains ligands of adenosine A2 (aa2ar), adrenoreceptors β1 (adrb1) and β2 (adrb2), chemokine CXCR4 (cxcr4) and dopamine DR3 (drd3). **B**, **C**, **D**, contains zoomed area from **A**. (Perplexity 100) 81
5-4 The dependence of the the resulting distance on the initial molecular similarity for the TAAR1 data set (Perplexity 100). Points' colors were set according to the density level: yellow means the highest density while magenta indicate the lowest one. 83
5-5 The mapping results for TAAR1 agonists data set (Perplexity 100). Points' colors were set according to the pEC50: yellow means the highest activity density while magenta indicate the lowest one 84

6-1 Different layers of SMILES invalidity * – graph grammar can solve the chemical invalidity partially due to the restriction of possible rules 89
6-2 Here is the representation of a production rule. Each rule consists of two parts: Left-hand side (LHS) part with only one non-terminal and Left-hand side (LHS) with only one external node. In RHS a non-terminal also can occur, and even more, it can be an external node. 91
6-3 The scheme of inference process: (-,=) – is a common signature of non-terminal (NT,rule A) and external node, (carbon atom, rule B). (-) – is a common signature for Rules C and D, respectively. 93
6-4 A scheme of restricted stochastic decoding 96
6-5 The distribution of different molecular representations. One can see that even unoptimized grammars can notably reduce the mean length of the dataset. Optimization of the grammar leads to better results. . 99
6-6 Frequent representative rules from the optimized grammar. 100

6-7 The comparison of grammar and SMILES models on the generation of chemical structures . . . 102
6-8 Examples of compounds that are generated at the first iterations. . . 103
6-9 Top-9 QED molecules generated by *Agent* with Legogram *250k kekulized* model . . . 104
6-10 Generation performance for drug-like compounds by reinforcement learning and Legogram. *Top:* QED median and standard deviation during *Agent* training, *Bottom:* the shift of the distribution of generated molecules towards higher QED values after training. . . . 105
6-11 SYBA scores for the compounds generated during QED optimization 106
6-12 Reference structures: 1-8 . . . 108
6-13 Modelling with different distance functions (**One can zoom this picture to see the details**) . . . 112
6-14 Visualisation of the grid search (**One can zoom this picture to see the details**) . . . 113

8-2 Chemical space of 1000 random points from *250k* dataset (gray). Points 1-8 are the reference structures. Points 9-11 do not correspond to any molecules. . . . 150

List of Tables

3.1 Accuracies of $\log_{10}$ BCF predictions by different models. RMSE stands for root mean square error, MAE stands for mean absolute error and R denotes Pearson's correlation coefficient. For cross-validadated models the standard deviations have been calculated. 54

4.1 The descriptors used in our experiment. Several descriptor blocks that are indicated by "(3D)" required 3D representation of molecules, which was calculated by using 2D to 3D structure conversion using *Corina* program. 61
4.2 The architecture of *dense7* neural network 63
4.3 The comparison of quality of two consensus attributed models with consensus multi-task model (averaged over all endpoints) 69
4.4 The comparison of RMSE for models based on Feature Net approach with multi and single task models (averaged over all endpoints) . . . 70

5.1 The results of application of the machine learning methods to the initial ECFP6 fingerprints and to the 2D mapped space (multiclass classification) 85

6.1 Mean scores of 64 sampled compounds for each model. *Agent* network was trained with different functions $G(m)$. R_{cut} for all models is 3 . . 111
6.2 Grid search over full *250k* dataset chemical space with step 10. The values that are > 0.85 are bold 111

8.1 The architecture of our encoding ANN for parametric t-SNE projection 150
8.2 The optimal hyperparameters of classifires both found by grid search optimization and default 151
8.3 Sets of parameters for grid search procedure 152
8.4 Parameters of the drug-likeness scoring function $M(m)$. If a compound satisfies a parameter the value of $M(m)$ increases to one, so the score for the compound lies between 0 (total mismatch) and 8 (ideal fit) 152
8.5 Endpoints extracted from RTECS dataset 153

Glossary

3D CoMFA 3D Comparative Molecular Field Analysis. 22

ADME Absorption, Distribution, Metabolism, and Excretion. 13, 37, 83

ANN Artificial neural network. 27–29, 45, 66, 71, 82, 109

API Application programming interface. 23

ASNN Associative neural networks. 66

AUC Area under curve. 34, 41

BCF Bioconcentration factor. 14, 15, 45–47, 52, 55, 114

BFS Breadth-First Search. 94

BPMF Bayesian Probabilistic Matrix Factorisation. 42

CAS Chemical Abstracts Service. 58

CDK Chemistry development kit. 22, 23

CNN Convolutional neural network. 14, 15, 28, 45, 49

CoMSIA Comparative molecular similarity indices analysis. 22

CVAE Character-based variational autoencoder. 88

DNN Deep neural network. 40, 41, 60, 64, 72, 117

DUDe A Database of Useful Decoys. 76, 77, 84, 85

ECFP Extended Connectivity Fingerprint. 77, 81

EPA Environmental Protection Agency. 47, 53

FN False negative. 33

FP False positive. 33

GPCR G-protein-coupled receptors. 78, 81, 84–86, 117

GVAE Grammar variational autoencoder. 88, 89

HRG Hyperegde replacement grammars. 89, 91

HTS High Throughput Screening. 37

JT-VAE Junction-tree variational autoencoder. 89

LHS Left-hand side. 91, 92, 117

LSTM Long Short-Term Memory. 28

MAE Mean Absolute Error. 33, 71

MDFT Molecular density functional theory. 44

MDS Multidimensional scaling. 82, 85–87

ML Machine learning. 13, 18, 24, 32, 35, 38

MM Molecular Mechanics. 11, 12

MTL Multitask learning. 31, 38–41, 114

NLC grammars Node-label controlled graph grammars. 91

NLP Natural language processing. 88, 90

OCHEM Online chemical modeling environment. 15, 21, 23, 60

OECD Organisation for Economic Cooperation and Development. 46, 73

PCA Principal component analysis. 76, 77, 81, 82, 85–87

PLS Partial least squares. 22, 42

pT-SNE Parametric t-distributed Stochastic Neighbor Embedding. 79, 85, 86, 117

QED Quantitative Estimate of Druglikeness. 102, 103, 105, 118

QM Quantum Mechanics. 10, 11

QSAR Quantitative Structure–Activity Relationship. 17, 18, 22, 73

QSAR/QSPR Quantitative Structure–Activity/Property Relationship. 12–14, 17–21, 24, 26, 27, 29, 35, 73, 116

QSPR Quantitative Structure–Property Relationship. 14, 47

REACH Registration, Evaluation, Authorisation and Restriction of Chemicals. 46

ReLU Rectified linear units. 81

RHS Right-hand side. 91, 117

RMSE Root Mean Square Error. 33, 47, 55, 67, 71, 117

RNN Recurrent neural network. 28, 88, 90, 101, 115

ROC Receiver operation curve. 34, 35, 41

RTECS Registry of Toxic Effects of Chemical Substances. 57, 58, 60, 62, 64, 74, 116

RVAE Recurrent variational autoencoder. 88

SAR Structure–Activity Relationship. 77, 82, 83

SD-VAE Syntax-directed variational autoencoder. 89

SMILES Simplified Molecular-Input Line-Entry System. 42, 63, 77, 88–90, 92, 95, 98, 99, 101, 102, 110, 115, 117, 118

SNE Stochastic Neighbour Embedding. 87

SOM Self-Organizing Maps. 76, 77

STL Single-task learning. 39–41, 73, 114

t-SNE t-distributed Stochastic Neighbor Embedding. 7, 14, 15, 63, 71, 72, 76–78, 80, 83, 87, 109, 110, 117

TAAR1 Trace Amine Associated Receptor 1. 76, 78, 83, 84, 86, 117

TEST Toxicity Estimation Software Tool. 47

TN True negative. 33

TP True positive. 33

Bibliography

Ahmed Abdelaziz, Hilde Spahn-Langguth, Karl-Werner Schramm, and Igor V. Tetko. Consensus modeling for HTS assays using in silico descriptors calculates the best balanced accuracy in tox21 challenge. *Frontiers in Environmental Science*, 4, February 2016. doi:10.3389/fenvs.2016.00002. URL https://doi.org/10.3389/fenvs.2016.00002.

George W. Adamson and David Bawden. A method of structure-activity correlation using wiswesser line notation. *Journal of Chemical Information and Modeling*, 15 (4):215–220, November 1975. doi:10.1021/ci60004a006. URL https://doi.org/10.1021/ci60004a006.

Subhash Ajmani and Vellarkad N. Viswanadhan. A neural network-based qsar approach for exploration of diverse multi-tyrosine kinase inhibitors and its comparison with a fragment-based approach. *Current Computer-Aided Drug Design*, 9(4): 482–490, 2013. ISSN 1573-4099/1875-6697. doi:10.2174/15734099113096660046. URL http://www.eurekaselect.com/node/116577/article.

C. L. Alden, A. Lynn, A. Bourdeau, D. Morton, F. D. Sistare, V. J. Kadambi, and L. Silverman. A Critical Review of the Effectiveness of Rodent Pharmaceutical Carcinogenesis Testing in Predicting for Human Risk. *Veterinary Pathology*, 48 (3):772–784, May 2011. ISSN 0300-9858. doi:10.1177/0300985811400445.

Jon A Arnot and Frank APC Gobas. A review of bioconcentration factor (bcf) and bioaccumulation factor (baf) assessments for organic chemicals in aquatic organisms. *Environmental Reviews*, 14(4):257–297, 2006. doi:10.1139/a06-005.

JonA. Arnot and FrankA.P.C. Gobas. A generic qsar for assessing the bioaccumulation potential of organic chemicals in aquatic food webs. *QSAR & Combinatorial Science*, 22(3):337–345, 2003. ISSN 1611-0218. doi:10.1002/qsar.200390023. URL http://dx.doi.org/10.1002/qsar.200390023.

M. Asadollahi-Baboli. Exploring QSTR analysis of the toxicity of phenols and thiophenols using machine learning methods. *Environmental Toxicology and Pharmacology*, 34(3):826–831, November 2012. ISSN 1382-6689. doi:10.1016/j.etap.2012.09.003.

Scott S. Auerbach, Ruchir R. Shah, Deepak Mav, Cynthia S. Smith, Nigel J. Walker, Molly K. Vallant, Gary A. Boorman, and Richard D. Irwin. Predicting the hepatocarcinogenic potential of alkenylbenzene flavoring agents using toxicogenomics

and machine learning. *Toxicology and Applied Pharmacology*, 243(3):300–314, March 2010. ISSN 0041-008X. doi:10.1016/j.taap.2009.11.021.

Mahendra Awale and Jean-Louis Reymond. Web-based 3D-visualization of the DrugBank chemical space. *Journal of Cheminformatics*, 8(1), December 2016. ISSN 1758-2946. doi:10.1186/s13321-016-0138-2.

Miriam Barlow. What antimicrobial resistance has taught us about horizontal gene transfer. In *Horizontal Gene Transfer*, pages 397–411. Humana Press, 2009. doi:10.1007/978-1-60327-853-9_23. URL https://doi.org/10.1007/978-1-60327-853-9_23.

Igor I. Baskin. Machine Learning Methods in Computational Toxicology. In *Computational Toxicology*, Methods in Molecular Biology, pages 119–139. Humana Press, New York, NY, 2018. ISBN 978-1-4939-7898-4 978-1-4939-7899-1. doi:10.1007/978-1-4939-7899-1_5.

Igor I Baskin and Nelly I Zhokhova. Continuous molecular fields and the concept of molecular co-fields in structure–activity studies. *Future Medicinal Chemistry*, 11 (20):2701–2713, October 2019. doi:10.4155/fmc-2018-0360. URL https://doi.org/10.4155/fmc-2018-0360.

Igor I. Baskin, David Winkler, and Igor V. Tetko. A renaissance of neural networks in drug discovery. *Expert Opinion on Drug Discovery*, 11(8):785–795, 2016. doi:10.1080/17460441.2016.1201262.

Igor I. Baskin, Vitaly P. Solov'ev, Alexander A. Bagatur'yants, and Alexandre Varnek. Predictive cartography of metal binders using generative topographic mapping. *Journal of Computer-Aided Molecular Design*, 31(8):701–714, Aug 2017. ISSN 1573-4951. doi:10.1007/s10822-017-0033-6. URL https://doi.org/10.1007/s10822-017-0033-6.

D. Beglov and B. Roux. An integral equation to describe the solvation of polar molecules in liquid water. *J. Phys. Chem.*, 101:7821–7826, 1997. doi:10.1021/jp971083h.

Arieh Ben-Naim. *Molecular Theory of Solutions*. OUP, Oxford, July 2006. ISBN 978-0-19-929969-0.

A. Patrícia Bento, Anna Gaulton, Anne Hersey, Louisa J. Bellis, Jon Chambers, Mark Davies, Felix A. Krüger, Yvonne Light, Lora Mak, Shaun McGlinchey, Michal Nowotka, George Papadatos, Rita Santos, and John P. Overington. The ChEMBL bioactivity database: An update. *Nucleic Acids Research*, 42(D1):D1083–D1090, January 2014. ISSN 0305-1048, 1362-4962. doi:10.1093/nar/gkt1031.

G. Richard Bickerton, Gaia V. Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L. Hopkins. Quantifying the chemical beauty of drugs. *Nature Chemistry*, 4(2):90–98, January 2012. doi:10.1038/nchem.1243. URL https://doi.org/10.1038/nchem.1243.

Christopher M. Bishop, Markus Svensén, and Christopher K. I. Williams. GTM: The Generative Topographic Mapping. *Neural Computation*, 10(1):215–234, January 1998. ISSN 0899-7667, 1530-888X. doi:10.1162/089976698300017953.

C.M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. URL http://research.microsoft.com/en-us/um/people/cmbishop/prml.

Lorenz C. Blum, Ruud van Deursen, and Jean-Louis Reymond. Visualisation and subsets of the chemical universe database GDB-13 for virtual screening. *Journal of Computer-Aided Molecular Design*, 25(7):637–647, July 2011. ISSN 0920-654X, 1573-4951. doi:10.1007/s10822-011-9436-y.

Leo Breiman. Random Forests. *Machine Learning*, 45(1):5–32, October 2001. ISSN 0885-6125, 1573-0565. doi:10.1023/A:1010933404324.

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *CoRR*, abs/1611.08097, 2016. URL http://arxiv.org/abs/1611.08097.

Justus J. Bürgi, Mahendra Awale, Silvan D. Boss, Tifany Schaer, Fabrice Marger, Juan M. Viveros-Paredes, Sonia Bertrand, Jürg Gertsch, Daniel Bertrand, and Jean-Louis Reymond. Discovery of Potent Positive Allosteric Modulators of the $a3\beta2$ Nicotinic Acetylcholine Receptor by a Chemical Space Walk in ChEMBL. *ACS Chemical Neuroscience*, 5(5):346–359, May 2014. ISSN 1948-7193, 1948-7193. doi:10.1021/cn4002297.

Dong-Sheng Cao, Yan-Ning Yang, Jian-Chao Zhao, Jun Yan, Shao Liu, Qian-Nan Hu, Qing-Song Xu, and Yi-Zend Liang. Computer-aided prediction of toxicity with substructure pattern and random forest. *Journal of Chemometrics*, 26(1-2): 7–15. ISSN 1099-128X. doi:10.1002/cem.1416.

Rich Caruana. Multitask learning. In *Learning to Learn*, pages 95–133. Springer US, 1998. doi:10.1007/978-1-4615-5529-2_5. URL https://doi.org/10.1007/978-1-4615-5529-2_5.

D.A. Case, R.M. Betz, W. Botello-Smith, D.S. Cerutti, III T.E. Cheatham, T.A. Darden, R.E. Duke, T.J. Giese, H. Gohlke, A.W. Goetz, N. Homeyer, S. Izadi, P. Janowski, J. Kaus, A. Kovalenko, T.S. Lee, S. LeGrand, P. Li, C. Lin, T. Luchko, R. Luo, B. Madej, D. Mermelstein, K.M. Merz, G. Monard, H. Nguyen, H.T. Nguyen, I. Omelyan, A. Onufriev, D.R. Roe, A. Roitberg, C. Sagui, C.L. Simmerling, J. Swails, R.C. Walker, J. Wang, R.M. Wolf, X. Wu, L. Xiao, D.M. York, and P.A. Kollman. Amber 2016, 2016. University of California, San Francisco.

D. Chandler, J. D. Mccoy, and S. J. Singer. Density functional theory of nonuniform polyatomic systems. 1. general formulation. *J. Chem. Phys.*, 85(10):5971–5976, 1986. doi:10.1063/1.451510. \cite{Chandler1986tqr}.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. *CoRR*, abs/1603.02754, 2016. URL http://arxiv.org/abs/1603.02754.

Artem Cherkasov. Inductive QSAR Descriptors. Distinguishing Compounds with Antibacterial Activity by Artificial Neural Networks. *International Journal of Molecular Sciences*, 6(1):63–86, January 2005. ISSN 1422-0067. doi:10.3390/i6010063.

Connor W. Coley, Luke Rogers, William H. Green, and Klavs F. Jensen. SCScore: Synthetic complexity learned from a reaction corpus. *Journal of Chemical Information and Modeling*, 58(2):252–261, January 2018. doi:10.1021/acs.jcim.7b00622. URL https://doi.org/10.1021/acs.jcim.7b00622.

Richard D. Cramer, David E. Patterson, and Jeffrey D. Bunce. Comparative molecular field analysis (CoMFA). 1. effect of shape on binding of steroids to carrier proteins. *Journal of the American Chemical Society*, 110(18):5959–5967, August 1988. doi:10.1021/ja00226a005. URL https://doi.org/10.1021/ja00226a005.

George E. Dahl, Navdeep Jaitly, and Ruslan Salakhutdinov. Multi-task Neural Networks for QSAR Predictions. *arXiv:1406.1231 [cs, stat]*, June 2014.

Hanjun Dai, Yingtao Tian, Bo Dai, Steven Skiena, and Le Song. Syntax-directed variational autoencoder for structured data. *CoRR*, abs/1802.08786, 2018. URL http://arxiv.org/abs/1802.08786.

EURAS Bioconcentration Factor (BCF) Gold Standard Database. Euras bioconcentration factor (bcf) gold standard database. URL http://ambit.sourceforge.net/euras/. Accessed: 2017-04-04.

Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 233–240, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595933832. doi:10.1145/1143844.1143874. URL https://doi.org/10.1145/1143844.1143874.

Antonio de la Vega de León and Jürgen Bajorath. Chemical space visualization: transforming multidimensional chemical spaces into similarity-based molecular networks. *Future Medicinal Chemistry*, 8(14):1769–1778, 2016. doi:10.4155/fmc-2016-0023. URL https://doi.org/10.4155/fmc-2016-0023. PMID: 27572425.

Antonio de la Vega de León, Beining Chen, and Valerie J. Gillet. Effect of missing data on multitask prediction methods. *Journal of Cheminformatics*, 10(1), December 2018. ISSN 1758-2946. doi:10.1186/s13321-018-0281-z.

S. Dimitrov, N. Dimitrova, T. Parkerton, M. Comber, M. Bonnell, and O. Mekenyan. Base-line model for identifying the bioaccumulation potential of chemicals. *SAR and QSAR in Environmental Research*, 16(6):531–554, 2005. doi:10.1080/10659360500474623. URL https://doi.org/10.1080/10659360500474623. PMID: 16428130.

Chris Drummond and Robert C Holte. Explicitly representing expected cost: An alternative to roc representation. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 198–207, 2000.

Malgorzata N. Drwal, Priyanka Banerjee, Mathias Dunkel, Martin R. Wettig, and Robert Preissner. ProTox: A web server for the in silico prediction of rodent oral toxicity. *Nucleic Acids Research*, 42(W1):W53–W58, July 2014. ISSN 0305-1048. doi:10.1093/nar/gku401.

Q. H. Du, D. Beglov, and B. Roux. Solvation free energy of polar and nonpolar molecules in water: An extended interaction site integral equation theory in three dimensions. 104(4):796–805, February 2000. doi:10.1021/jp992712l.

D. Duvenaud, D. Maclaurin, J. Aguilera-Iparraguirre, R. Gómez-Bombarelli, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams. Convolutional Networks on Graphs for Learning Molecular Fingerprints. *ArXiv e-prints*, September 2015.

U.S. EPA. User's guide for t.e.s.t.(toxicity estimation software tool), 2016. URL https://www.epa.gov/sites/production/files/2016-05/documents/600r16058.pdf. Accessed: May 05, 2017.

Peter Ertl and Ansgar Schuffenhauer. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of Cheminformatics*, 1(1), June 2009. doi:10.1186/1758-2946-1-8. URL https://doi.org/10.1186/1758-2946-1-8.

Jun Feng, Laura Lurati, Haojun Ouyang, Tracy Robinson, Yuanyuan Wang, Shenglan Yuan, and S. Stanley Young. Predictive Toxicology: Benchmarking Molecular Descriptors and Statistical Methods. *Journal of Chemical Information and Computer Sciences*, 43(5):1463–1470, September 2003. ISSN 0095-2338. doi:10.1021/ci034032s.

Spencer M. Free and James W. Wilson. A mathematical contribution to structure-activity studies. *Journal of Medicinal Chemistry*, 7(4):395–399, July 1964. doi:10.1021/jm00334a001. URL https://doi.org/10.1021/jm00334a001.

Jerome H. Friedman. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378, February 2002. doi:10.1016/s0167-9473(01)00065-2. URL https://doi.org/10.1016/s0167-9473(01)00065-2.

Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202, April 1980. doi:10.1007/bf00344251. URL https://doi.org/10.1007/bf00344251.

Simon Funk. Smorms3 - blog entry: Rmsprop loses to smorms3 - beware the epsilon!, 2015. URL http://sifter.org/simon/journal/20150420.html. Accessed: 2017-04-04.

Hongyang Gao and Shuiwang Ji. Graph u-nets. *CoRR*, abs/1905.05178, 2019. URL http://arxiv.org/abs/1905.05178.

Héléna A. Gaspar, Igor I. Baskin, Gilles Marcou, Dragos Horvath, and Alexandre Varnek. Chemical data visualization and analysis with incremental generative topographic mapping: Big data challenge. *Journal of Chemical Information and Modeling*, 55(1):84–94, 2015. doi:10.1021/ci500575y. URL https://doi.org/10.1021/ci500575y. PMID: 25423612.

Anna Gaulton, Anne Hersey, Michał Nowotka, A. Patrícia Bento, Jon Chambers, David Mendez, Prudence Mutowo, Francis Atkinson, Louisa J. Bellis, Elena Cibrián-Uhalte, Mark Davies, Nathan Dedman, Anneli Karlsson, María Paula Magariños, John P. Overington, George Papadatos, Ines Smit, and Andrew R. Leach. The ChEMBL database in 2017. *Nucleic Acids Research*, 45(D1):D945–D954, January 2017. ISSN 0305-1048, 1362-4962. doi:10.1093/nar/gkw1074.

Lionel Gendre, Rosa Ramirez, and Daniel Borgis. Classical density functional theory of solvation in molecular solvents: Angular grid implementation. *Chemical Physics Letters*, 474(4):366–370, June 2009. ISSN 0009-2614. doi:10.1016/j.cplett.2009.04.077. URL http://www.sciencedirect.com/science/article/pii/S0009261409004175.

Samuel Genheden, Tyler Luchko, Sergey Gusarov, Andriy Kovalenko, and Ulf Ryde. An MM/3D-RISM approach for ligand binding affinities. *J. Phys. Chem. B*, 114(25):8505–8516, July 2010. ISSN 1520-6106. doi:10.1021/jp101461s. URL http://dx.doi.org/10.1021/jp101461s.

Irène Gijbels. Censored data. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(2):178–188, March 2010. doi:10.1002/wics.80. URL https://doi.org/10.1002/wics.80.

Robert C. Glen, Andreas Bender, Catrin H. Arnby, Lars Carlsson, Scott Boyer, and James Smith. Circular fingerprints: Flexible molecular descriptors with applications from physical chemistry to ADME. *IDrugs: the investigational drugs journal*, 9(3):199–204, March 2006. ISSN 1369-7056.

Alexander Golbraikh and Alexander Tropsha. Beware of q2! *Journal of Molecular Graphics and Modelling*, 20(4):269–276, January 2002. doi:10.1016/s1093-3263(01)00123-1.

V. Golkov, M. J. Skwark, A. Mirchev, G. Dikov, A. R. Geanes, J. Mendenhall, J. Meiler, and D. Cremers. 3D Deep Learning for Biological Function Prediction from Physical Fields. *ArXiv e-prints*, April 2017.

Laurent Gomez. Decision making in medicinal chemistry: The power of our intuition. *ACS Medicinal Chemistry Letters*, 9(10):956–958, September 2018. doi:10.1021/acsmedchemlett.8b00359. URL https://doi.org/10.1021/acsmedchemlett.8b00359.

Rafael Gómez-Bombarelli, Jennifer N. Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D. Hirzel, Ryan P. Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276, January 2018. doi:10.1021/acscentsci.7b00572. URL https://doi.org/10.1021/acscentsci.7b00572.

Paola Gramatica and Ester Papa. An update of the bcf qsar model based on theoretical molecular descriptors. *QSAR & Combinatorial Science*, 24(8):953–960, 2005. ISSN 1611-0218. doi:10.1002/qsar.200530123. URL http://dx.doi.org/10.1002/qsar.200530123.

Paola Gramatica and Alessandro Sangion. 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, June 2016. doi:10.1021/acs.jcim.6b00088. URL https://doi.org/10.1021/acs.jcim.6b00088.

Sitarama B. Gunturi, Kotu Archana, Akash Khandelwal, and Ramamurthi Narayanan. Prediction of hERG Potassium Channel Blockade Using kNN-QSAR and Local Lazy Regression Methods. *QSAR & Combinatorial Science*, 27(11-12):1305–1317, December 2008. ISSN 1611-0218. doi:10.1002/qsar.200810072.

Martin Gütlein, Andreas Karwath, and Stefan Kramer. Ches-mapper - chemical space mapping and visualization in 3d. *Journal of Cheminformatics*, 4(1):7, Mar 2012. ISSN 1758-2946. doi:10.1186/1758-2946-4-7. URL https://doi.org/10.1186/1758-2946-4-7.

Stefan Güssregen, Hans Matter, Gerhard Hessler, Evanthia Lionta, Jochen Heil, and Stefan M. Kast. Thermodynamic characterization of hydration sites from integral equation-derived free energy densities: Application to protein binding sites and ligand series. *J. Chem. Inf. Model.*, 57(7):1652–1666, July 2017. ISSN 1549-9596. doi:10.1021/acs.jcim.6b00765. URL http://dx.doi.org/10.1021/acs.jcim.6b00765.

Lowell H. Hall and Lemont B. Kier. Electrotopological State Indices for Atom Types: A Novel Combination of Electronic, Topological, and Valence State Information. *Journal of Chemical Information and Computer Sciences*, 35(6):1039–1045, November 1995. ISSN 0095-2338. doi:10.1021/ci00028a014.

Louis P. Hammett. The effect of structure upon the reactions of organic compounds. benzene derivatives. *Journal of the American Chemical Society*, 59(1):96–103, January 1937. doi:10.1021/ja01280a022. URL https://doi.org/10.1021/ja01280a022.

Corwin. Hansch and Toshio. Fujita. p-- analysis. a method for the correlation of biological activity and chemical structure. *Journal of the American Chemical Society*, 86(8):1616–1626, April 1964. doi:10.1021/ja01062a035. URL https://doi.org/10.1021/ja01062a035.

Jean-Pierre Hansen and Ian R. McDonald. *Theory of Simple Liquids, Fourth Edition: with Applications to Soft Matter*. Academic Press, Amstersdam, 4 edition edition, October 2013. ISBN 978-0-12-387032-2.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer New York, 2009. doi:10.1007/978-0-387-84858-7. URL https://doi.org/10.1007/978-0-387-84858-7.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *arXiv:1502.01852*, 2015. URL http://arxiv.org/abs/1502.01852.

Ping He, Cheng-Jian Xu, Yi-Zeng Liang, and Kai-Tai Fang. Improving the classification accuracy in chemistry via boosting technique. *Chemometrics and Intelligent Laboratory Systems*, 70(1):39–46, January 2004. doi:10.1016/j.chemolab.2003.10.001. URL https://doi.org/10.1016/j.chemolab.2003.10.001.

Reiko Heckel. Graph transformation in a nutshell. *Electronic Notes in Theoretical Computer Science*, 148(1):187–198, February 2006. doi:10.1016/j.entcs.2005.12.018. URL https://doi.org/10.1016/j.entcs.2005.12.018.

Jennifer Hemmerich, Ece Asilar, and Gerhard F. Ecker. Cover: conformational oversampling as data augmentation for molecules. *Journal of Cheminformatics*, 12 (1):18, Mar 2020. ISSN 1758-2946. doi:10.1186/s13321-020-00420-z. URL https://doi.org/10.1186/s13321-020-00420-z.

Fumio Hirata. *Molecular Theory of Solvation*. Kluwer Academic Publishers, New York, 2003.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, November 1997. doi:10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

Johnny X. Huang, Mark A. Blaskovich, and Matthew A. Cooper. Cell- and biomarker-based assays for predicting nephrotoxicity. *Expert Opinion on Drug Metabolism & Toxicology*, 10(12):1621–1635, December 2014. ISSN 1742-5255. doi:10.1517/17425255.2014.967681.

James Inglese, Douglas S. Auld, Ajit Jadhav, Ronald L. Johnson, Anton Simeonov, Adam Yasgar, Wei Zheng, and Christopher P. Austin. Quantitative high-throughput screening: A titration-based approach that efficiently identifies biological activities in large chemical libraries. *Proceedings of the National Academy of Sciences*, 103(31):11473–11478, August 2006. ISSN 0027-8424, 1091-6490. doi:10.1073/pnas.0604348103.

Institute of Medicine (US) Committee on Internet Access to the National Library of Medicine's Toxicology and Environmental Health Databases. *Internet Access to the National Library of Medicine's Toxicology and Environmental Health*

Databases. The National Academies Collection: Reports funded by National Institutes of Health. National Academies Press (US), Washington (DC), 1998. ISBN 978-0-309-06299-2.

Sergey Ioffe and Christian Szegedy. 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*, ICML'15, pages 448–456, Lille, France, 2015. JMLR.org.

Ovidiu Ivanciuc. Applications of support vector machines in chemistry. In *Reviews in Computational Chemistry*, pages 291–400. John Wiley & Sons, Inc., February 2007. doi:10.1002/9780470116449.ch6. URL https://doi.org/10.1002/9780470116449.ch6.

Sabrina Jaeger, Simone Fulle, and Samo Turk. Mol2vec: Unsupervised machine learning approach with chemical intuition. *Journal of Chemical Information and Modeling*, 58(1):27–35, January 2018. doi:10.1021/acs.jcim.7b00616. URL https://doi.org/10.1021/acs.jcim.7b00616.

Araz Jakalian, David B. Jack, and Christopher I. Bayly. Fast, efficient generation of high-quality atomic charges. am1-bcc model: Ii. parameterization and validation. *Journal of Computational Chemistry*, 23(16):1623–1641, 2002. ISSN 1096-987X. doi:10.1002/jcc.10128. URL http://dx.doi.org/10.1002/jcc.10128.

Ebejer Jean-Paul, Morris Garrett, and Deane Charlotte. Freely available conformer generation methods: How good are they? *Journal of Chemical Information and Modeling*, 52(5):1146–1158, 2012. doi:10.1021/ci2004658. PMID: 22482737.

Guillaume Jeanmairet, Maximilien Levesque, Rodolphe Vuilleumier, and Daniel Borgis. Molecular density functional theory of water. *J. Phys. Chem. Lett.*, 4(4):619–624, February 2013. ISSN 1948-7185. doi:10.1021/jz301956b. URL http://dx.doi.org/10.1021/jz301956b.

Changge Ji, Fredrik Svensson, Azedine Zoufir, and Andreas Bender. eMolTox: Prediction of molecular toxicity with confidence. *Bioinformatics*. doi:10.1093/bioinformatics/bty135.

Wengong Jin, Regina Barzilay, and Tommi S. Jaakkola. Junction tree variational autoencoder for molecular graph generation. *CoRR*, abs/1802.04364, 2018. URL http://arxiv.org/abs/1802.04364.

Hiroshi Kajino. Molecular hypergraph grammar with its application to molecular optimization. *CoRR*, abs/1809.02745, 2018. URL http://arxiv.org/abs/1809.02745.

Dmitry S. Karlov, Sergey Sosnin, Igor V. Tetko, and Maxim V. Fedorov. Chemical space exploration guided by deep neural networks. *RSC advances*, (9):5151–5157, 2019. ISSN 2046-2069. doi:10.1039/c8ra10182e. Place: United Kingdom Publisher: United Kingdom.

Dmitry S. Karlov, Sergey Sosnin, Maxim V. Fedorov, and Petr Popov. graphDelta: MPNN scoring function for the affinity prediction of protein–ligand complexes. *ACS Omega*, 5(10):5150–5159, March 2020. doi:10.1021/acsomega.9b04162. URL https://doi.org/10.1021/acsomega.9b04162.

Pavel Karpov, Guillaume Godin, and Igor V. Tetko. Transformer-CNN: Swiss knife for QSAR modeling and interpretation. *Journal of Cheminformatics*, 12 (1), March 2020. doi:10.1186/s13321-020-00423-w. URL https://doi.org/10.1186/s13321-020-00423-w.

Bertram G. Katzung and Anthony J. Trevor. *Basic and Clinical Pharmacology 13 E*. McGraw-Hill Education / Medical, New York, 13 edition edition, December 2014. ISBN 978-0-07-182505-4.

Gregory W. Kauffman and Peter C. Jurs. QSAR and k-nearest neighbor classification analysis of selective cyclooxygenase-2 inhibitors using topologically-based numerical descriptors. *Journal of Chemical Information and Computer Sciences*, 41(6):1553–1560, September 2001. doi:10.1021/ci010073h. URL https://doi.org/10.1021/ci010073h.

Shilva Kayastha, Ryo Kunimoto, Dragos Horvath, Alexandre Varnek, and Jürgen Bajorath. From bird's eye views to molecular communities: two-layered visualization of structure–activity relationships in large compound data sets. *Journal of Computer-Aided Molecular Design*, 31(11):961–977, Nov 2017. ISSN 1573-4951. doi:10.1007/s10822-017-0070-1. URL https://doi.org/10.1007/s10822-017-0070-1.

Steven Kearnes, Kevin McCloskey, Marc Berndl, Vijay Pande, and Patrick Riley. Molecular graph convolutions: moving beyond fingerprints. *Journal of Computer-Aided Molecular Design*, 30(8):595–608, 2016. ISSN 1573-4951. doi:10.1007/s10822-016-9938-8. URL http://dx.doi.org/10.1007/s10822-016-9938-8.

Sunghwan Kim, Paul A. Thiessen, Evan E. Bolton, Jie Chen, Gang Fu, Asta Gindulyte, Lianyi Han, Jane He, Siqian He, Benjamin A. Shoemaker, Jiyao Wang, Bo Yu, Jian Zhang, and Stephen H. Bryant. Pubchem substance and compound databases. *Nucleic Acids Research*, 44(D1):D1202–D1213, 2016. doi:10.1093/nar/gkv951. URL http://dx.doi.org/10.1093/nar/gkv951.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, dec 2014. URL http://arxiv.org/abs/1412.6980v9. Accessed: 2017-04-04.

N. Kireeva, I. I. Baskin, H. A. Gaspar, D. Horvath, G. Marcou, and A. Varnek. Generative Topographic Mapping (GTM): Universal Tool for Data Visualization, Structure-Activity Modeling and Dataset Comparison. *Molecular Informatics*, 31 (3-4):301–312, April 2012. ISSN 18681743. doi:10.1002/minf.201100163.

Peter Kirkpatrick and Clare Ellis. Chemical space. *Nature*, 432(7019):823–823, December 2004. doi:10.1038/432823a. URL https://doi.org/10.1038/432823a.

Gerhard Klebe and Ute Abraham. *Journal of Computer-Aided Molecular Design*, 13 (1):1–10, 1999. doi:10.1023/a:1008047919606. URL https://doi.org/10.1023/a:1008047919606.

Gilles Klopman. MULTICASE 1. a hierarchical computer automated structure evaluation program. *Quantitative Structure-Activity Relationships*, 11(2):176–184, 1992. doi:10.1002/qsar.19920110208. URL https://doi.org/10.1002/qsar.19920110208.

Walter Koch. Electronic structure of matter – wave functions and density functionals, January 1999.

Teuvo Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43(1):59–69, 1982. ISSN 0340-1200, 1432-0770. doi:10.1007/BF00337288.

Yury I. Kostyukevich, Gleb Vladimirov, Elena Stekolschikova, Daniil G. Ivanov, Arthur Yablokov, Alexander Ya Zherebker, Sergey Sosnin, Alexey Orlov, Maxim Fedorov, Philipp Khaitovich, and Evgeny N. Nikolaev. Hydrogen/Deuterium exchange aids compounds identification for LC-MS and MALDI imaging lipidomics. *Analytical Chemistry*, 2019. ISSN 1520-6882; 0003-2700. doi:10.1021/acs.analchem.9b02461. Place: United States Publisher: United States.

A Kovalenko. Three-dimensional rism theory for molecular liquids and solid-liquid interfaces. In F. Hirata, editor, *Molecular Theory of Solvation*, Understanding Chemical Reactivity, pages 169–275. Springer Netherlands, 2003. ISBN 978-1-4020-1562-5. edited by F. Hirata.

A. Kovalenko and F. Hirata. Self-consistent description of a metal-water interface by the kohn-sham density functional theory and the three-dimensional reference interaction site model. *J. Chem. Phys.*, 110:10095–10112, 1999. doi:10.1063/1.478883.

Andriy Kovalenko and Fumio Hirata. Hydration free energy of hydrophobic solutes studied by a reference interaction site model with a repulsive bridge correction and a thermodynamic perturbation method. *The Journal of Chemical Physics*, 113(7):2793–2805, 2000a. doi:10.1063/1.1305885.

Andriy Kovalenko and Fumio Hirata. Potentials of mean force of simple ions in ambient aqueous solution. i. three-dimensional reference interaction site model approach. *J. Chem. Phys*, 112(23):10391–10402, June 2000b. ISSN 0021-9606, 1089-7690. doi:10.1063/1.481676. URL http://scitation.aip.org/content/aip/journal/jcp/112/23/10.1063/1.481676.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

J. B. Kruskal. Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29(2):115–129, Jun 1964. ISSN 1860-0980. doi:10.1007/BF02289694. URL https://doi.org/10.1007/BF02289694.

Hugo Kubinyi. Free wilson analysis. theory, applications and its relationship to hansch analysis. *Quantitative Structure-Activity Relationships*, 7(3):121–133, 1988. doi:10.1002/qsar.19880070303. URL https://doi.org/10.1002/qsar.19880070303.

S. Kullback and R. A. Leibler. On information and sufficiency. *Ann. Math. Statist.*, 22(1):79–86, 03 1951. doi:10.1214/aoms/1177729694. URL https://doi.org/10.1214/aoms/1177729694.

Matt J. Kusner, Brooks Paige, and José Miguel Hernández-Lobato. Grammar Variational Autoencoder. *arXiv:1703.01925 [stat]*, March 2017.

Peter S. Kutchukian, Nadya Y. Vasilyeva, Jordan Xu, Mika K. Lindvall, Michael P. Dillon, Meir Glick, John D. Coley, and Natasja Brooijmans. Inside the mind of a medicinal chemist: The role of human bias in compound prioritization during drug discovery. *PLoS ONE*, 7(11):e48476, November 2012. doi:10.1371/journal.pone.0048476. URL https://doi.org/10.1371/journal.pone.0048476.

V. E. Kuz'min, A. G. Artemenko, and E. N. Muratov. Hierarchical QSAR technology based on the Simplex representation of molecular structure. *Journal of Computer-Aided Molecular Design*, 22(6-7):403–421, 2008 Jun-Jul. ISSN 0920-654X. doi:10.1007/s10822-008-9179-6.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521 (7553):436–444, May 2015. ISSN 0028-0836. Insight.

Alpha A. Lee, Qingyi Yang, Vishnu Sresht, Peter Bolgar, Xinjun Hou, Jacquelyn L. Klug-McLeod, and Christopher R. Butler. Molecular transformer unifies reaction prediction and retrosynthesis across pharma chemical space. *Chemical Communications*, 55(81):12152–12155, 2019. doi:10.1039/c9cc05122h. URL https://doi.org/10.1039/c9cc05122h.

Nicole A. Lerminiaux and Andrew D.S. Cameron. Horizontal transfer of antibiotic resistance genes in clinical environments. *Canadian Journal of Microbiology*, 65 (1):34–44, January 2019. doi:10.1139/cjm-2018-0275. URL https://doi.org/10.1139/cjm-2018-0275.

Xiang Li, Youjun Xu, Luhua Lai, and Jianfeng Pei. Prediction of human cytochrome P450 inhibition using a multi-task deep autoencoder neural network. *Molecular Pharmaceutics*, May 2018. ISSN 1543-8384. doi:10.1021/acs.molpharmaceut.8b00110.

Xiao Li, Yuan Zhang, Hongna Chen, Huanhuan Li, and Yong Zhao. Insights into the Molecular Basis of the Acute Contact Toxicity of Diverse Organic Chemicals in the Honey Bee. *Journal of Chemical Information and Modeling*, 57(12):2948–2957, December 2017. ISSN 1549-9596. doi:10.1021/acs.jcim.7b00476.

Rong Liu, Hai Yuan Zhang, Zhao Xia Ji, Robert Rallo, Tian Xia, Chong Hyun Chang, Andre Nel, and Yoram Cohen. Development of structure–activity relationship for metal oxide nanoparticles. *Nanoscale*, 5(12):5644–5653, June 2013. ISSN 2040-3372. doi:10.1039/C3NR01533E.

Tyler Luchko, Sergey Gusarov, Daniel R. Roe, Carlos Simmerling, David A. Case, Jack Tuszynski, and Andriy Kovalenko. Three-dmensional molecular theory of solvation coupled with molecular dynamics in amber. *J. Chem. Theory Comput.*, 6(3):607–624, March 2010. ISSN 1549-9618. doi:10.1021/ct900460m. URL http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2861832/.

Ka Lum, David Chandler, and John D. Weeks. Hydrophobicity at small and large length scales. *J. Phys. Chem. B*, 103(22):4570–4577, June 1999. ISSN 1520-6106. doi:10.1021/jp984327m. URL http://dx.doi.org/10.1021/jp984327m.

Junshui Ma, Robert P. Sheridan, Andy Liaw, George E. Dahl, and Vladimir Svetnik. Deep neural nets as a method for quantitative structure–activity relationships. *Journal of Chemical Information and Modeling*, 55(2):263–274, February 2015a. doi:10.1021/ci500747n. URL https://doi.org/10.1021/ci500747n.

Junshui Ma, Robert P. Sheridan, Andy Liaw, George E. Dahl, and Vladimir Svetnik. Deep neural nets as a method for quantitative structure–activity relationships. *Journal of Chemical Information and Modeling*, 55(2):263–274, 2015b. doi:10.1021/ci500747n. URL http://dx.doi.org/10.1021/ci500747n. PMID: 25635324.

Yvonne C. Martin. Remembrances of corwin hansch. *Journal of Computer-Aided Molecular Design*, 25(6):519–523, June 2011. doi:10.1007/s10822-011-9452-y. URL https://doi.org/10.1007/s10822-011-9452-y.

Vijay H. Masand and Vesna Rastija. PyDescriptor : A new PyMOL plugin for calculating thousands of easily understandable molecular descriptors. *Chemometrics and Intelligent Laboratory Systems*, 169:12–18, October 2017. ISSN 01697439. doi:10.1016/j.chemolab.2017.08.003.

Nobuyuki Matubayasi and Masaru Nakahara. Theory of solutions in the energetic representation. i. formulation. *The Journal of Chemical Physics*, 113(15):6070–6081, October 2000. ISSN 0021-9606. doi:10.1063/1.1309013. URL http://aip.scitation.org/doi/abs/10.1063/1.1309013.

Andreas Mayr, Günter Klambauer, Thomas Unterthiner, and Sepp Hochreiter. DeepTox: Toxicity prediction using deep learning. *Frontiers in Environmental Science*, 3, February 2016. doi:10.3389/fenvs.2015.00080. URL https://doi.org/10.3389/fenvs.2015.00080.

Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115–133, December 1943. doi:10.1007/bf02478259. URL https://doi.org/10.1007/bf02478259.

Kiran N. Meekings, Cory S.M. Williams, and John E. Arrowsmith. Orphan drug development: an economically viable strategy for biopharma r&d. *Drug Discovery Today*, 17(13-14):660–664, July 2012. doi:10.1016/j.drudis.2012.02.005. URL https://doi.org/10.1016/j.drudis.2012.02.005.

Cleber Melo-Filho, Rodolpho Braga, and Carolina Andrade. 3d-QSAR approaches in drug design: Perspectives to generate reliable CoMFA models. *Current Computer Aided-Drug Design*, 10(2):148–159, July 2014. doi:10.2174/1573409910666140410111043. URL https://doi.org/10.2174/1573409910666140410111043.

Maksim Misin. Can approximate integral equation theories accurately predict solvation thermodynamics? April 2017. doi:10.5281/zenodo.495336. URL http://arxiv.org/abs/1704.05246. arXiv: 1704.05246.

Maksim Misin, Maxim V. Fedorov, and David S. Palmer. Communication: Accurate hydration free energies at a wide range of temperatures from 3D-RISM. *J. Chem. Phys.*, 142(9):091105, March 2015. ISSN 0021-9606, 1089-7690. doi:10.1063/1.4914315. URL http://scitation.aip.org/content/aip/journal/jcp/142/9/10.1063/1.4914315.

Maksim Misin, Maxim V. Fedorov, and David S. Palmer. Hydration free energies of molecular ions from theory and simulation. *J. Phys. Chem. B*, 120(5):975–983, February 2016a. ISSN 1520-6106. doi:10.1021/acs.jpcb.5b10809. URL http://dx.doi.org/10.1021/acs.jpcb.5b10809.

Maksim Misin, Petteri A. Vainikka, Maxim V. Fedorov, and David S. Palmer. Salting-out effects by pressure-corrected 3d-rism. *The Journal of Chemical Physics*, 145(19):194501, 2016b. doi:10.1063/1.4966973.

John B O Mitchell. Machine learning methods in chemoinformatics. *Wiley Interdisciplinary Reviews. Computational Molecular Science*, 4(5):468–481, September 2014. ISSN 1759-0876. doi:10.1002/wcms.1183.

Seyed Mohamad Moosavi, Arunraj Chidambaram, Leopold Talirz, Maciej Haranczyk, Kyriakos C. Stylianou, and Berend Smit. Capturing chemical intuition in synthesis of metal-organic frameworks. *Nature Communications*, 10(1), February 2019. doi:10.1038/s41467-019-08483-9. URL https://doi.org/10.1038/s41467-019-08483-9.

Hirotomo Moriwaki, Yu-Shi Tian, Norihito Kawashita, and Tatsuya Takagi. Mordred: a molecular descriptor calculator. *Journal of Cheminformatics*, 10(1), February 2018. doi:10.1186/s13321-018-0258-y. URL https://doi.org/10.1186/s13321-018-0258-y.

Kyaw-Zeyar Myint, Lirong Wang, Qin Tong, and Xiang-Qun Xie. Molecular fingerprint-based artificial neural networks qsar for ligand biological activity predictions. *Molecular Pharmaceutics*, 9(10):2912–2923, 2012. doi:10.1021/mp300237z. URL http://dx.doi.org/10.1021/mp300237z. PMID: 22937990.

Michael M. Mysinger, Michael Carchia, John. J. Irwin, and Brian K. Shoichet. Directory of useful decoys, enhanced (dud-e): Better ligands and decoys for better benchmarking. *Journal of Medicinal Chemistry*, 55(14):6582–6594, 2012. doi:10.1021/jm300687e. URL https://doi.org/10.1021/jm300687e. PMID: 22716043.

Bruno J. Neves, Rodolpho C. Braga, Cleber C. Melo-Filho, José Teófilo Moreira-Filho, Eugene N. Muratov, and Carolina Horta Andrade. QSAR-based virtual screening: Advances and applications in drug discovery. *Frontiers in Pharmacology*, 9, November 2018. doi:10.3389/fphar.2018.01275. URL https://doi.org/10.3389/fphar.2018.01275.

Noel OBoyle and Andrew Dalke. DeepSMILES: An Adaptation of SMILES for Use in Machine-Learning of Chemical Structures. September 2018. doi:10.26434/chemrxiv.7097960.v1.

OECD. Test no. 305: Bioaccumulation in fish: Aqueous and dietary exposure. 2012. doi:http://dx.doi.org/10.1787/9789264185296-en. URL /content/book/9789264185296-en. Paris.

Marcus Olivecrona, Thomas Blaschke, Ola Engkvist, and Hongming Chen. Molecular de-novo design through deep reinforcement learning. *Journal of Cheminformatics*, 9(1), September 2017. doi:10.1186/s13321-017-0235-x. URL https://doi.org/10.1186/s13321-017-0235-x.

Sergey Osipenko, Inga Bashkirova, Sergey Sosnin, Oxana Kovaleva, Maxim Fedorov, Eugene Nikolaev, and Yury Kostyukevich. Machine learning to predict retention time of small molecules in nano-HPLC. *Analytical and Bioanalytical Chemistry*, August 2020. doi:10.1007/s00216-020-02905-0. URL https://doi.org/10.1007/s00216-020-02905-0.

Dmitry I Osolodkin, Eugene V Radchenko, Alexey A Orlov, Andrey E Voronkov, Vladimir A Palyulin, and Nikolay S Zefirov. Progress in visual representations of chemical space. *Expert Opinion on Drug Discovery*, 10(9):959–973, September 2015. ISSN 1746-0441, 1746-045X. doi:10.1517/17460441.2015.1060216.

David S. Palmer, Andrey I. Frolov, Ekaterina L. Ratkova, and Maxim V. Fedorov. Towards a universal method for calculating hydration free energies: a 3D reference interaction site model with partial molar volume correction. *J. Phys.: Condens. Matter*, 22(49):492101, December 2010. ISSN 0953-8984. doi:10.1088/0953-8984/22/49/492101. URL http://iopscience.iop.org/0953-8984/22/49/492101.

David S. Palmer, Maksim Misin, Maxim V. Fedorov, and Antonio Llinas. Fast and general method to predict the physicochemical properties of druglike molecules using the integral equation theory of molecular liquids. *Mol. Pharmaceutics*, 12(9):3420–3432, September 2015. ISSN 1543-8384. doi:10.1021/acs.molpharmaceut.5b00441. URL http://dx.doi.org/10.1021/acs.molpharmaceut.5b00441.

Fabio Pammolli, Laura Magazzini, and Massimo Riccaboni. The productivity crisis in pharmaceutical r&d. *Nature Reviews Drug Discovery*, 10(6):428–438, June 2011. doi:10.1038/nrd3405. URL https://doi.org/10.1038/nrd3405.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

E. Papa, J.C. Dearden, and P. Gramatica. Linear qsar regression models for the prediction of bioconcentration factors by physicochemical properties and structural theoretical molecular descriptors. *Chemosphere*, 67(2):351–358, 2007. ISSN 0045-6535. doi:https://doi.org/10.1016/j.chemosphere.2006.09.079. URL http://www.sciencedirect.com/science/article/pii/S0045653506012732.

Júlia G.B. Pedreira, Lucas S. Franco, and Eliezer J. Barreiro. Chemical intuition in drug design and discovery. *Current Topics in Medicinal Chemistry*, 19(19): 1679–1693, October 2019. doi:10.2174/1568026619666190620144142. URL https://doi.org/10.2174/1568026619666190620144142.

John S. Perkyns and Montgomery B. Pettitt. A dielectrically consistent interaction site theory for solvent–electrolyte mixtures. *Chemical Physics Letters*, 190(6):626–630, 1992. ISSN 0009-2614. doi:http://dx.doi.org/10.1016/0009-2614(92)85201-K. URL http://www.sciencedirect.com/science/article/pii/000926149285201K.

P. V. Pogodin, A. A. Lagunin, D. A. Filimonov, and V. V. Poroikov. PASS Targets: Ligand-based multi-target computational system based on a public data and naïve Bayes approach. *SAR and QSAR in Environmental Research*, 26(10):783–793, October 2015. ISSN 1062-936X. doi:10.1080/1062936X.2015.1078407.

Mariya Popova, Olexandr Isayev, and Alexander Tropsha. Deep reinforcement learning for de novo drug design. *Science Advances*, 4(7):eaap7885, July 2018. doi:10.1126/sciadv.aap7885. URL https://doi.org/10.1126/sciadv.aap7885.

Vladimir Potemkin and Maria Grishina. Principles for 3D/4D QSAR classification of drugs. *Drug Discovery Today*, 13(21-22):952–959, November 2008. ISSN 1359-6446. doi:10.1016/j.drudis.2008.07.006.

Rosa Ramirez and Daniel Borgis. Density functional theory of solvation and its relation to implicit solvent models. *J. Phys. Chem. B*, 109(14):6754–6763, April 2005. ISSN 1520-6106. doi:10.1021/jp045453v. URL http://dx.doi.org/10.1021/jp045453v.

Bharath Ramsundar, Peter Eastman, Patrick Walters, Vijay Pande, Karl Leswing, and Zhenqin Wu. *Deep Learning for the Life Sciences*. O'Reilly Media, 2019. https://www.amazon.com/Deep-Learning-Life-Sciences-Microscopy/dp/1492039837.

Yingqing Ran and Samuel H Yalkowsky. Prediction of drug solubility by the general solubility equation (gse). *Journal of chemical information and computer sciences*, 41(2):354–357, 2001.

Milan Randic. Characterization of molecular branching. *Journal of the American Chemical Society*, 97(23):6609–6615, November 1975. doi:10.1021/ja00856a001. URL https://doi.org/10.1021/ja00856a001.

A. K. Rappe, C. J. Casewit, K. S. Colwell, W. A. Goddard, and W. M. Skiff. Uff, a full periodic table force field for molecular mechanics and molecular dynamics simulations. *Journal of the American Chemical Society*, 114(25):10024–10035, 1992. doi:10.1021/ja00051a040. URL http://dx.doi.org/10.1021/ja00051a040.

Ekaterina L. Ratkova, David S. Palmer, and Maxim V. Fedorov. Solvation thermodynamics of organic molecules by the molecular integral equation theory: Approaching chemical accuracy. *Chem. Rev.*, 115(13):6312–6356, July 2015. ISSN 0009-2665. doi:10.1021/cr5000283. URL http://dx.doi.org/10.1021/cr5000283.

Jean-Louis Reymond, Ruud van Deursen, Lorenz C. Blum, and Lars Ruddigkeit. Chemical space as a source for new drugs. *MedChemComm*, 1(1):30, 2010. ISSN 2040-2503, 2040-2511. doi:10.1039/c0md00020e.

Ann M. Richard and ClarLynda R. Williams. Distributed structure-searchable toxicity (DSSTox) public database network: A proposal. *Mutation Research*, 499(1):27–52, January 2002. ISSN 0027-5107.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, April 2010. doi:10.1021/ci100050t. URL https://doi.org/10.1021/ci100050t.

Valerie S. Rose, Ian F. Croall, and Halliday J. H. Macfie. An Application of Unsupervised Neural Network Methodology Kohonen Topology-Preserving Mapping to QSAR Analysis. *Quantitative Structure-Activity Relationships*, 10(1):6–15, 1991. ISSN 09318771, 15213838. doi:10.1002/qsar.19910100103.

B. Roux and M. Karplus. Ion transport in a model gramicidin channel. structure and thermodynamics. *Biophysical Journal*, 59(5):961–981, May 1991. ISSN 0006-3495. doi:10.1016/S0006-3495(91)82311-6. URL http://www.sciencedirect.com/science/article/pii/S0006349591823116.

Sebastian Ruder. An overview of multi-task learning in deep neural networks. *CoRR*, abs/1706.05098, 2017. URL http://arxiv.org/abs/1706.05098.

Fiorella Ruggiu, Gilles Marcou, Alexandre Varnek, and Dragos Horvath. ISIDA property-labelled fragment descriptors. *Molecular Informatics*, 29(12):855–868,

December 2010. doi:10.1002/minf.201000099. URL https://doi.org/10.1002/minf.201000099.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, October 1986. doi:10.1038/323533a0. URL https://doi.org/10.1038/323533a0.

Takaya Saito and Marc Rehmsmeier. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLOS ONE*, 10(3):e0118432, March 2015. doi:10.1371/journal.pone.0118432. URL https://doi.org/10.1371/journal.pone.0118432.

Robert E. Schapire. A brief introduction to boosting. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'99, page 1401–1406, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

J. Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117, 2015. doi:10.1016/j.neunet.2014.09.003. Published online 2014; based on TR arXiv:1404.7828 [cs.NE].

Ansgar Schuffenhauer, Peter Ertl, Silvio Roggo, Stefan Wetzel, Marcus A. Koch, and Herbert Waldmann. The scaffold tree visualization of the scaffold universe by hierarchical scaffold classification. *Journal of Chemical Information and Modeling*, 47(1):47–58, 2007. doi:10.1021/ci600338x. URL https://doi.org/10.1021/ci600338x. PMID: 17238248.

Philippe Schwaller, Theophile Gaudin, David Lanyi, Costas Bekas, and Teodoro Laino. "Found in Translation": Predicting Outcomes of Complex Organic Chemistry Reactions using Neural Sequence-to-Sequence Models. *arXiv:1711.04810 [cs, stat]*, November 2017.

Philippe Schwaller, Riccardo Petraglia, Valerio Zullo, Vishnu H. Nair, Rico Andreas Haeuselmann, Riccardo Pisoni, Costas Bekas, Anna Iuliano, and Teodoro Laino. Predicting retrosynthetic pathways using transformer-based models and a hyper-graph exploration strategy. *Chemical Science*, 11(12):3316–3325, 2020. doi:10.1039/c9sc05704h. URL https://doi.org/10.1039/c9sc05704h.

Volodymyr Sergiievskyi, Guillaume Jeanmairet, Maximilien Levesque, and Daniel Borgis. Solvation free-energy pressure corrections in the three dimensional reference interaction site model. *J. Chem. Phys.*, 143(18):184116, November 2015. ISSN 0021-9606, 1089-7690. doi:10.1063/1.4935065. URL http://scitation.aip.org/content/aip/journal/jcp/143/18/10.1063/1.4935065.

Robert P. Sheridan, Wei Min Wang, Andy Liaw, Junshui Ma, and Eric M. Gifford. Extreme Gradient Boosting as a Method for Quantitative Structure–Activity Relationships. *Journal of Chemical Information and Modeling*, 56(12):2353–2360, 2016. doi:10.1021/acs.jcim.6b00591.

Jaak Simm, Adam Arany, Pooya Zakeri, Tom Haber, Jörg K. Wegner, Vladimir Chupakhin, Hugo Ceulemans, and Yves Moreau. Macau: Scalable bayesian multi-relational factorization with side information using mcmc, 2015.

C. O. S. Sorzano, J. Vargas, and A. Pascual Montano. A survey of dimensionality reduction techniques. *arXiv:1403.2877 [cs, q-bio, stat]*, 2014.

Sergey Sosnin, Dmitry Karlov, Igor V. Tetko, and Maxim V. Fedorov. Comparative Study of Multitask Toxicity Modeling on a Broad Chemical Space. *Journal of Chemical Information and Modeling*, 2018a. ISSN 1549-9596; 1549-960X. doi:10.1021/acs.jcim.8b00685. Place: United States Publisher: United States.

Sergey Sosnin, Maksim Misin, David Palmer, and Maxim Fedorov. 3D matters! 3D-RISM and 3D convolutional neural network for accurate bioaccumulation prediction. *Journal of Physics Condensed Matter*, 30(32), 2018b. ISSN 1361-648X; 0953-8984. doi:10.1088/1361-648x/aad076. Place: United Kingdom Publisher: United Kingdom.

Sergey Sosnin, Mariia Vashurina, Michael Withnall, Pavel Karpov, Maxim Fedorov, and Igor V. Tetko. A Survey of Multi-Task Learning Methods in Chemoinformatics. *Molecular informatics*, 37, 2018c. ISSN 1868-1751; 1868-1743. doi:10.1002/minf.201800108. Place: Weinheim, Germany, Germany Publisher: Weinheim, Germany, Germany.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.

Christoph Steinbeck, Yongquan Han, Stefan Kuhn, Oliver Horlacher, Edgar Luttmann, and Egon Willighagen. The Chemistry Development Kit (CDK): An Open-Source Java Library for Chemo- and Bioinformatics. *Journal of Chemical Information and Computer Sciences*, 43(2):493–500, March 2003. ISSN 0095-2338. doi:10.1021/ci025584y.

S. C. Suddarth and Y. L. Kergosien. Rule-injection hints as a means of improving network performance and learning time. In *Neural Networks*, pages 120–129. Springer Berlin Heidelberg, 1990. doi:10.1007/3-540-52255-7_33. URL `https://doi.org/10.1007/3-540-52255-7_33`.

Masatake Sugita and Fumio Hirata. Predicting the binding free energy of the inclusion process of 2-hydroxypropyl- -cyclodextrin and small molecules by means of the MM/3D-RISM method. *J. Phys.: Condens. Matter*, 28(38): 384002, 2016. ISSN 0953-8984. doi:10.1088/0953-8984/28/38/384002. URL `http://stacks.iop.org/0953-8984/28/i=38/a=384002`.

Iurii Sushko, Sergii Novotarskyi, Robert Körner, Anil Kumar Pandey, Matthias Rupp, Wolfram Teetz, Stefan Brandmaier, Ahmed Abdelaziz, Volodymyr V. Prokopenko, Vsevolod Y. Tanchuk, Roberto Todeschini, Alexandre Varnek, Gilles Marcou, Peter Ertl, Vladimir Potemkin, Maria Grishina, Johann Gasteiger,

Christof Schwab, Igor I. Baskin, Vladimir A. Palyulin, Eugene V. Radchenko, William J. Welsh, Vladyslav Kholodovych, Dmitriy Chekmarev, Artem Cherkasov, Joao Aires-de Sousa, Qing-You Zhang, Andreas Bender, Florian Nigsch, Luc Patiny, Antony Williams, Valery Tkachenko, and Igor V. Tetko. Online chemical modeling environment (ochem): web platform for data storage, model development and publishing of chemical information. *Journal of Computer-Aided Molecular Design*, 25(6):533–554, Jun 2011. ISSN 1573-4951. doi:10.1007/s10822-011-9440-2. URL https://doi.org/10.1007/s10822-011-9440-2.

Iurii Sushko, Elena Salmina, Vladimir A. Potemkin, Gennadiy Poda, and Igor V. Tetko. ToxAlerts: A Web Server of Structural Alerts for Toxic Chemicals and Compounds with Potential Adverse Reactions. *Journal of Chemical Information and Modeling*, 52(8):2310–2316, August 2012. ISSN 1549-9596, 1549-960X. doi:10.1021/ci300245q.

Vladimir Svetnik, Andy Liaw, Christopher Tong, J. Christopher Culberson, Robert P. Sheridan, and Bradley P. Feuston. Random forest: a classification and regression tool for compound classification and QSAR modeling. *Journal of Chemical Information and Computer Sciences*, 43(6):1947–1958, November 2003. doi:10.1021/ci034160g. URL https://doi.org/10.1021/ci034160g.

Vladimir Svetnik, Ting Wang, Christopher Tong, Andy Liaw, Robert P. Sheridan, and Qinghua Song. Boosting: an ensemble learning tool for compound classification and QSAR modeling. *Journal of Chemical Information and Modeling*, 45(3):786–799, May 2005. doi:10.1021/ci0500379. URL https://doi.org/10.1021/ci0500379.

S. J. Swamidass, C.-A. Azencott, K. Daily, and P. Baldi. A CROC stronger than ROC: measuring, visualizing and optimizing early retrieval. *Bioinformatics*, 26 (10):1348–1356, April 2010. doi:10.1093/bioinformatics/btq140. URL https://doi.org/10.1093/bioinformatics/btq140.

I. V. Tetko, V. Y. Tanchuk, T. N. Kasheva, and A. E. Villa. Estimation of aqueous solubility of chemical compounds using E-state indices. *Journal of Chemical Information and Computer Sciences*, 41(6):1488–1493, 2001 Nov-Dec. ISSN 0095-2338.

Igor V. Tetko. Neural network studies. 4. Introduction to associative neural networks. *Journal of Chemical Information and Computer Sciences*, 42(3):717–728, 2002 May-Jun. ISSN 0095-2338.

Igor V. Tetko. Associative neural network. In *Methods in Molecular Biology™*, pages 180–197. Humana Press, 2008. doi:10.1007/978-1-60327-101-1_10. URL https://doi.org/10.1007/978-1-60327-101-1_10.

Igor V. Tetko and Gennadiy I. Poda. Application of ALOGPS 2.1 to predict log DDistribution coefficient for pfizer proprietary compounds. *Journal of Medicinal Chemistry*, 47(23):5601–5604, November 2004. doi:10.1021/jm049509l. URL https://doi.org/10.1021/jm049509l.

Igor V. Tetko and Vsevolod Yu Tanchuk. Application of associative neural networks for prediction of lipophilicity in ALOGPS 2.1 program. *Journal of Chemical Information and Computer Sciences*, 42(5):1136–1145, 2002 Sep-Oct. ISSN 0095-2338.

Igor V Tetko, Ola Engkvist, and Hongming Chen. Does 'big data' exist in medicinal chemistry, and if so, how can it be harnessed? *Future Medicinal Chemistry*, 8 (15):1801–1806, October 2016a. doi:10.4155/fmc-2016-0163. URL https://doi.org/10.4155/fmc-2016-0163.

Igor V. Tetko, Ola Engkvist, Uwe Koch, Jean-Louis Reymond, and Hongming Chen. BIGCHEM: Challenges and opportunities for big data analysis in chemistry. *Molecular Informatics*, 35(11-12):615–621, July 2016b. doi:10.1002/minf.201600073. URL https://doi.org/10.1002/minf.201600073.

G Thijs, W Langenaeker, and H De Winter. Application of spectrophores™ to map vendor chemical space using self-organising maps. *Journal of Cheminformatics*, 3 (Suppl 1):P7, April 2011. ISSN 1758-2946. doi:10.1186/1758-2946-3-S1-P7.

Russell S. Thomas, Richard S. Paules, Anton Simeonov, Suzanne C. Fitzpatrick, Kevin M. Crofton, Warren M. Casey, and Donna L. Mendrick. The US Federal Tox21 Program: A strategic and operational plan for continued leadership. *ALTEX - Alternatives to animal experimentation*, 35(2):163–168, April 2018. ISSN 1868-8551. doi:10.14573/altex.1803011.

Michael Thormann, David Vidal, Michael Almstetter, and Miquel Pons. Nomen Est Omen: Quantitative Prediction of Molecular Properties Directly from IUPAC Names. *The Open Applied Informatics Journal*, 1(1):28–32, December 2007. ISSN 18741363. doi:10.2174/1874136300701010028.

T. Tieleman and G. Hinton. Coursera: Neural networks for machine learning, lecture 6.5 – rmsprop, 2012. URL http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf. Accessed: 2017-04-04.

Roberto Todeschini and Viviana Consonni. *Handbook of Molecular Descriptors*. Wiley, September 2000. doi:10.1002/9783527613106. URL https://doi.org/10.1002/9783527613106.

Roberto Todeschini and Viviana Consonni. *Molecular Descriptors for Chemoinformatics*. Methods and principles in medicinal chemistry. Wiley-VCH, Weinheim, 2009. ISBN 978-3-527-31852-0.

Seiya Tokui, Kenta Oono, Shohei Hido, and Justin Clayton. Chainer: a next-generation open source framework for deep learning. In *NIPS*, 2015a.

Seiya Tokui, Kenta Oono, Shohei Hido, and Justin Clayton. Chainer: A Next-Generation Open Source Framework for Deep Learning. In *Proceedings of Workshop on Machine Learning Systems (LearningSys) in The Twenty-Ninth Annual Conference on Neural Information Processing Systems (NIPS)*, 2015b.

Jean-François Truchon, B. Montgomery Pettitt, and Paul Labute. A cavity corrected 3D-RISM functional for accurate solvation free energies. *J. Chem. Theory Comput.*, 10(3):934–941, March 2014. ISSN 1549-9618. doi:10.1021/ct4009359. URL http://dx.doi.org/10.1021/ct4009359.

Thomas Unterthiner, Andreas Mayr, Günter Klambauer, and Sepp Hochreiter. Toxicity Prediction using Deep Learning. *arXiv:1503.01445 [cs, q-bio, stat]*, March 2015.

Laurens van der Maaten. Learning a Parametric Embedding by Preserving Local Structure. In David van Dyk and Max Welling, editors, *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 384–391, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, April 2009. PMLR.

Laurens van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008. ISSN ISSN 1533-7928.

Alexandre Varnek, Denis Fourches, Dragos Horvath, Olga Klimchuk, Cedric Gaudin, Philippe Vayer, Vitaly Solov'ev, Frank Hoonakker, Igor Tetko, and Gilles Marcou. ISIDA - Platform for Virtual Screening Based on Fragment and Pharmacophoric Descriptors. *Current Computer Aided-Drug Design*, 4(3):191–198, September 2008. ISSN 15734099. doi:10.2174/157340908785747465.

Alexandre Varnek, Cedric Gaudin, Gilles Marcou, Igor Baskin, Anil Kumar Pandey, and Igor V. Tetko. Inductive transfer of knowledge: Application of multi-task learning and feature net approaches to model tissue-air partition coefficients. *Journal of Chemical Information and Modeling*, 49(1):133–144, January 2009. doi:10.1021/ci8002914. URL https://doi.org/10.1021/ci8002914.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL http://arxiv.org/abs/1706.03762.

C. Lee Ventola. The antibiotic resistance crisis: part 1: causes and threats. *P & T : a peer-reviewed journal for formulary management*, 40(4):277–283, Apr 2015. ISSN 1052-1372. URL https://pubmed.ncbi.nlm.nih.gov/25859123. 25859123[pmid].

Martin Vogt. Progress with modeling activity landscapes in drug discovery. *Expert Opinion on Drug Discovery*, 13(7):605–615, 2018. doi:10.1080/17460441.2018.1465926. URL https://doi.org/10.1080/17460441.2018.1465926. PMID: 29656681.

Modest von Korff and Thomas Sander. Toxicity-Indicating Structural Patterns. *Journal of Chemical Information and Modeling*, 46(2):536–544, March 2006. ISSN 1549-9596. doi:10.1021/ci050358k.

Milan Voršilák, Michal Kolář, Ivan Čmelo, and Daniel Svozil. SYBA: Bayesian estimation of synthetic accessibility of organic compounds. *Journal of Cheminformatics*, 12(1), May 2020. doi:10.1186/s13321-020-00439-2. URL https://doi.org/10.1186/s13321-020-00439-2.

Y. Wang, M. Zheng, J. Xiao, Y. Lu, F. Wang, J. Lu, X. Luo, W. Zhu, H. Jiang, and K. Chen. Using support vector regression coupled with the genetic algorithm for predicting acute toxicity to the fathead minnow. *SAR and QSAR in Environmental Research*, 21(5-6):559–570, July 2010. ISSN 1062-936X. doi:10.1080/1062936X.2010.502300.

Welch. A technique for high-performance data compression. *Computer*, 17(6):8–19, June 1984. doi:10.1109/mc.1984.1659158. URL https://doi.org/10.1109/mc.1984.1659158.

Harry Wiener. Structural determination of paraffin boiling points. *Journal of the American Chemical Society*, 69(1):17–20, January 1947. doi:10.1021/ja01193a005. URL https://doi.org/10.1021/ja01193a005.

Egon L. Willighagen, John W. Mayfield, Jonathan Alvarsson, Arvid Berg, Lars Carlsson, Nina Jeliazkova, Stefan Kuhn, Tomáš Pluskal, Miquel Rojas-Chertó, Ola Spjuth, Gilleain Torrance, Chris T. Evelo, Rajarshi Guha, and Christoph Steinbeck. The chemistry development kit (CDK) v2.0: atom typing, depiction, molecular formulas, and substructure searching. *Journal of Cheminformatics*, 9(1), June 2017. doi:10.1186/s13321-017-0220-4. URL https://doi.org/10.1186/s13321-017-0220-4.

Chi Heem Wong, Kien Wei Siah, and Andrew W Lo. Estimation of clinical trial success rates and related parameters. *Biostatistics*, January 2018. ISSN 1465-4644, 1468-4357. doi:10.1093/biostatistics/kxx069.

Youjun Xu, Jianfeng Pei, and Luhua Lai. Deep Learning Based Regression and Multiclass Models for Acute Oral Toxicity Prediction with Automatic Chemical Feature Extraction. *Journal of Chemical Information and Modeling*, 57(11):2672–2685, November 2017a. ISSN 1549-9596, 1549-960X. doi:10.1021/acs.jcim.7b00244.

Yuting Xu, Junshui Ma, Andy Liaw, Robert P. Sheridan, and Vladimir Svetnik. Demystifying Multitask Deep Neural Networks for Quantitative Structure-Activity Relationships. *Journal of Chemical Information and Modeling*, 57(10):2490–2504, October 2017b. ISSN 1549-960X. doi:10.1021/acs.jcim.7b00087.

Chun Wei Yap. PaDEL-descriptor: An open source software to calculate molecular descriptors and fingerprints. *Journal of Computational Chemistry*, 32(7):1466–1474, December 2010. doi:10.1002/jcc.21707. URL https://doi.org/10.1002/jcc.21707.

Li Zhang, Haixin Ai, Wen Chen, Zimo Yin, Huan Hu, Junfeng Zhu, Jian Zhao, Qi Zhao, and Hongsheng Liu. CarcinoPred-EL: Novel models for predicting the carcinogenicity of chemicals using molecular fingerprints and ensemble

learning methods. *Scientific Reports*, 7(1):2118, May 2017. ISSN 2045-2322. doi:10.1038/s41598-017-02365-0.

Yu Zhang and Qiang Yang. A survey on multi-task learning. *arXiv preprint arXiv:1707.08114*, 2017.

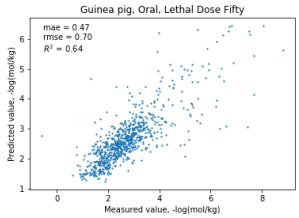
Chunyan Zhao, Elena Boriani, Antonio Chana, Alessandra Roncaglioni, and Emilio Benfenati. A new hybrid system of qsar models for predicting bioconcentration factors (bcf). *Chemosphere*, 73(11):1701–1707, 2008. ISSN 0045-6535. doi:http://doi.org/10.1016/j.chemosphere.2008.09.033. URL `http://www.sciencedirect.com/science/article/pii/S0045653508011922`.

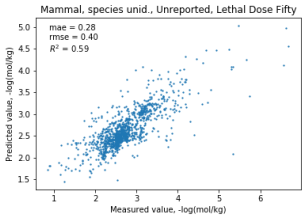
Jure Zupan and Johann Gasteiger. *Neural Networks for Chemists: An Introduction*. John Wiley Sons, Inc., USA, 1993. ISBN 3527286039.

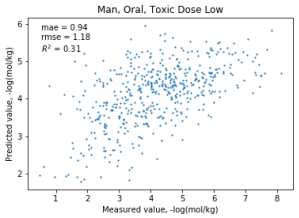
Chapter 8

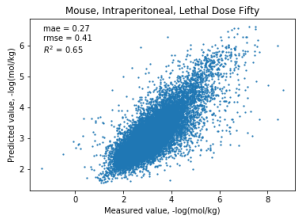
Supplementary Material

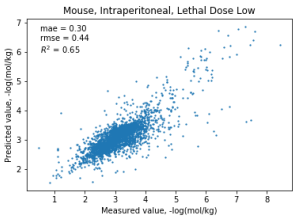
Acute toxicity endpoints prediction charts

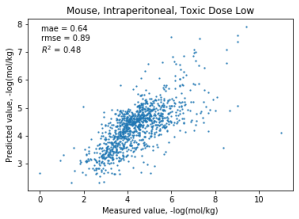


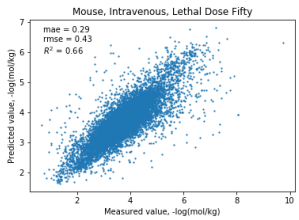


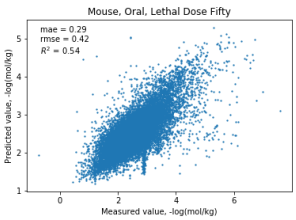


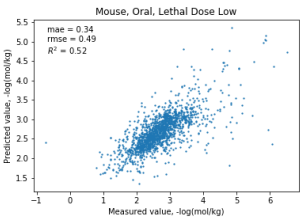


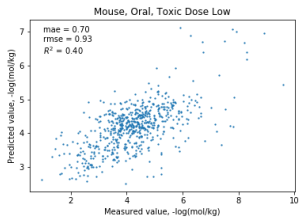


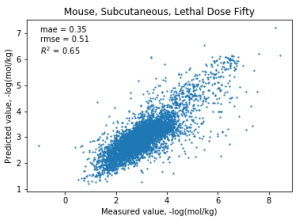


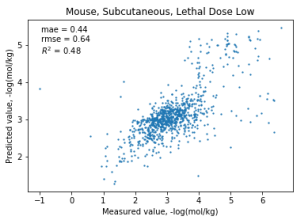


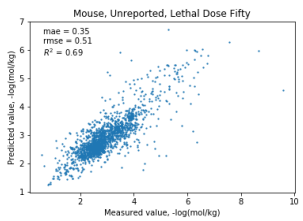


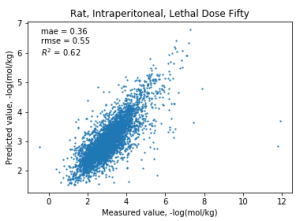


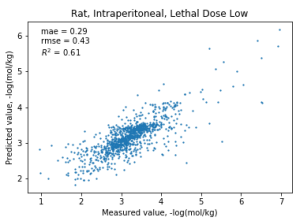


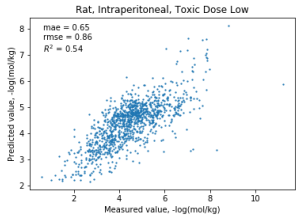


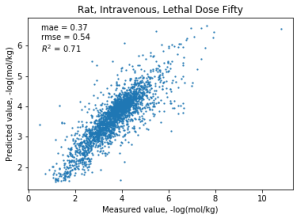


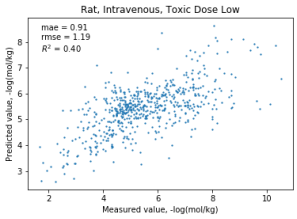


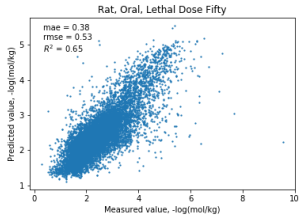


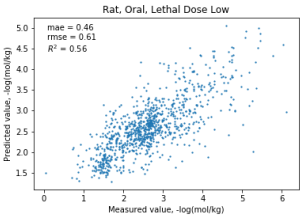


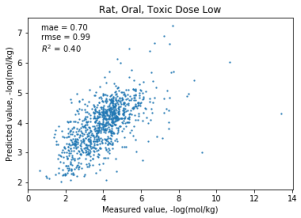

Rat, Intraperitoneal, Toxic Dose Low
mae = 0.65
rmse = 0.86
R^2 = 0.54
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)

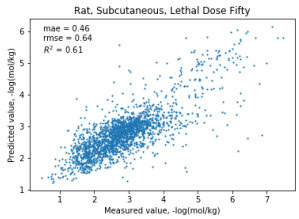

Rat, Intravenous, Lethal Dose Fifty
mae = 0.37
rmse = 0.54
R^2 = 0.71
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)

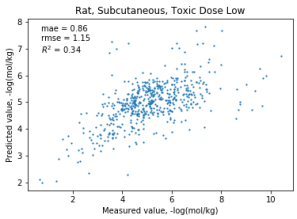

Rat, Intravenous, Toxic Dose Low
mae = 0.91
rmse = 1.19
R^2 = 0.40
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)

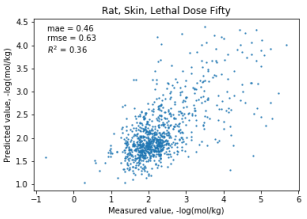

Rat, Oral, Lethal Dose Fifty
mae = 0.38
rmse = 0.53
R^2 = 0.65
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)

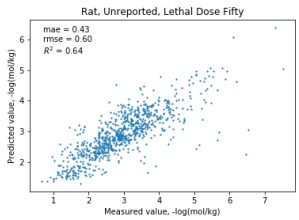

Rat, Oral, Lethal Dose Low
mae = 0.46
rmse = 0.61
R^2 = 0.56
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)

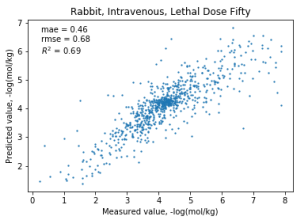

Rat, Oral, Toxic Dose Low
mae = 0.70
rmse = 0.99
R^2 = 0.40
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)

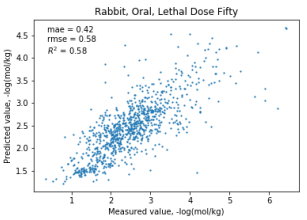

Rat, Subcutaneous, Lethal Dose Fifty
mae = 0.46
rmse = 0.64
R^2 = 0.61
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)

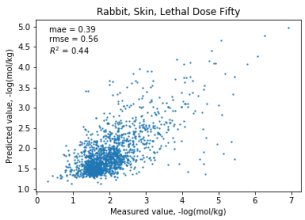

Rat, Subcutaneous, Toxic Dose Low
mae = 0.86
rmse = 1.15
R^2 = 0.34
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)


Rat, Skin, Lethal Dose Fifty
mae = 0.46
rmse = 0.63
R^2 = 0.36
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)


Rat, Unreported, Lethal Dose Fifty
mae = 0.43
rmse = 0.60
R^2 = 0.64
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)


Rabbit, Intravenous, Lethal Dose Fifty
mae = 0.46
rmse = 0.68
R^2 = 0.69
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)


Rabbit, Oral, Lethal Dose Fifty
mae = 0.42
rmse = 0.58
R^2 = 0.58
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)


Rabbit, Skin, Lethal Dose Fifty
mae = 0.39
rmse = 0.56
R^2 = 0.44
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)

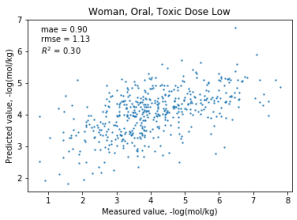

Woman, Oral, Toxic Dose Low
mae = 0.90
rmse = 1.13
R^2 = 0.30
Predicted value, -log(mol/kg)
Measured value, -log(mol/kg)

Table 8.1: The architecture of our encoding ANN for parametric t-SNE projection

Layer	Neurons	Batch Normalization
Input	2048	Yes
1	1024	Yes
2	1024	Yes
3	1024	Yes
Output	2	No

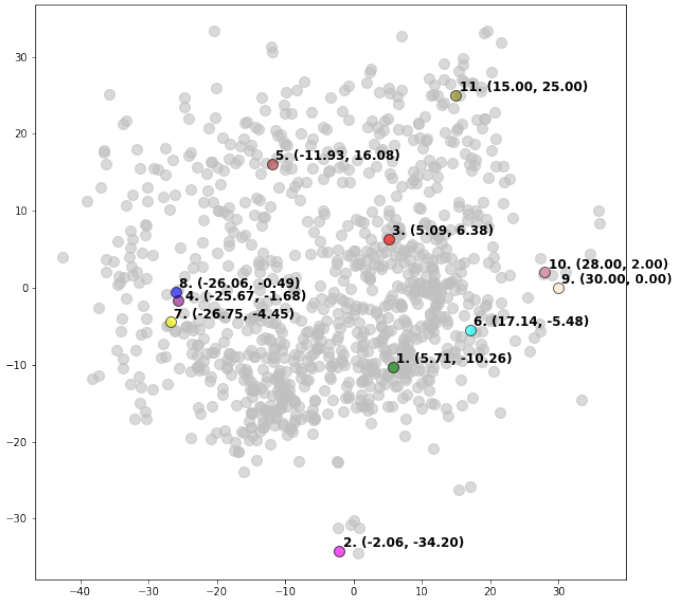


Figure 8-2: Chemical space of 1000 random points from *250k* dataset (gray). Points 1-8 are the reference structures. Points 9-11 do not correspond to any molecules.

The architecture of our encoding ANN for parametric t-SNE projection

The structure of our neural network is presented in *Table 8.1*. To train our network we used Adam optimizer with $learning\_rate = 10^{-5}$ Neural networks training was performed using PyTorch 0.4 [https://pytorch.org/] with NVIDIA GeForce GTX 1080 Ti (Driver Version 390.42, CUDA V8.0.61). The parameters provided the best performance are listed in *Table 5.1*. The grid search space are given in *Table 8.3*.

Table 8.2: The optimal hyperparameters of classifires both found by grid search optimization and default

Descriptor set	ML method	Parameters	
		GPCR ligands	NR ligands
ECFP6 descriptors	kNN	n_neighbours = 24 weights = all_equal	n_neighbours = 9 weights = all_equal
	SVM	C = 0.015625 kernel = linear	C = 0.01562 kernel = linear
	XGBoost	learning_rate = 0.05 l2 = 0.01 max_depth = 3	learning_rate = 0.05 l2 = 1. max_depth = 3
	Random forest	n_estimators = 10 max_features = all min_sample_leaf = 10	n_estimators = 100 max_features = sqrt(all) min_sample_leaf = 100
pTSNE mapping	kNN	n_neighbours = 24 weights = all_equal	n_neighbours = 9 weights = all_equal
	SVM	C = 64 kernel = polynomial gamma = 0.001	C = 0.25 Kernel = rbf gamma = 0.003
	XGBoost	learning_rate = 0.05 l2 = 0.001 max_depth = 3	learning_rate = 0.05 l2 = 0.01 max_depth = 3
	Random forest	n_estimators = 10 max_features = all min_sample_leaf = 10	n_estimators = 300 max_features = auto min_sample_leaf = 10
PCA mapping	kNN	n_neighbours = 24 weights = all_equal	n_neighbours =9 weights = all_equal
	SVM	l2 = 0.015625 kernel = linear	l2 = 0.015625 kernel = linear
	XGBoost	learning_rate = 0.05 l2 = 0.1 max_depth = 3	learning_rate = 0.05 l2 = 1. max_depth = 3
	Random forest	n_estimators = 10 max_features = all min_sample_leaf = 10	n_estimators = 13 max_features = sqrt(all) min_sample_leaf = 100
MDS mapping	kNN	n_neighbours = 24 weights = all_equal	n_neighbours =9 weights = all_equal
	SVM	l2 = 0.015625 kernel = linear	l2 = 0.015625 kernel = linear
	XGBoost	learning_rate = 0.05 l2 = 1. max_depth = 4	learning_rate = 0.05 l2 = 1. max_depth = 3
	Random forest	n_estimators = 3000 max_features = log2(all) min_sample_leaf = 100	n_estimators = 300 max_features = all min_sample_leaf = 30

Table 8.3: Sets of parameters for grid search procedure

Method	Parameter	Values
kNN	Number of neigbors to consider	1, 3, 9, 12, 15, 18, 21, 24, 27, 30
	Distance metric	manhattan
	weights for neghbours	all equal, inverse distance
RF	Number of estimators	10, 30, 100, 300, 1000, 3000
	maximum number of features	all, sqrt(all), log2(all)
	minimum number of samples in leafs	10, 30, 100, 300
SVM	constant at L2 penalty	0.015625, 0.0625, 0.25, 1, 4, 16, 64, 256, 1024
	kernel type	linear, rbf, polynomial of degree 3
	kernel coefficient (rbf, polinomial)	0.01, 0.003, 0.001, 0.0003, 0.0001, 0.00003, 0.00001
XGBoost	booster	gbtree
	learning rate	0.05, 0.1, 0.15, 0.2, 0.25, 0.3
	max depth	3, 4, 5, 6, 7, 8, 9
	L2 penalty	0.001, 0.01, 0.1, 1, 10

Table 8.4: Parameters of the drug-likeness scoring function $M(m)$. If a compound satisfies a parameter the value of $M(m)$ increases to one, so the score for the compound lies between 0 (total mismatch) and 8 (ideal fit)

Parameter	**Range**
Molecular weight	$\geq$ 160 and $\leq$ 480
LogP	$\geq$ -0.4 and $\leq$ 5.6
Atom count	$\geq$ 20 and $\leq$ 70
Molar refractivity	$\geq$ 40 and $\leq$ 130
Rings number	$>$0
Number of rotatable bonds	$<$5
Number of Hydrogen Bond Acceptors	$\leq$ 10
Number of Hydrogen Bond Donors	$\leq$ 5

Table 8.5: Endpoints extracted from RTECS dataset

Species	Administration	Type of Toxicity	No. of records
Guinea pig	Oral	Lethal Dose Fifty	799
Mammal, species unid.	Unreported	Lethal Dose Fifty	1121
Man	Oral	Toxic Dose Low	512
Mouse	Intraperitoneal	Lethal Dose Fifty	37202
Mouse	Intraperitoneal	Lethal Dose Low	2965
Mouse	Intraperitoneal	Toxic Dose Low	1057
Mouse	Intravenous	Lethal Dose Fifty	17742
Mouse	Oral	Lethal Dose Fifty	24355
Mouse	Oral	Lethal Dose Low	1565
Mouse	Oral	Toxic Dose Low	646
Mouse	Subcutaneous	Lethal Dose Fifty	7221
Mouse	Subcutaneous	Lethal Dose Low	921
Mouse	Unreported	Lethal Dose Fifty	1804
Rat	Intraperitoneal	Lethal Dose Fifty	5041
Rat	Intraperitoneal	Lethal Dose Low	1029
Rat	Intraperitoneal	Toxic Dose Low	1117
Rat	Intravenous	Lethal Dose Fifty	2538
Rat	Intravenous	Toxic Dose Low	608
Rat	Oral	Lethal Dose Fifty	10743
Rat	Oral	Lethal Dose Low	966
Rat	Oral	Toxic Dose Low	955
Rat	Subcutaneous	Lethal Dose Fifty	2014
Rat	Subcutaneous	Toxic Dose Low	555
Rat	Skin	Lethal Dose Fifty	930
Rat	Unreported	Lethal Dose Fifty	838
Rabbit	Intravenous	Lethal Dose Fifty	764
Rabbit	Oral	Lethal Dose Fifty	910

Continued on next page

Table 8.5 – Endpoints extracted from RTECS dataset (continued from previous page)

Species	Administration	Type of Toxicity	No. of records
Rabbit	Skin	Lethal Dose Fifty	1734
Woman	Oral	Toxic Dose Low	490