Thesis for The Degree of Doctor of Philosophy

Learning to Make Decisions for Autonomous Drug Design

Hampus Gummesson Svensson

Department of Computer Science and Engineering
Chalmers University of Technology | University of Gothenburg
Gothenburg, Sweden, 2025

Learning to Make Decisions for Autonomous Drug Design
Hampus Gummesson Svensson

ISBN 978-91-8103-335-9

Acknowledgements, dedications, and similar personal statements in this thesis, reflect the author's own views.

Selected material from the author's Licentiate thesis [Gum23] is republished in this Ph.D. thesis.

Doktorsavhandlingar vid Chalmers tekniska högskola, Ny serie nr 5792.
ISSN 0346-718X
10.63959/chalmers.dt/5792

Department of Computer Science and Engineering
Chalmers University of Technology | University of Gothenburg
SE-412 96 Göteborg,
Sweden
Phone: +46(0)31 772 1000

Printed by Chalmers Digitaltryck,
Gothenburg, Sweden 2025.

Till min familj.
To my family.

Learning to Make Decisions for Autonomous Drug Design

Hampus Gummesson Svensson

Department of Computer Science and Engineering
Chalmers University of Technology | University of Gothenburg

Abstract

Drug design is an iterative process aimed at identifying suitable molecules for specific biological targets. Modern computer-aided drug design increasingly leverages machine learning to inform decision-making throughout this process. However, a key challenge remains: the interactive acquisition of new knowledge to improve machine learning models using relevant data. This thesis examines sequential decision-making problems in machine learning for optimizing data collection strategies in computer-aided drug design.

To experimentally test a molecule's properties, it must first be synthesized through a sequence of chemical reactions to obtain the desired product. Machine learning can identify and validate suitable chemical reactions by predicting reaction outcomes, but this approach requires sufficient data for each reaction type of interest. This thesis presents work that combinatorially investigates different aspects of active learning to improve predictive capabilities for determining whether a reaction will produce a sufficient amount of product. In practice, only a limited number of molecules can be synthesized per design cycle due to cost and time constraints, whereas current generative models can produce numerous molecular candidates. Therefore, another work in this thesis investigates how to optimally select which generated molecules to test, given a constrained experimental budget. We formulate this challenge as a multi-armed bandit problem and propose a novel algorithm to address it.

To generate novel molecules with desired predicted properties, previous research has successfully employed reinforcement learning to align generative model outputs to a specific biological target. This thesis examines additional perspectives on applying reinforcement learning to sequentially utilize and collect target-specific data. We present a systematic comparison of various reinforcement learning algorithms for generating drug molecules and investigate methods for effectively learning from generated samples. Moreover, designing a diverse set of promising molecules is crucial for a successful drug discovery pipeline. Therefore, we propose new methods to enhance chemical exploration by adaptively modifying the reward signal. We also introduce a mini-batch diversification framework for on-policy reinforcement learning and apply it to molecular generation, thereby improving chemical exploration during the generative process. Together, these contributions advance sequential decision-making in drug design by optimizing the acquisition of new data.

Keywords: reinforcement learning, active learning, multi-armed bandits, *de novo* drug design, reaction yield prediction, chemical exploration

List of Publications

This thesis is based on the following papers and manuscript produced during the author's PhD studies.

[**Paper 1**] S. Viet Johansson*, **H. Gummeson Svensson***, E. Bjerrum, A. Schliep, M. Haghir Chehreghani, C. Tyrchan, O. Engkvist, *Using Active Learning to Develop Machine Learning Models for Reaction Yield Prediction*. Molecular Informatics, 41(12), 2200043, 2022.

[**Paper 2**] **H. Gummesson Svensson**, E. Jannik Bjerrum, C. Tyrchan, O. Engkvist, M. Haghir Chehreghani, *Autonomous Drug Design with Multi-Armed Bandits*. Proceedings of the 2022 IEEE International Conference on Big Data (IEEE Big Data 2022), 5584-5592, 2022.

[**Paper 3**] **H. Gummesson Svensson**, C. Tyrchan, O. Engkvist, M. Haghir Chehreghani, *Utilizing Reinforcement Learning for Drug Design*. Machine Learning, 113(7), 4811-4843, 2024.

[**Paper 4**] **H. Gummesson Svensson**, C. Tyrchan, O. Engkvist, M. Haghir Chehreghani, *Diversity-Aware Reinforcement Learning for de novo Drug Design*. Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2025), 9194-9204, 2025.[1]

[**Paper 5**] **H. Gummesson Svensson**, O. Engkvist, J. P. Janet, C. Tyrchan, M. Haghir Chehreghani, *Diverse Mini-Batch Selection in Reinforcement Learning for Efficient Chemical Exploration in de novo Drug Design*. Submitted, under review.

*Equal contribution.

[1] Also presented at the 18th European Workshop on Reinforcement Learning (EWRL 2025).

Summary of Contributions

The contributions of the appended papers are listed below:

[**Paper 1**] Co-designed the study, co-developed the code, jointly performed the experiments, collaboratively analyzed the results, and co-wrote the manuscript.

[**Paper 2**] Co-designed the study, developed the code, performed the experiments, analyzed the results, and wrote the manuscript.

[**Paper 3**] Co-designed the study, developed the code, performed the experiments, analyzed the results, and wrote the manuscript.

[**Paper 4**] Co-designed the study, developed the code, performed the experiments, analyzed the results, and wrote the manuscript.

[**Paper 5**] Co-designed the study, developed the code, performed the experiments, analyzed the results, and wrote the manuscript.

Acknowledgment

First and foremost, this thesis would not have been possible without the support of my academic supervisor, Morteza Haghir Chehreghani. Thank you, Morteza, for all the interesting discussions and for being the reliable supervisor I needed. I would also want to thank Ola Engkvist for always taking the time to listen to my ideas and for your constructive feedback. I am also grateful to Christian Tyrchan for your advice and interesting ideas. I would also like to express my gratitude to my co-supervisor, Alexander Schliep, and examiner, Graham Kemp, for their crucial guidance and support. Thank you all for believing in me.

I would also want to thank all my colleagues at DSAI and Molecular AI. All of you have made my PhD journey fun and memorable. Thank you all for creating such a friendly environment where it is possible to develop as both an individual and a researcher. Especially, I appreciate all my PhD colleagues and their support. Doing a PhD is very special, and it is a privilege to have spent this time with like-minded, thoughtful people who are also on the same journey.

This work would not have been possible without the support of my family and friends. I am extremely grateful to my best friend and wife, Sophia, for her endless emotional support and for always providing new perspectives. Thank you for all the small things you do and for being a reliable life partner. Thanks to my son, Lorentz, for always lighting up the darkest days and for making me understand what is most important in life. I am also truly grateful to my mum, Carola, for her support and for always teaching me new things. I would also like to express my deepest gratitude to my brother, Hannes, for being the best brother one can wish for. I am grateful for the support from all of my friends.

This work was partially supported by the Wallenberg Artificial Intelligence, Autonomous Systems, and Software Program (WASP), funded by the Knut and Alice Wallenberg Foundation, Sweden. Thank you for making this possible and for providing a platform to meet other ambitious people. The computations of this thesis were partially enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

Contents

Part I

Introductory Chapters

Chapter 1

Introduction

Developing a new drug is a complex process that can take up to a decade and cost more than US $1 billion [WML20; Pau+10]. A drug discovery campaign is initiated when a disease with an unmet need for medication has been identified [Hug+11]. The next step in this campaign involves identifying a specific biological target—such as a protein, gene, or RNA molecule—that, when modulated by the drug candidate, will produce the desired therapeutic response for the disease under investigation. Once a target has been identified, comprehensive validation must be completed to gain sufficient confidence that the target produces the desired therapeutic response, preferably using multiple validation approaches.

Following biological target identification and validation, the objective is to identify molecules exhibiting the desired therapeutic response. Bioactivity, or biological activity, describes how a drug interacts and influences living organisms, including both its beneficial and adverse effects. Molecules demonstrating significant activity against the validated target during initial screening are classified as active molecules, or *actives* for short. These actives subsequently undergo additional testing, during which those confirmed to have the desired bioactivity are classified as *hit* molecules. A commonly used screening method is high-throughput screening (HTS), which tests an extensive collection of molecules against the validated target, usually conducted in parallel in wells of a microtitre plate by a robotic system [Mac+11; Wil+17]. Once multiple hit molecules have been identified, the most promising candidates are selected, where it is essential to select structurally diverse molecules to maximize the probability of success in the subsequent steps. Next, the promising hits are refined to enhance their desired effects on the biological target, producing so-called lead molecules. When such lead molecules have been developed, their drug-like properties are optimized, e.g, lowering the concentration needed to obtain a desired response against the biological target. From the optimized lead molecules, a preclinical candidate and a backup candidate are usually selected [VS17]. This is the end of the drug discovery process (illustrated in Fig. 1.1) and the beginning of the drug development process, where the goal is to transform the selected lead molecule into a commercial medicine. The work

Figure 1.1: The drug discovery process. UMN, unmet medical needs.

of this thesis regards drug discovery and, in particular, drug design.

Throughout this process, numerous decisions are made, potentially with a significant impact on future decisions, requiring informed decision-making. Drug design is a crucial part of the drug discovery process, involving the design of new drug molecules or the improvement of existing ones. The primary goal is to design a molecule that both produces a desired response at low concentrations on a biological target and is free from side effects. Thus, drug design is involved in the hit identification, lead generation and lead optimization steps described above.

Drug design is an iterative process involving trial-and-error testing; hence, sequential decision-making is a natural part of it. The drug design process is therefore often modeled as the so-called Design-Make-Test-Analyze (DMTA) cycle, illustrated in Fig. 1.2, where each cycle acquires new knowledge to guide the design choices [Plo+12]. The DMTA cycle begins with the *Design* step, in which the goal is to develop molecules expected to exhibit the desired experimental properties. The designed molecules proceed to the *Make* step, where the molecules are synthesized. Successfully synthesized molecules advance to the *Test* step, where the molecules are experimentally tested to determine their properties. In the final *Test* step, the acquired knowledge is analyzed and summarized for the next cycle. This systematic cycle continues until a sufficient number of molecules meeting predetermined candidate criteria has been obtained.

Conventional drug design involves human expertise in designing, synthesizing, and testing new molecules. Human experts play a key role in the decision-making process for designing new drugs, which have so far enabled the discovery of thousands of approved medicinal drugs that both save lives and improve the quality of life for humans. It has been estimated that the ensemble of academic, commercial, and proprietary chemical databases includes a magnitude of 10^8 existing chemical compounds [RA12]. In contrast, only the number of synthesizable, drug-like molecules is estimated to be in the order of 10^{33} [PMV13]. Thus, conventional drug design methods seem to concentrate on a relatively small fraction of the chemical space, and the entire space is too large for exhaustive screening.

Nowadays, sophisticated machine learning and automation for drug design constitute fundamental strategies to enhance productivity in pharmaceutical research [Vam+19; Sch18]. Significant advances have occurred in recent years in applying machine learning to drug design, especially with advances in deep learning [Che+18]. These recent advances can enable a closed-loop drug design platform, where drug molecules are designed in an automated system under human supervision; however, no such system has been achieved to date [Bil+22]. For such a system to be achieved, it must be able to make several decisions on

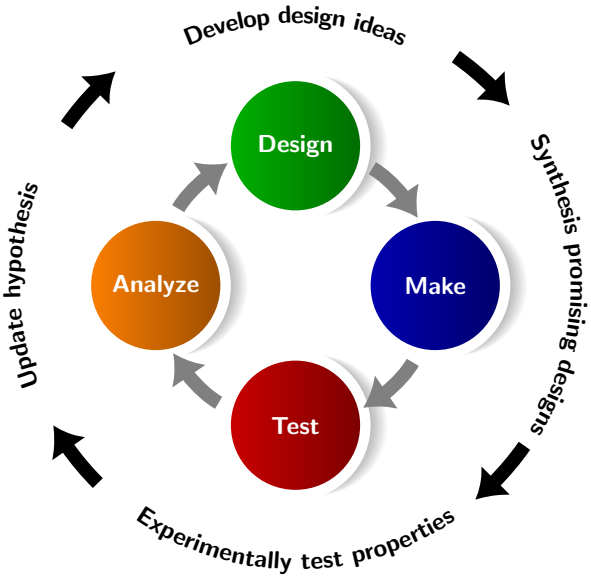


Figure 1.2: The Design-Make-Test-Analyze cycle utilized in drug design.

its own, such as where in chemical space to focus the search for novel molecules, which molecules to synthesize from the search, and how to synthesize them. Experts in the field have argued that a closed-loop platform is necessary for machine learning to make an impact in drug discovery [Sai+19]. This thesis focuses on sequential decision-making in computer-aided drug design to enable autonomous drug design, where human experts oversee the process.

A central advancement of this thesis is the use of generative models to design molecules that exhibit desired properties. In contrast to human experts proposing new molecules, these methods leverage deep learning to optimize predicted property values that correspond to the desired experimental properties. Since their recent introduction, a vast number and variety of generative models have been applied to molecular design [Bil+22]. However, the quantitative structure-activity relationship (QSAR) models utilized to predict the desired properties are incomplete or introduce uncertainties and biases due to limited training data [Ren+19]. Thus, it is crucial not only to discover high-quality molecules—those that fulfill the desired predicted properties—but also to find structurally diverse molecules to increase the likelihood of identifying potential drug candidates. This thesis specifically studies the use of *reinforcement learning* [SB18] in the context of fine-tuning a generative model for drug design. In reinforcement learning, an artificial agent interacts with an environment and receives feedback in the form of rewards. The goal of the agent is to learn a policy for acting in the environment that maximizes the agent's reward over time. We explore different perspectives on deploying reinforcement learning to improve drug design efficiency. In particular, this thesis demonstrates that it

is beneficial to learn from both high- and low-rewarding molecules, and that appropriately modifying the environment's original reward and learning from a diverse batch substantially improves chemical exploration.

Given a set of generated molecules predicted to have promising properties, not all of them can be experimentally tested to verify their properties and acquire new knowledge. A human expert usually decides which molecules to make, but to allow for an autonomous system, this decision must be learned. Therefore, this thesis introduces the learning problem of what to make next and shows how it can be formulated as a multi-armed bandit problem [Sli24]. In particular, we propose an algorithm to solve this problem and demonstrate how it can balance the trade-off between verifying promising molecules and acquiring new knowledge.

When it has been decided on which molecules to make, the next decision is on *how* to make them. Thus, another recent advancement, central to this thesis, is the use of machine learning to assist in deciding how to synthesize molecules [CGJ18]. For instance, instead of a human expert performing experiments to test different synthetic routes to a target molecule, machine learning can be used to virtually validate reaction outcomes. This requires labelled data on the reaction types of interest, preferably including both unsuccessful and successful outcomes. This thesis studies best practices for *active learning* [Set09] to enhance predictions of reaction outcomes. In active learning, labels of training instances are queried based on how much they are expected to improve the predictive capabilities of the machine learning model. This is especially useful when new labels are expensive to obtain, e.g., in scientific experimentation. Therefore, we study different aspects of active learning for predicting reaction outcomes, including how it is affected by the initial training data and the machine learning model. This thesis demonstrates that active learning can be used to predict whether a reaction is successful while requiring fewer new training labels.

This thesis studies various sequential decision-making tasks in drug design and is structured as follows. The first part of the thesis comprises the introductory chapters. Chapter 2 provides a brief introduction to the use of machine learning in drug design, including relevant chemoinformatics and computer-aided drug design concepts. Subsequently, Chapter 3 introduces the sequential decision-making problems in machine learning that are studied in the appended papers. This includes a concise introduction to active learning, multi-armed bandit, and reinforcement learning problems. The purpose of these two chapters is to aid in the understanding of the appended papers. Furthermore, Chapter 4 introduces the challenges and consequent research questions considered in this thesis. Chapter 5 summarizes the problems, methods, results, and contributions of the appended papers. Chapter 6 concludes the main research outcomes of the appended papers and discusses possible future directions. The second part of this thesis comprises the five appended papers.

Chapter 2

Machine Learning in Drug Design

This chapter introduces computational representations and methods that are central to this thesis.

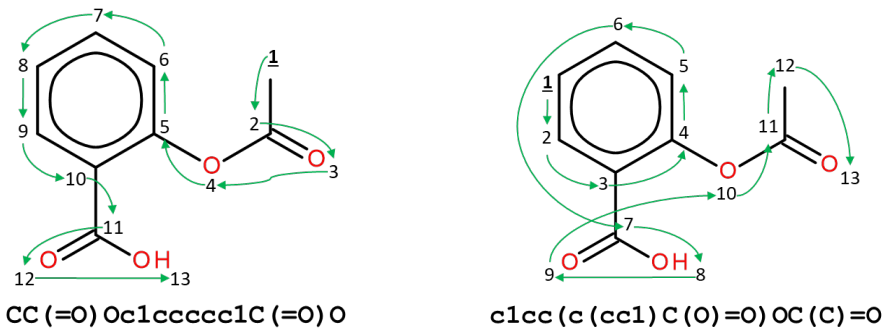
2.1 Data Curation

Modern methods often require a vast amount of data, especially with the recent rise of deep learning. Large datasets utilized in drug design originate from various sources such as ChEMBL, which consists of extracted and manually curated structure-activity relationship (SAR) data from the primary medicinal chemistry and pharmacology literature [Zdr+24]. Hence, data curation is an essential aspect of machine learning in drug design. Data curation includes several steps of cleaning and standardization of the chemical data, such as the removal of mixtures, inorganics and salts, and standardization of chemical structure and bioactivity data [Tro10]. It also includes the removal of duplicates and treatment of tautomeric forms. Tautomers of a molecule only differ by an intramolecular movement of a hydrogen atom from one atom to another. Tautomers usually have different molecular fingerprints and other properties, such that similar molecules encoded as different tautomers are unintentionally considered, which can influence the predictive ability [Mar09; Mas+14]. Removal of duplicates can be accomplished by standardizing the representation of the chemical structure, e.g., canonicalization of SMILES string (see Section 2.2.1), and removing all chemical structures with the same representation. In addition, descriptors calculated from a 2D representation will usually recognize molecules with minor differences in 3D structure (e.g., molecules that are mirror images of each other) as duplicates [Tro10].

2.2 Molecular Representation

Computer-readable representations of molecular structures are vital for enabling machine learning-based methods in drug design. Several molecular representations are used in machine learning for drug design; see David et al. [Dav+20] for an extensive summary. As relevant background for the appended papers, this section provides a concise overview of two of them: the simplified molecular input line entry system (SMILES) and molecular fingerprints. SMILES is one among several string-based representations, while molecular fingerprints are vector-based encodings. Many molecular representations (e.g., SMILES) are based on the molecular graph representation, in which nodes and edges in a labeled graph correspond to atoms and bonds, respectively, of a molecule. The label of each node corresponds to the atom type of the atom represented by that node, while the label of each vertex corresponds to the corresponding bond type. There are several other graph representations, but they are not considered in this thesis.

2.2.1 Simplified Molecular-Input Line-Entry System



(a) Canonical representation

(b) Randomized representation

Figure 2.1: Canonical and randomized SMILES representation of Aspirin. The canonical representation assigns a canonical ordering of the atoms to provide a unique string representation for each molecule. The randomized representation assigns a random initial atom and then traverses the molecular graph starting at the corresponding node. Figure extracted from original work by [Arú+19].

The simplified molecular input line entry system (SMILES) was introduced by Weininger [Wei88] and is a popular line notation system that represents the 2-dimensional molecular graph as a linear string of characters. To obtain a SMILES representation, each atom in the molecule is first assigned a unique number in the molecular graph (hydrogen atoms are usually omitted). Subsequently, the molecular graph is traversed in the order given by the assigned numbers, appending each traversed atom and non-single bond to the string. The unique number of each atom can be assigned in different ways, leading to different atom orderings in the string representation, as illustrated in Fig. 2.1.

In the canonical SMILES representation, a deterministic algorithm computes a specific atom ordering in the molecular graph, ensuring each molecule has exactly one unique SMILES representation. In contrast, the randomized SMILES representation selects an arbitrary atom ordering, yielding a non-unique but valid string representation of the same molecular structure.

2.2.2 Molecular Fingerprints

Molecular fingerprints encode chemical structures as binary vectors or occurrence counts, where each position in the vector corresponds to a local pattern of the molecular structure [Dav+20; MCT17]. These representations are generated by systematically analyzing a molecule according to pre-defined algorithmic rules that identify and index substructure patterns. Depending on which types of patterns are identified in the structure, molecular fingerprints can be categorized into two types: (1) circular fingerprints and (2) path fingerprints. Circular fingerprints are generated by exhaustively enumerating circular patterns around each heavy atom up to a given radius, while path fingerprints are computed by analyzing the paths of the molecular graph.

2.2.2.1 Extended-Connectivity Fingerprints

A well-known family of circular fingerprints is the extended-connectivity fingerprints [RH10], which are generated by utilizing a variant of the Morgan algorithm [Mor65]. The extended-connectivity fingerprints (ECFPs) are generated through a three-step process, as described by Rogers and Hahn [RH10]. In the first step, each atom is assigned an integer identifier, e.g., its atomic number, but ignoring hydrogen atoms and bonds. These initial atom identifiers are collected into an initial fingerprint set. In the second step, the identifier of each atom is iteratively updated to reflect the identifiers of its neighbors. This is done by each atom collecting its own and immediate neighbors' identifiers into an array and, subsequently, applying a hash function to produce a new integer identifier. The old atom identifiers are thereafter replaced by the new identifiers, and the new identifiers are added to the fingerprint set. The number of iterations of this procedure is determined by the prespecified radius of this circular fingerprint. In the third step, duplicate identifiers in the fingerprint set are removed, and the final set defines an ECFP fingerprint. Alternatively, the duplicate identifiers can be kept, thereby preserving information about multiple occurrences, yielding a final fingerprint set that defines an ECFP fingerprint with counts. To be used in practice, the remaining identifiers are usually represented by a vector, e.g., where identifier x implies that bit x is active (1) in the vector, optionally including counts of multiple occurrences. The size of the space of identifiers, which determines the size of the vector, depends on the output size of the hash function. To create practical and computationally efficient representations, ECFP fingerprints typically consist of 1024 or 2048 bits, generated by hashing into a smaller, fixed-length space. RDKit [Lan06], which is a widely adopted toolkit for cheminformatics, computes ECFP-like fingerprints denoted by "Morgan fingerprints". Thus, the work of this thesis

uses the notation of "Morgan fingerprints" when referring to the ECFP-like fingerprints computed by RDKit.

2.2.2.2 Atom-Pair Fingerprints

A group of path fingerprints is atom-pair fingerprints. Carhart, Smith and Venkataraghavan [CSV85] describe them as molecular descriptors based on substructures with the following desirable characteristics: (1) they should be intrinsic properties of a structure calculated through well-defined algorithmic procedures and not rely on any subjective assumptions of which types of functional groups and ring systems that are important; (2) they should be able to capture long-range relationships between atoms; (3) they should be generalizable to three-dimensional structures or molecular representations; (4) these substructures should be easily interpretable and their representation should be compact enough to enable efficient analysis across large chemical databases. Based on these characteristics, they define an atom-pair as a substructure composed of two non-hydrogen atoms and an interatomic separation:

$$\langle \text{atom 1 description} \rangle - \langle \text{separation} \rangle - \langle \text{atom 2 description} \rangle$$

The two atoms in an atom pair do not require a direct connection, and the $\langle$separation$\rangle$ value indicates the distance between them. The separation between the two atoms is measured by the number of atoms in the shortest bond-by-bond path containing both atoms. The $\langle$description$\rangle$ of each atom tells its chemical type, number of non-hydrogen atoms attached to it, and the number of bonding π electrons that it has (i.e., the number of valence electrons that participate in the formation of π bonds). An atom-pair fingerprint of a molecular structure is defined by its set of atom-pairs, where, in the standard form, the bits include counts over each atom-pair.

2.2.2.3 Comparing Fingerprints

To evaluate how similar two molecules are, it is common to compute the similarity (or dissimilarity) between their corresponding fingerprint sets. A common statistic for computing similarity is the Jaccard coefficient, also referred to as Jaccard similarity in the chemoinformatics community. For two finite sets A and B, e.g., sets of bits, the Jaccard coefficient [Jac01] is defined by

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}, \tag{2.1}$$

which gives a value between 0 and 1. As a result, the dissimilarity between two sets, the so-called Jaccard distance, is obtained by $d_J(A, B) = 1 - J(A, B)$. This statistic was independently formulated by Tanimoto [Tan58] and is therefore also known as the Tanimoto coefficient. In this thesis, we use Jaccard and Tanimoto interchangeably to refer to this similarity statistic.

Another common similarity statistic is the Dice coefficient (or Dice similarity) [Dic45; Sör48], defined by

$$DSC(A, B) = \frac{2|A \cap B|}{|A| + |B|}, \tag{2.2}$$

where $|A|$ and $|B|$ are the cardinalities of the two sets (i.e., the number of non-zero bits). This yields a similarity measure ranging from 0 to 1. In RDKit [Lan06], both the Tanimoto and Dice similarity implementations count duplicate bits (only relevant if counts are included in the fingerprints) multiple times if they are duplicates in both sets.

Both the Tanimoto coefficient and the Dice coefficient are widely adopted for fingerprint-based similarity calculations [BRH15; Tod+12]. The Tanimoto coefficient is commonly adopted for ECFPs, while the Dice coefficient is typically used for atom-pair fingerprints. Note that identical fingerprints do not imply that the corresponding molecules have identical structures, since different characteristics in the molecular structure can lead to the same bit being active.

2.3 Molecular Diversity

Molecular diversity is of utmost importance in drug design, since a local optimum in the drug design process does not necessarily translate into usefulness later in the drug discovery pipeline. Hence, it is meaningful to measure the molecular diversity among a set of molecules. Given a set of molecules $\mathcal{M} \subseteq \mathcal{S}$, where $\mathcal{S}$ is the (drug-like) chemical space, a molecular diversity metric $\rho : \mathcal{P}(\mathcal{S}) \to \mathbb{R}_{\geq 0}$ is a function that provides a non-negative number which assesses the molecular diversity among the molecules in $\mathcal{M}$. Hu et al. [Hu+24] suggests dividing diversity metrics into two categories: reference-based and distance-based. Reference-based metrics compare a molecular set $\mathcal{M}$ with a reference set $\mathcal{R}$, which may be a set of molecules or molecular fragments. Such a reference-based metric can formally be defined by

$$\rho(\mathcal{M}; \mathcal{R}) \triangleq \sum_{r \in \mathcal{R}} \left[\exists m \in \mathcal{M} \mid r \subseteq m\right], \tag{2.3}$$

where r and m are sets themselves consisting of elements defining a molecule or fragment (e.g., the vertices and edges in a molecular graph) and $[\cdot]$ is the Iverson bracket defined by

$$[P] = \begin{cases} 1 & \text{if } P \text{ is true,} \\ 0 & \text{otherwise.} \end{cases} \tag{2.4}$$

Distance-based metrics evaluate the diversity based on the pairwise distances between the molecules in $\mathcal{M}$. Such a distance-based metric can be defined by

$$\rho(\mathcal{M}; d) \triangleq f\left(\{d(x, y) \mid \forall x \neq y \in \mathcal{M}\}\right), \tag{2.5}$$

where $d : \mathcal{S} \times \mathcal{S} \to \mathbb{R}_{\geq 0}$ measures the dissimilarity between two molecules and f is a function defined by the specific metric. The following sections introduce the concepts relevant to the diversity metrics central to this thesis. We introduce scaffold analysis, which can be used as a reference-based metric, and diverse actives, which is a distance-based metric.

2.3.1 Scaffold Analysis

The scaffold of a molecule is defined as its core structure. This is a typical structure characterizing a group of molecules. This provides a basis for a systematic investigation of molecular core structures and building blocks. A popular approach for deriving molecular scaffolds from molecules was formulated by Bemis and Murcko [BM96] and is therefore known as the Bemis-Murcko scaffold. It identifies side chain atoms in the graph representation of a molecule and removes these from the graph, as illustrated in Fig. 2.2. In this thesis, we denote the Bemis-Murcko scaffold as the *molecular scaffold*. It is also common to derive a more generic scaffold to analyze the topological relationships between molecules. A topological scaffold can be derived from the Bemis-Murcko scaffold, e.g., by converting all atom types into carbon atoms and all bonds into single bonds.

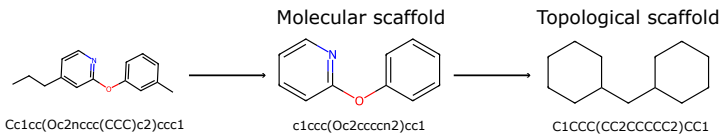


Figure 2.2: The structural formula and SMILES strings for an arbitrary molecule, and its molecular scaffold and a corresponding topological scaffold based on the Bemis-Murcko algorithm.

Once the core structure of a molecule has been derived, it is commonly used to identify structurally distinct molecules with similar activity, known as *scaffold hopping* [HSB16]. This can aid in providing several structural alternatives when designing drug molecules, e.g., utilizing computational (virtual) screening approaches. To define a reference-based diversity metric based on molecular and topological scaffolds, the reference set comprises all molecular and topological scaffolds, respectively.

2.3.2 Diverse Actives

Xie et al. [Xie+23] proposes three principles that a good diversity metric should satisfy, namely monotonicity, subadditivity, and dissimilarity. Monotonicity refers to that for any two molecular sets $\mathcal{M}_1, \mathcal{M}_2 \subseteq \mathcal{S}$ it holds that

$$\rho\left(\mathcal{M}_1 \cup \mathcal{M}_2\right) \geq \max\left(\rho\left(\mathcal{M}_1\right), \rho\left(\mathcal{M}_2\right)\right). \tag{2.6}$$

Sudadditivity of a diversity metric means that for any two molecular sets $\mathcal{M}_1, \mathcal{M}_2 \subseteq \mathcal{S}$ it holds that

$$\rho\left(\mathcal{M}_1 \cup \mathcal{M}_2\right) \leq \rho\left(\mathcal{M}_1\right) + \rho\left(\mathcal{M}_2\right). \tag{2.7}$$

Lastly, Xie et al. [Xie+23] argues that a good molecular diversity metric should also prefer dissimilar elements. Hence, for any three molecules $x, y, z \in \mathcal{S}$, if

$d(x, y) \geq d(x, z)$, it holds that

$$\rho(\{x, y\}) \geq \rho(\{x, z\}). \tag{2.8}$$

Given these principles, Xie et al. [Xie+23] proposes to study the local neighborhoods covered by a molecular set $\mathcal{M}$. In a given dissimilarity space, each molecule in $\mathcal{M}$ defines the center of a circle with a predefined radius. To measure the chemical space coverage, they define a diversity metric #Circles, which is given by counting the maximum number of mutually exclusive circles that can fit into the dissimilarity space of $\mathcal{M}$. Furthermore, Renz, Luukkonen and Klambauer [RLK24] utilizes this metric to evaluate molecular generators, referring to this metric by the number of diverse hits. For drug design, we prefer the terminology of actives, rather than hits, and therefore refer to this metric by the number of *diverse actives*. Formally, we define the number of diverse actives by

$$\mu(\mathcal{M}; D, d) = \max_{\mathcal{C} \in \mathcal{P}(\mathcal{M})} |\mathcal{C}| \text{ s.t. } \forall x \neq y \in \mathcal{C}, d(x, y) \geq D, \tag{2.9}$$

where $D \in (0, 1]$ is a distance threshold that determines the radius of each circle in the dissimilarity space. To connect this to the definition of distance-based metrics, f counts the number of non-overlapping circles.

2.4 Quantitative Structure-Activity and Property Relationship

Quantitative structure-activity and property relationships (QSA/PR) are methods that aim to predict a molecule's chemical bioactivity and physical properties, given its structure [Tyr+22]. They are regression or classification models that aim to learn relationships between molecular structures and the properties of interest. These models are based on the principle that similar molecules tend to have similar properties [BG04]. QSAR models are commonly used for the computational (virtual) screening of millions of compounds to reduce the number of candidates to be synthesized and tested experimentally, ultimately speeding up the identification of possible drug candidates [Nev+18]. Random forest models [Ho95; Bre01] are still considered the standard for QSAR methods, but gradient boosting and deep learning methods are nowadays popular alternatives to learn relationships between [Mur+20].

The molecular structure can be described by a set of descriptors that quantitatively characterize a molecule [MCT17]. Unfortunately, the set of descriptors of a molecule is non-unique, such that different descriptors can be used to describe the same molecular structure. The descriptors can encode structurally derived properties from a molecule's 2D and 3D structures, such as topological, geometrical, or electronic features. For instance, molecular fingerprints are commonly used to encode relevant structural properties. Logically, the descriptors should be selected to represent the molecular features relevant to the properties of interest [DK16]. Hence, setting up a QSAR model

requires careful consideration of both experimental errors in the data and the model's generalization errors. Experimental errors can be caused by errors in the chemical structures in the data, while generalization error is possibly caused by an insufficient relationship between the descriptor(s) of the considered molecular structures and the response variables [Tro10]. QSAR models use chemical structure descriptors, and small errors in the chemical structure can lead to a significant reduction in predictive ability [You+08]. To achieve more robust performance by relying less on manually designed descriptors for property prediction, several methods learn to extract their own descriptors from molecular structures [Kea+16; Hei+23].

2.4.1 Inverse QSA/PR

In drug design, the aim is to identify molecules with high bioactivity towards the desired target, along with other properties that make them suitable drugs. If it is possible to determine the desired properties of a drug molecule, the drug design process can entail determining structures with such properties. This is the focus of the inverse QSA/PR problem, where the aim is to identify structural features, i.e., structural descriptors, such that desired properties are fulfilled [Tyr+22]. This makes it suitable for *de novo* drug design (see Section 2.5), which aims to identify novel drug molecules that meet a set of desired properties. A fundamental problem in inverse QSAR is that structural descriptors are not continuous or unique, and the same holds for the property space. Hence, molecules with equivalent structural features can have different properties, while molecules with distinct structural features can have similar true properties.

2.5 *De Novo* Drug Design

Traditionally, virtual screening has been employed to computationally screen large libraries of commercially available drug-like molecules [McI07]. This limits the search to molecules that have been synthesized before, which leads to the discovery of molecules that are not structurally novel and can limit the extent to which a potential patent can cover the drug [Lio+14]. In contrast, *de novo* design aims to produce molecules that have not been synthesized before while satisfying predefined criteria [Tyr+22]. This can be formulated as an optimization problem in which the objective is to find a molecular structure that optimizes the ground-truth property values, as defined by the predefined criteria.

As a result of the recent progress in machine learning, especially in deep learning, deep generative models are now widely used in *de novo* drug design to traverse the drug-like chemical space [MFB21; Pan+23]. The goal is for these models to learn to effectively identify chemical structures with desirable properties. As described by Pang et al. [Pan+23], the process of generating drug-like molecules using generative models consists of several components: (1) a database of molecules; (2) a molecular representation to encode the

molecules into; (3) a generative model; (4) metrics to evaluate the properties of the generated molecules. There are a few different datasets used in molecular generation, e.g., ZINC [IS05; Irw+20], ChEMBL [Dav+15; Zdr+24] and PubChem [Kim+25]. These datasets are used as training sets to train the generative model to learn the distribution of the chemical space and the syntax of the molecular representation of choice. There are many molecular representations, of which we introduced the molecular fingerprint and SMILES representations above. In addition, graph representations are commonly used, as graphs naturally describe molecules, where each node and edge represent an atom and a bond, respectively [Mer+21]. For the generative model, a variety of different model architectures have been employed, including architectures such as variational autoencoders (VAEs), diffusion models, recurrent neural networks (RNNs) and transformers [Góm+18; APW24; Seg+18; Gre21]. These architectures can often be trained conditionally to explicitly learn specific molecular properties. Still, it is common to use machine learning techniques such as genetic algorithms, Monte-Carlo tree search, Bayesian optimization, and reinforcement learning to find molecules with specific properties, either by fine-tuning a deep generative model or by using a standalone generative model [Yos+18; Jen19; Mos+20; JBJ18; Ata+22].

The last component is the metrics used to evaluate the generated molecules. To evaluate the generated molecules as a whole, metrics such as validity, uniqueness and novelty can be used [Bro+19; Pan+23]. Validity assesses how many of the generated molecules are actually valid (e.g., follow the SMILES syntax and/or do not violate the rules of chemical valency), uniqueness assesses how many of the generated molecules are unique (and not duplicates), and novelty assesses whether the generated molecules appear in the training set. Molecular diversity is a type of metric to evaluate the structural differences among the generated molecules. As discussed in Section 2.3, the molecular diversity can be measured in different ways and depends on the molecular representation. Other metrics evaluate the properties of a single molecule and score the degree to which the property is fulfilled, e.g., quantitative estimate of drug-likeness (QED) and synthetic accessibility [Bic+12; ES09]. Moreover, for drug design, it is essential to evaluate the activity against the validated target, e.g., using a QSAR model. The scores of a single molecule are often combined to create a *scoring function*. Such a scoring function can be utilized as a reward in reinforcement learning (see Section 3.3).

This thesis uses an RNN-based model to generate SMILES representations of molecules. We focus on the problem of sequentially fine-tuning this generative model to output molecules with desired properties. Moreover, we also seek to generate a set of diverse molecules that, to a high extent, fulfill the desired properties.

2.6 Computer-Aided Synthesis Planning

Molecular synthesis is a complex and challenging task. Synthesis planning is the process of developing a sequence of chemical reactions to produce a

target molecule. A common technique for solving this problem is retrosynthetic analysis, which involves strategically deconstructing the target molecule into synthetically accessible intermediates until reaching available building blocks. After identifying synthetic routes and building blocks (reactants), they have to be verified and further optimized to yield a sufficient quantity of the desired molecule. High-throughput experimentation (HTE) is a workflow to perform multiple reactions in parallel. This approach is widely used to explore and validate different reaction mechanisms and reaction parameters to obtain an acceptable amount of product [Men+19]. This enables parallel trial-and-error testing of many reactions to obtain a target molecule. This leads to two main problems in computer-aided synthesis planning: synthetic route prediction (retrosynthesis) and forward prediction [Joh+19]. The former problem aims to predict synthesis routes and the building blocks necessary to synthesize a specific molecule. In contrast, the latter problem seeks to predict reaction outcomes given building blocks and reaction conditions (e.g., temperature, solvent, and catalyst). Thus, a forward prediction model can be employed to confirm that the proposed reaction produces the desired product and to recommend optimal reaction conditions [Sch+19].

In this thesis, we study a type of forward prediction problem, namely, reaction yield prediction [Sch+21]. Reaction yield prediction seeks to predict the yield of a reaction, which describes the quantity (usually in percentage) of the building blocks that are converted to the desired product(s) in the reaction. This is normally done by either explicitly predicting the reaction yield or predicting if the yield will reach a desired quantity. The latter is of more interest in drug discovery, where the objective is to find a successful synthetic route to experimentally test properties. The former is usually of more interest in the drug development process, where the goal is to transform a newly discovered drug into a commercial medicine, since the desired product then needs to be manufactured in a sufficient quantity. Since this thesis focuses on drug design in the context of drug discovery, the work considers reaction yield prediction to determine whether a reaction will provide a desired minimum reaction yield.

Chapter 3

Sequential Decision-Making in Machine Learning

This chapter introduces the sequential decision-making problems in machine learning central to this thesis: active learning, multi-armed bandits, and reinforcement learning.

3.1 Active Learning Problems

In supervised learning, given subsets of data instances $X \subset \mathcal{X}$ and labels $Y \subset \mathcal{Y}$, a learner chooses a mapping $h : \mathcal{X} \to \mathcal{Y}$ from a set of possible mappings $\mathcal{H}$ (called a hypothesis set). The aim is to approximate an unknown target function by learning a mapping $h \in \mathcal{H}$ that, given a data instance $x \in \mathcal{X}$, provides a prediction $\hat{y}$ of the true label $y \in \mathcal{Y}$. A suitable mapping is usually inferred by using a training set $\mathcal{L} = \{(x^{(l)}, y^{(l)})\}_{l=1}^{L} \subseteq X \times Y \subset \mathcal{X} \times \mathcal{Y}$, which consists of pairs of data instance $x^{(l)}$ and true label $y^{(l)}$. To construct such a training set, the true label $y^{(l)}$ needs to be acquired for each data instance $x^{(l)}$. In many real-world problems, the true labels are often difficult or expensive to obtain. For instance, medical image analysis requires labels from human experts, while reaction yield prediction needs wet-lab experiments to determine the true yield. To address this challenge, *active learning* is a machine learning paradigm that seeks to reduce the data labeling cost/time by actively obtaining labels for only the most informative data instances. In essence, active learning studies how to improve the generalization of the learning (i.e., identifying an appropriate mapping h) by utilizing a carefully chosen training set. This is done by the learner sequentially deciding which label(s) to query an oracle (e.g., a human annotator) based on the information acquired up to this point. On the other hand, in *passive learning*, the learner has no control over the training set, e.g., what data instance to query is randomly chosen or based on a pre-defined scheme that does not utilize the learner's current hypothesis.

Two common active learning scenarios are stream-based and pool-based active learning [Set12]. In stream-based active learning, the learner is prompted

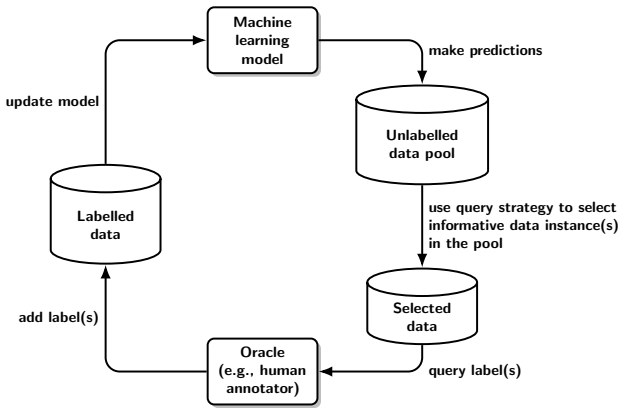


Figure 3.1: Pool-based active learning.

with one unlabelled data instance at a time and must immediately choose between two options: keep the data instance and query its label, or discard it. This is in contrast to pool-based active learning, where the learner has access to a pool of unlabelled data instances. At each iteration, the learner needs to decide which unlabelled data instance(s) from the pool to add to the training set and query their labels from the oracle. The strategy for deciding which label(s) to query is known as the *query strategy*. A well-known and established type of query strategy is *uncertainty sampling* [Yan+15; SU07]. Uncertainty sampling is based on the principle that if the learner is uncertain about the predicted label of an instance, then this instance-label pair is more informative for the learner to know, compared to a label where the predictive uncertainty is low. Thus, the goal of uncertainty sampling is to minimize the uncertainty in the predicted labels. There are different ways to quantify uncertainty, e.g., by using the predicted probabilities of a single learner or by measuring disagreement among multiple learners. This thesis focuses on uncertainty sampling using predicted probabilities in the pool-based active learning setting.

3.2 Multi-Armed Bandit Problems

Imagine going to a casino with M slot machines, also known as one-armed bandits. In each round, you can choose a slot machine to play by pulling its "lever". For each machine, there is a certain probability of winning a payout. Over a total of T rounds, you want to maximize the sum of payouts by identifying the slot machine with the highest average payout. Hence, in each round, you need to decide whether to try a new machine, play a machine that you have only tried a few times, or play the machine that has given the highest average payout so far. This is known as the multi-armed bandit (MAB) problem, first discussed by Robbins [Rob52] and later formalized by Lai, Robbins et al. [LR+85]. It models the exploitation-exploration dilemma described above, in which each round we must decide whether to remain with the current best

alternative or explore other options. A simple strategy to tackle this dilemma is to choose the action with the maximum empirical expected outcome with probability $1 - \epsilon$, and otherwise choose a random action. This is known as the epsilon-greedy strategy (denoted ϵ-greedy), where the greedy strategy (i.e., $\epsilon = 0$) always chooses the action that at the current time is believed to be optimal. By selecting the current best action, we are exploiting our existing knowledge; while when choosing a random action, we are exploring to learn more about the outcomes of all possible actions.

Algorithm 1 The Multi-Armed Bandit Problem

Input: time horizon T, policy π
Initialization: history H_0
for $t = 1, \ldots, T$ **do**
  Choose action a_t according to policy $\pi(\cdot|H_{t-1})$
  Perform action a_t and observe reward r_t
  Update history $H_t \leftarrow H_{t-1} \oplus ((a_t, r_t))$
end for

The multi-armed bandit problem is a sequential game between a learner and an environment in which the learner tries to learn the environment's probable outcomes for different actions. In general, it is possible to model numerous sequential decision-making problems as MAB problems, thereby extending the original problem, e.g., in the design of clinical trials, news recommendation, finance, navigation, and bottleneck identification [VBW15; Pre09; Li+10a; She+15; ÅCC20; ÅHC22].

The problem is illustrated in Algorithm 1, where $\oplus$ here denotes the concatenation of two histories, and is formally defined as follows. In each round $t \in [T]$, a learner chooses an action a_t from a set $\mathcal{M}$ of M possible actions, also known as *arms*. Subsequently, the learner observes a reward $r_t \in \mathbb{R}$ from the environment. The learner decides on which action a_t to choose, based on the history $H_{t-1} = ((a_1, r_1), \ldots, (a_{t-1}, r_{t-1})) \in (\mathcal{M} \times \mathbb{R})^{t-1}$ of previous actions and rewards, using a mapping from histories to actions—a *policy* π. A policy can be either stochastic (i.e., mapping to a distribution $\Delta(\mathcal{M})$ over actions) or deterministic for a fixed history H_{t-1}. The most common objective for the learner is to learn a policy that maximizes the expected cumulative reward of the learner $\mathbb{E}\left[\sum_{t=1}^{T} r_t\right]$. This is done in an unknown environment where the learner only knows that it is part of a class of environments $\mathcal{E}$, i.e., a set of possible environments.

One type of environment class is the stochastic MAB problem, commonly known as stochastic bandits, where the reward for each action a is drawn independently from a fixed probability distribution $\mathcal{D}_a$ with unknown parameter(s), e.g., a Bernoulli distribution with an unknown parameter. For each possible action $a \in \mathcal{M}$, there exists an unknown expected value $\mu_a = \mathbb{E}[\mathcal{D}_a]$ that (partially) determines the reward distribution for that action. To optimize the cumulative reward of the learner, the objective is to identify the best action $a^* = \operatorname{argmax}_{a \in \mathcal{M}} \mu_a$ (i.e., the action with the highest expected value), where

the expected value of the optimal arm is denoted by μ^*. There are several ways to measure the performance of a policy π, where the most common objective is to minimize the loss suffered by the learner relative to the optimal policy (i.e., playing the optimal arm a^*) over the time horizon T [LS20]. The loss suffered by the learner relative to the optimal policy is usually measured by the *regret*, which can be formally defined by

$$R(T) = \mu^* T - \mathbb{E}\left[\sum_{t=1}^{T} r_t\right], \tag{3.1}$$

where the expectation is taken over the learner's actions and the environment's rewards. Hereafter, for the relevance of this thesis, we focus on stochastic bandits and refer to the work by Slivkins [Sli24] and Lattimore and Szepesvári [LS20] for a comprehensive overview of different extensions of the "original" MAB problem. Below is a brief introduction to some extensions of the "original" MAB problem relevant to this thesis, based on the introduction in the appended Paper 2.

3.2.1 Contextual Bandits

Algorithm 2 The Contextual MAB Problem

Input: time horizon T, policy π
Initialization: history H_0
for $t = 1, \ldots, T$ **do**
  Observe context x_t
  Choose arm a_t according to policy $\pi(\cdot|H_{t-1}, x_t)$
  Play arm a_t and observe reward $r_t \sim \mathcal{D}_{(a_t, x_t)}$
  Update history $H_t \leftarrow H_{t-1} \oplus ((a_t, r_t, X_t))$
end for

In the contextual MAB problem, before choosing which arm to play in the current round t, the learner observes a feature vector x_t, known as the *context*. The context can either provide a single aggregated feature vector to the learner or provide feature vectors for each arm. The reward r_t in each round is assumed to be drawn independently from a distribution $\mathcal{D}_{(a_t, x_t)}$, with the expected value denoted by $\mu(a_t|x_t)$, that depends on both the observed context and the chosen action a_t. The contextual MAB problem is illustrated in Algorithm 2. The contextual MAB problem has been broadly studied under the linear realizability assumption, introduced by Abe, Biermann and Long [ABL03], where the expected reward is assumed to be linear with respect to the context vector of each arm [Chu+11; AG13; Li+10b; Aue02]. Given an action a and context x, the linear realizability assumption implies that the expected reward is of the form

$$\mu(a|x) = x \cdot \theta_a, \tag{3.2}$$

where θ_a is a fixed unknown vector specific to action a. There have been several successes in using the contextual MAB problem to model real-world applications, such as recommender systems, health applications, information retrieval and navigation [BRA20; Nil+24].

3.2.2 Multiple-Play Bandits

Algorithm 3 The Multiple-Play MAB Problem

Input: time horizon T, policy π
Initialization: history H_0
for $t = 1, \ldots, T$ **do**
  Choose super arm S_t of K base arms according to policy $\pi(\cdot|H_{t-1})$
  Play and observe reward(s) for super arm S_t
  Update history H_t
end for

Up to this point, we have assumed that the learner only chooses one arm ($K = 1$) in each round. Allowing the learner to select more than one arm in each round ($K > 1$) is known as the multiple-play MAB problem [AHT+90; KHN15], first introduced by [AVW87]. In this problem, as illustrated in Algorithm 3, a super arm $S \subset \mathcal{M}$ consisting of a combination of $K \leq M$ base arms is played in each round. In this setting, the basic actions are referred to as "base arms" to distinguish them from the super arms. In the multiple-play problem, all combinations of (unique) K base arms are usually allowed. There are mainly two reward feedback settings: full-bandit and semi-bandit. In the former setting, the learner observes the aggregated reward of the chosen super arm; in the latter, the learner observes the reward for each base arm. There are other similar problems, e.g., the combinatorial MAB problem, where certain combinations of base arms are allowed.

3.2.3 Bandits With Volatile Arms

Algorithm 4 The MAB Problem with Volatile Arms

Input: time horizon T, policy π
Initialization: history H_0
for $t = 1, \ldots, T$ **do**
  Observe available arms $\mathcal{M}_t$
  Choose arm $a_t \in \mathcal{M}_t$ according to policy $\pi(\cdot|H_{t-1}, \mathcal{M}_t)$
  Play arm a_t and observe reward r_t
  Update history H_t
end for

The standard MAB problem assumes that there is a fixed set $\mathcal{M}$ of M available arms in each round. However, in real-world applications, the set of available arms may differ between rounds, e.g., a slot machine is occupied

by another player for some round(s). Hence, in each round t, there is a set $\mathcal{M}_t \subseteq \mathcal{M}$ of available arms in this round. This setting is studied by Kleinberg, Niculescu-Mizil and Sharma [KNS10] by introducing *sleeping bandits*, where the set of available arms in each round t is chosen from a fixed and finite pool of actions by an adversary. They propose an algorithm that prioritizes playing an arm that has become available for the first time. Otherwise, it plays the arm with the largest upper confidence bound, inspired by the UCB1 algorithm [ACF02]. The volatile MAB problem explored by Bnaya et al. [Bna+13] is similar, but here each arm is associated with a lifespan during which the arm is available, such that the optimal arm may change over time. The lifespan of each arm is unknown in advance, and the goal is to play the best available arm in each round.

Both the sleeping and volatile MAB problems consider time-varying arm sets with *volatile arms* that can "appear" and/or "disappear" in each round. In this thesis, by "volatile arms" we mean that the available arms to select from in each round may change, as illustrated in Algorithm 4. The available arms in each round are unknown in advance, and the same arm can appear and disappear several times. This is a realistic setting for decision-making in drug design, where the same molecule can be suggested in various design steps DMTA cycle (see Figure 1.2), and it needs be decided if it should proceed to the test step.

3.2.4 Bandits With Similarity Information

Although an extensive collection of MAB algorithms for problems with a fixed small number of arms has been proposed in the literature, MAB problems with infinite or exponentially large arm sets are relatively under-examined. For such a problem, a common approach is to use similarity information (or a metric) between contexts and/or arms, by assuming that similar actions yield similar rewards.

Algorithm 5 Zooming Algorithm [Sli24]

Initialization: set of active arms $\mathcal{A} \leftarrow \emptyset$
for $t = 1, \ldots, T$ **do**
  ▷ Activation rule
  if some arm y is not covered by the confidence balls of active arms **then**
    Pick any such arm y and "activate" it: $\mathcal{A} \leftarrow S \cup \{y\}$
  end if
  ▷ Selection Rule
  Play an active arm $x \in \mathcal{A}$ with the largest $\text{index}_t(x)$
end for

For instance, Kleinberg, Slivkins and Upfal [KSU08] introduce the *Zooming algorithm*, where the similarity information is given as a metric space $(\mathcal{M}, \mathcal{D}_{\mathcal{M}})$ of arms [KSU19], where $\mathcal{M}$ is the set of arms and $\mathcal{D}_{\mathcal{X}}$ measures the distance between two arms in $\mathcal{M}$. The Zooming algorithm aims to approximate the expected rewards over the metric space by probing different "regions" of the

space, which leads to an adaptive partitioning of the metric space [Sli24]. At each round t, there is a set of active arms $\mathcal{A}$, determined by an activation rule. Each active arm $x \in \mathcal{A}$ covers a region of the metric space. This region is given by the confidence ball of the arm $B(x, r_t(x))$, which is a ball with the arm at its center. The radius of the ball is the confidence radius $r_t(x)$ of the empirical average reward (of the active arm) at round t. The confidence radius is related to the size of the one-sided confidence interval of the empirical average reward and guarantees, with high probability, that the difference between the true expected reward and the empirical average reward is not larger than the confidence radius. To determine what active arm to play, it chooses an arm with the largest upper confidence bound, as illustrated in Algorithm 5, similar to the arm selection of the UCB1 algorithm [ACF02].

Slivkins [Sli11] extends the Zooming algorithm to the contextual setting, introducing the contextual Zooming algorithm, where the similarity information is provided by a metric space of context-arm pairs. The work of this thesis extends the contextual Zooming algorithm to allow volatile arms and multiple-play. The context at each round specifies the available arms, but we relax the context in the similarity information and define the space of arms only by their corresponding feature vectors.

3.3 Reinforcement Learning Problems

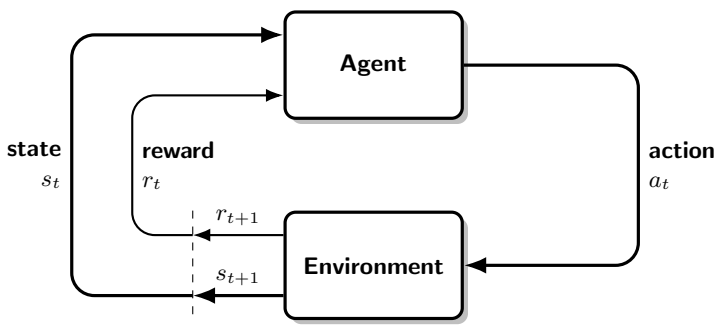


Figure 3.2: Interaction between the agent (learner) and environment in reinforcement learning problems.

A more general type of sequential decision-making problem is the reinforcement learning (RL) problem, which is extensively explained and overviewed by the work of Sutton and Barto [SB18] and Szepesvári [Sze10]. In each round t, a learner observes the current state $s_t \in \mathcal{S}$ of the environment and, given this state, chooses an action $a_t \in \mathcal{M}$ using a policy $\pi(\cdot|s_t) : \mathcal{S} \mapsto \mathcal{M}$. The policy is a learnable mapping from states to possible actions, or a probability distribution over the action space $\mathcal{M}$. Subsequently, a reward $r_{t+1} \in \mathbb{R}$ and new state $s_{t+1} \in \mathcal{S}$ is observed by the learner. In RL problems, the learner is often referred to as the *agent*. The learner's objective is to learn an optimal policy. Optimality is usually measured in terms of maximization of the discounted

future reward, defining the (infinite) return G_t following round t

$$G_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}, \tag{3.3}$$

where γ is the discount factor, $0 \leq \gamma \leq 1$, which is used to penalize the uncertainty of future rewards. For an infinite number of rounds, the return could itself be infinite, which is not desired since we want to maximize it. This can be handled using the discounted cumulative reward, where $\gamma < 1$. Multi-armed bandit problems, as described above, can be seen as reinforcement learning problems with only one state. However, in the MAB problem, the history $H_{t-1} = ((a_1, r_1), \ldots, (a_{t-1}, r_{t-1}))$ of previous actions and rewards can be considered as the *information state*, but this "state" does not affect the behaviour of the environment. In both the RL problem and the MAB problem, an action affects the reward in the current round; in the RL problem, it can also affect the next state and future states of the environment.

A reinforcement learning problem is often formalized by a Markov decision process (MDP), where Figure 3.2 illustrates this learner-environment interaction. A Markov decision process is described by a tuple $(\mathcal{S}, \mathcal{M}, P_a, \gamma)$. As introduced above, $\mathcal{S}$ is the set of states, $\mathcal{M}$ is the set of possible actions, and γ is the discount factor. Furthermore, $P_a(r, s, s') = \Pr(s_{t+1} = s', r_{t+1} = r | s_t = s, a_t = a)$ is the probability of observing s' and r as the next state and reward, respectively, when at state s and performing action a. The state transitions of the MDP satisfy the Markov property since, given the current state s and action a, the probability of the next state s' and reward r is independent of all previous states and actions.

Whereas the reward signal provides feedback on the immediate outcome of a policy, it is also necessary to assess its long-term performance. This is usually done by using a statistic of the expected future random rewards produced by the interactions between the learner's policy and the environment. The first such statistic is the expected return following a policy π starting at state s, known as the *state-value* $v_\pi(s)$, and is defined as follows

$$v_\pi(s) = \mathbb{E}_\pi \left[G_t | s_t = s \right] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s \right]. \tag{3.4}$$

In the same way, the expected return of taking action a at state s and subsequently following a policy π is known as the *action-value*, and is defined as follows

$$q_\pi(s, a) = \mathbb{E}_\pi \left[G_t | s_t = s, a_t = a \right] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s, a_t = a \right]. \tag{3.5}$$

Many RL algorithms strive to estimate the state-value and state-action functions from experiences (i.e., interactions with the environment) to improve and assess a policy's behaviour.

There are two main classes of algorithms to solve reinforcement learning problems, as illustrated in Figure 3.3: *model-based* and *model-free* algorithms.

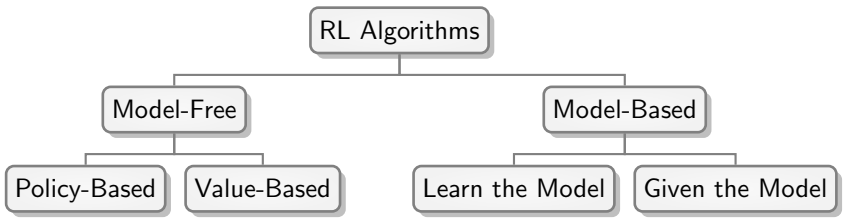


Figure 3.3: Taxonomy of reinforcement learning algorithms [ZY20].

Model-based algorithms either learn or have access to a model of the environment [ATB17; Sil+17; Kai+20; Chu+18]. By using a model of the environment that infers state transitions and rewards, it is possible to learn without interacting with the real environment, thereby reducing sample complexity. However, in many real-world applications, a model is not readily available and can be difficult to learn. Also, learning a model increases computational complexity, since both an environment model and a policy are learnt, and the policy's performance depends heavily on the accuracy of the environment model. Model-free algorithms do not use a model of the environment; instead, they only learn from experiences by interacting with the real environment [Mni+13; Sut+99]. This reduces computational complexity and enables adaptability in complex environments where accurately modeling the environment is challenging.

There are two main types of model-free algorithms: *policy-based* and *value-based* algorithms. The objective of value-based algorithms is to learn the state-action function $q_{\pi^*}(s, a)$ of the optimal policy π^* and use the learned state-action function to interact with the environment. If the state-action function of the optimal policy is known, it is possible to determine the optimal action at each state, learning a deterministic policy. Value-based algorithms usually use a more explorative policy, known as the *behavior policy*, to interact with the environment, in contrast to the policy we aim to learn, the so-called *target policy*. For policy-based algorithms, the optimal policy is learned by directly parameterizing it, often denoted π_θ with parameters θ. This is also known as *policy optimization* since the objective is to optimize the policy to maximize the return G_t. Learning a parameterized policy enables the agent to build a stochastic policy directly. It is especially more efficient for large action spaces, where it is inefficient or infeasible to estimate the state-action value $q_\pi(s, a)$ for each action. In policy-based algorithms, the policy can be updated based on experiences gathered by another policy (e.g., the policy from a previous round) or by the current policy.

Learning a policy from experiences gathered by another policy is known as *off-policy* learning, where the experiences used for learning are often sampled from a memory of past and current experiences. On the contrary, in *on-policy* learning, the same policy is used both to gather experiences from the environment and to update the policy based on those experiences. In off-policy learning, different target and behavior policies are used, whereas the same policy is used for both in on-policy learning. In this thesis, we consider only online RL, which at each round produces new experiences by interacting

with the environment; in contrast, offline RL relies on limited training sets of experiences.

To scale to problems with large action and/or state spaces, it is common to use neural networks as function approximators. Neural networks can approximate a wide range of functions [HSW89], and most modern reinforcement learning algorithms use deep neural networks to learn policies, value functions and models to generalize to unobserved or rarely observed state-action pairs [Sil+17; Lil+15; Aru+17]. This gives rise to the term deep reinforcement learning, which refers to reinforcement learning algorithms that use deep learning methods for function approximation. Many deep reinforcement learning algorithms utilize a technique called *experience replay*, where past experiences are stored in a replay buffer and are replayed during the learning phase [Sil+17; Wan+16]. Hence, the learning is averaged over its previous states and actions. This provides a way to remove correlations of trajectories used for updates, while not forgetting possibly rare experiences, ultimately stabilizing the learning [Sch+15].

The focus of this thesis is on policy-based algorithms that utilize neural networks. In particular, we consider policy optimization algorithms based on popular algorithms such as Proximal Policy Optimization (PPO) [Sch+17], Advantage Actor-Critic (A2C) [Mni+16a], Soft Actor-Critic (SAC) [Haa+18] and Actor-Critic with Experience Replay (ACER) [Wan+16]. In addition, for *de novo* drug design, the RL algorithm proposed by Olivecrona et al. [Oli+17] has been widely used in previous work. Several of our works center around this algorithm to fine-tune a molecular generative model. We denote this algorithm as *regularized maximum likelihood estimation* (Reg. MLE). Given a sequence of actions $a_{1:T} := (a_1, a_2, \ldots, a_T)$, the algorithm seeks to learn a optimal parameterized policy π_θ by minimizing the following loss

$$\mathcal{L}(\theta) = \left(\log \pi_{\text{prior}}(a_{1:T}) + \sigma r_{a_{1:T}} - \log \pi_\theta(a_{1:T})\right)^2, \tag{3.6}$$

where $\log \pi(a_{1:T}) := \log \pi(a_1) + \sum_{t=2}^{T} \log \pi(a_t | a_{1:t-1})$, $r_{a_{1:T}}$ is the cumulative reward for the action sequence $a_{1:T}$, π_{prior} is the initial policy obtained by self-supervised pre-training on an unlabelled dataset and σ is a tunable hyperparameter.

3.3.1 Exploration Techniques in Reinforcement Learning

One of the fundamental challenges in RL is achieving an optimal balance between extensive exploration of the state-action space and taking actions known to yield high (future) rewards. This exploration-exploitation dilemma is a critical trade-off that significantly affects the efficacy and convergence properties of reinforcement learning algorithms. Two significant environmental challenges to efficient exploration in deep reinforcement learning are a large state-action space and sparse rewards [Hao+23]. A large state-action space makes it intractable to explore all possible state-action pairs; instead, the learner must rely on generalization to decide which parts of the space to focus on. Similarly, when rewards are sparse—i.e., the learner receives feedback (a reward) only at the end of an episode rather than at every step—the learner

receives little guidance on how to improve its actions. Both these challenges are relevant to the work of this thesis. For instance, in RL-based *de novo* drug design, the state space consists of all possible substructures that can be created by the given actions, leading to a large state space; and a molecule can only be evaluated when its whole structure (a full episode) has been generated, such that no intermediate rewards are provided. In *de novo* drug design, the large parts of the state are often low-rewarding (close to zero reward), which makes it even more challenging to find promising solutions when rewards are sparse. Also, when rewards are sparse, it is difficult to determine which actions in an episode led to a specific outcome, a problem known as the credit assignment problem in reinforcement learning.

There are several techniques for promoting exploration in the state-action space. One common technique is based on the concept of *intrinsic motivation* [Sch91], which aims to push an agent to exhibit a specific behaviour without direct environmental feedback [AMH19]. Intrinsic motivation is derived from an intrinsic reward, which should motivate the agent to do something for its inherent satisfaction rather than for the prospect of a higher reward. In this way, the agent can be motivated to explore the state-action space and, consequently, find more rewarding solutions [Bel+16]. The environment's original reward function R is modified by adding an exploration bonus R_I, and this new reward $R + R_I$ is instead used in the learning process. To design the intrinsic rewards, one popular technique is to estimate the state novelty to motivate the agent to visit less-visited states [Hao+23]. For instance, random network distillation (RND) introduced by [Bur+19] is a popular method that estimates state novelty from the prediction error between a fixed random network and a learnable network.

A domain-specific exploration technique that has become popular in RL-based *de novo* drug design is the memory-assisted technique proposed by Blaschke et al. [Bla+20]. The proposed technique reduces the original reward function of an experience (e.g., a single or a sequence of state-action pairs) based on its novelty with respect to the molecules in memory. The memory consists of promising experiences with at least a reward of h. This could be seen as a negated intrinsic reward, but because it can remove the original reward function from the learning process, it also modifies the task. This provides a dynamic environment in which the agent must adapt to a new reward function while the end task is to find diverse solutions to the original problem. This notion of a non-stationary environment is captured by the definition of continual reinforcement learning by Abel et al. [Abe+23]. They think of learning as endless adaptation, where continual reinforcement learning refers to a setting in which the best agents never stop learning. In large solution spaces, as in drug design, endless adaptation is needed to explore the whole space.

Chapter 4

Research Challenges and Questions

Machine learning (ML) is widely deployed at every stage of the drug discovery process [Pit+25], but several challenges remain to fully leverage ML in drug discovery [Bla+23]. This thesis particularly concerns the use of ML in drug design. It addresses the challenge of combining ML's predictive power with algorithms for sequential decision-making to efficiently update ML models. Solving this challenge could enable autonomous optimization of the drug design process and minimize the time and resources required to develop new drugs. Based on this challenge, we seek answers to the following question: *How can different algorithms for sequential decision-making be used to enhance the deployment of ML in the DMTA cycle?* Given this broad challenge and question, this thesis focuses on five research challenges that motivate our work, identified through gaps in the literature. Furthermore, we formulate eight research questions based on these challenges.

4.1 Reseach Challenges

This thesis seeks to address the following research challenges (Cs), which are motivated by gaps in the literature:

C1 Unclear performance gain in using active learning for reaction yield prediction.

For reaction yield prediction, active learning still struggles to show a significant performance gain compared to randomly selecting data instances to query, so-called random sampling, when only a few data instances have been labeled [EGJ20]. Previous work on active learning investigates how its performance depends on model-centric and data-centric aspects such as the initial size of the labeled data and the neural network's capacity [BSC21]. However, this may be task-dependent, and no prior work has examined the performance of active learning for reaction yield

prediction across different problem settings, including both model-centric and data-centric aspects of the active learning problem.

C2 No existing approach considers what to make next based on suggestions developed by *de novo* drug design.

There has been a significant focus on *de novo* drug design to optimize a fixed objective function in the Design step of the DMTA cycle [MFB21]. However, to the best of our knowledge, no prior work has considered the sequential decision of which *de novo* generated molecules to make next in the DMTA cycle when the design objective is iteratively updated.

C3 There is no systematic comparison of how the sample efficiency of reinforcement learning-based *de novo* drug design is affected by iteratively learning from subsets of current and/or previous samples.

For *de novo* drug design using reinforcement learning, previous work proposes approaches based on hill climbing algorithms that learn from the best previous samples [Tho+22a; Nei+18; Bro+19]. However, to the best of our knowledge, no prior work in this domain systematically examines approaches for leveraging current and previous samples.

C4 There is no systematic evaluation on how modification of the environment's original reward function in reinforcement learning-based *de novo* drug design affects the chemical exploration.

In *de novo* drug design, it is desired to avoid generating similar molecules, i.e., mode collapse, but instead continually explore the vast chemical space to find diverse solutions. To prevent mode collapse, previous work either reduces the environment's original reward for molecules similar to previously generated ones [Bla+20] or uses intrinsic rewards by providing a bonus for novel molecules [Par+25]. However, to the best of our knowledge, no prior work has combined these techniques or systematically explored how these techniques affect chemical exploration.

C5 No existing reinforcement learning approach induces exploration via mini-batch diversification.

Many real-world applications, including *de novo* drug design, rely on costly, time-consuming human evaluations and computer experiments to assess data instances, where proper evaluation is essential in RL to obtain rewards for updating the policy's weights. In addition, ensuring sufficient exploration of the state-action space is a key challenge in RL and is important for finding diverse solutions in RL-based *de novo* drug design. Thus, it is crucial to select an informative mini-batch for policy updates to enable efficient exploration while limiting the number of evaluations. To the best of our knowledge, no prior work has studied diverse mini-batch selection in on-policy reinforcement learning and its effects on exploration when fine-tuning generative models.

4.2 Research Questions

Motivated by the above research challenges, this thesis considers the following research questions (RQs):

RQ1 *How is the performance of active learning for reaction yield prediction affected by the problem setting?*

To tackle challenge (C1), we propose studying how the performance of active learning, compared to passive learning (e.g., random sampling), varies across different settings of the active learning problem, including the initial size of the labeled data, machine learning algorithm and reaction dataset.

RQ2 *How many data instances are needed to be queried by active learning, compared to passive learning, when the objective is to reach a certain level of predictive ability on a reaction yield prediction task?*

In real-world applications, such as reaction yield prediction, a predictive model should have a sufficiently "good" predictive ability on a validation/test set to be usable. Therefore, to tackle challenge (C1), we also propose investigating how much training data is needed to achieve a given level of predictive performance using active learning, compared to passive learning (e.g., random sampling).

RQ3 *Can the problem of what to make next in the DMTA cycle be formulated as a multi-armed bandit problem?*

To tackle challenge (C2), we propose a solution by formulating the problem as a multi-armed bandit problem, as introduced in Section 3.2, and study different approaches.

RQ4 *Can the contextual Zooming algorithm be extended to provide a solution to the problem of what to make next in the DMTA cycle? How should it select arms covered by the same ball?*

To tackle challenge (C2), we propose to extend the contextual Zooming algorithm [Sli11] to the formulated multi-armed bandit problem of what to make next in the DMTA cycle. Also, we consider how the extended version should distinguish between arms covered by the same balls to improve sample efficiency and novelty.

RQ5 *How can string-based de novo drug design utilizing reinforcement learning improve its sample efficiency by learning from subsets of previous and current samples? How does it compare with using all samples in the current round for learning?*

To tackle challenge (C3), we propose to systematically investigate different approaches for learning from subsets of previous and current samples. This should obviously be compared with learning from the complete set of samples in the current round.

RQ6 *For string-based de novo drug design, how does the reinforcement learning algorithm affect the sample efficiency when learning from subsets of previous and current samples?*

To tackle challenge (C3), we propose to systematically study different reinforcement learning algorithms when learning from subsets of previous and current samples.

RQ7 *For string-based de novo drug design, how does reducing the environment's original reward and/or providing bonus reward affect the reinforcement learning algorithm's chemical exploration?*

To tackle challenge (C4), we propose to systematically study two ways to modify the environment's original reward: reducing the original reward for similar molecules (reward penalty) and providing a bonus reward for novel molecules. We also propose combining these approaches to enhance chemical exploration. We systematically study their effects on the chemical exploration in string-based *de novo* drug design when there is a maximum number of queries for the (original) reward.

RQ8 *For string-based de novo drug design, how does mini-batch diversification affect the reinforcement learning algorithm's chemical exploration?*

To tackle challenge (C5), we propose a framework for diverse mini-batch selection in on-policy reinforcement learning. We study how this approach affects the chemical exploration in *de novo* drug design over the generative process.

Chapter 5

Summary of Included Papers

In this chapter, the five papers included in this thesis are summarized, along with their research contributions. All papers concern sequential decision-making for drug design. Paper 1 examines the application of active learning to enhance reaction yield prediction. Paper 2 formulates what to make next in the DMTA cycle as a multi-armed bandit problem and proposes an algorithm to solve it. Paper 3 systematically investigates different deep reinforcement learning algorithms and replay buffers for SMILES-based *de novo* drug design. Paper 4 presents a framework to enhance diversity in RL-based *de novo* drug design. Paper 5 introduces mini-batch diversification in on-policy reinforcement learning to enhance the chemical exploration in *de novo* drug design.

5.1 Paper 1: Using Active Learning to Develop Machine Learning Models for Reaction Yield Prediction

In Paper 1, we investigate the use of active learning to iteratively improve machine learning models for predicting whether a reaction will output a sufficiently high yield, i.e., binary reaction yield prediction. When a reaction produces a sufficiently high yield, we denote the reaction as successful; otherwise, we denote it as unsuccessful. Given an initial set of labeled reaction data, we iteratively query the label of an unlabeled data instance and subsequently retrain the model, including the newly acquired label. The objective is to investigate, under different settings, how the predictive performance is affected by using active learning for label querying, and the relative change in the amount of training data needed to achieve different levels of predictive ability.

For predicting whether a reaction will be successful, we compare a neural network with a single hidden layer, a neural network with three hidden layers, a Bayesian matrix factorization model, and a random forest model. The

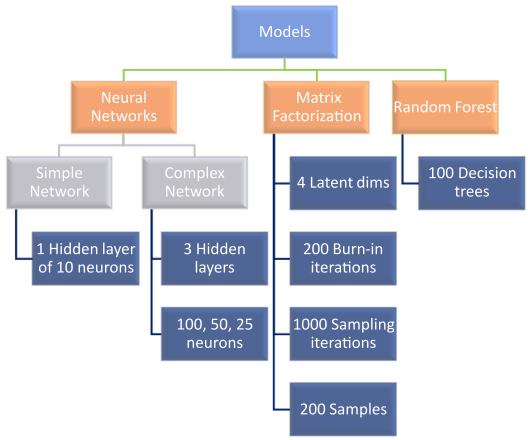


Figure 5.1: Configurations of the investigated machine learning models.

configurations of these machine learning models are illustrated in Figure 5.1. For these models, we evaluate active learning using a well-known uncertainty sampling approach based on the output margin, namely the *Margin* query strategy. This approach selects a new training instance to label by following the acquisition function

$$x^* = \underset{x}{\operatorname{argmin}} \left[P_\theta(\hat{y}_1|x) - P_\theta(\hat{y}_2|x) \right], \tag{5.1}$$

where $P_\theta(y|x)$ is the model's (with parameters θ) estimated probability that the true label of data instance x is label y, and $\hat{y}_1$ and $\hat{y}_2$ are the most and second most probable labels, respectively, according to the model. Note that in the problem considered in this paper, there are only two labels: "successful" and "unsuccessful". To study the effectiveness of active learning for this problem, we compare it with *random sampling* (i.e., passive learning), in which the label to query is chosen uniformly at random. In addition to how active learning depends on the model, we are interested in how other aspects affect its performance. Therefore, this paper investigates active learning on two fully combinatorial data sets with two different reaction types and different reaction variables, namely the Buchwald-Hartwig dataset by Ahneman et al. [Ahn+18] and the Suzuki reaction dataset by Perera et al. [Per+18]. We also investigate how the size of the initial labeled data pool affects predictive performance. This is done to study whether random bias in the initial data can affect the active learning process, where we investigate initial data with 10, 100, or 1000 labeled instances.

This paper presents a retrospective study in which all true labels (i.e., whether a reaction is successful) are known a priori, but, in our setting, they remain unknown until they are queried. One-hot encodings are used to examine how predictive ability is affected by active learning when only combinatorial patterns in the data are learned. The paper measures predictive ability using the area under the receiver operating characteristic curve (AUROC) and

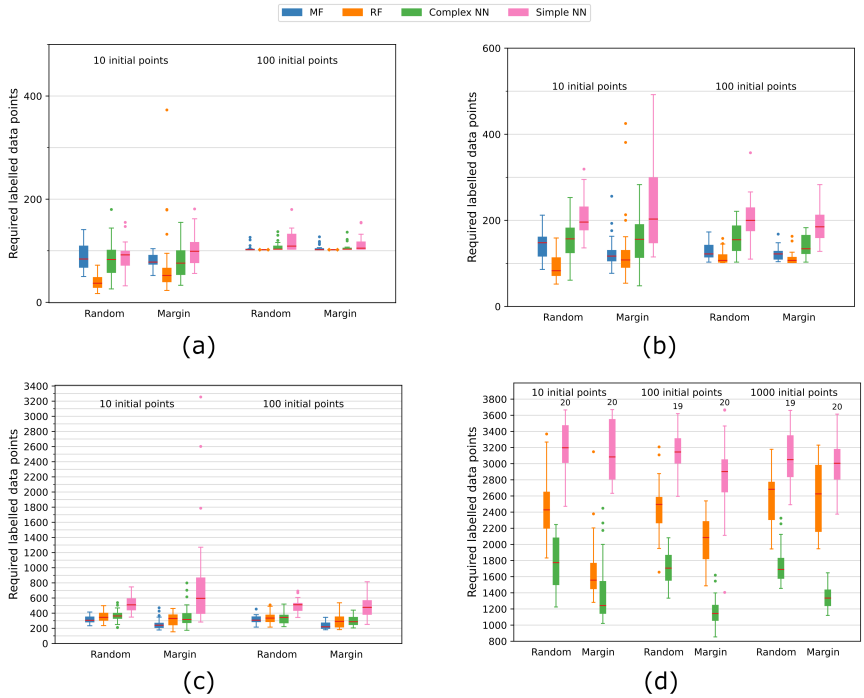


Figure 5.2: Boxplots showing the number of required labeled data instances of the Suzuki reaction dataset to achieve an AUROC of (a) 0.800, (b) 0.850, (c) 0.900 and (d) 0.950. When using an initial size of 1000, all models reached an AUROC of 0.800, 0.850 and 0.900 by using only the initial labels and, therefore, these settings are not displayed. No setting reached an AUROC of 0.975.

investigates how many unlabeled instances are needed to achieve a given AUROC. Figure 5.2 shows the number of labeled data instances needed to obtain an AUROC of 0.800, 0.850, 0.900 and 0.950 on the Suzuki reaction dataset. The results indicate that the greater predictive ability we require of the models, the larger the gain from active learning. We also conduct a feature importance analysis, as shown in Figure 5.3, to investigate differences in the effectiveness of active learning. We observe that several features seem to play an essential role in the prediction outcome. Our findings suggest a relationship between how well the machine learning models have learned from the labeled reaction data and the magnitude of the positive impact of active learning on predictive performance.

Contributions Hampus Gummesson Svensson and Simon Viet Johansson equally performed the main work, and the work was jointly supervised by Morteza Haghir Chehreghani, Ola Engkvist, Esben Jannik Bjerrum, Alexander Schliep and Christian Tyrchan.

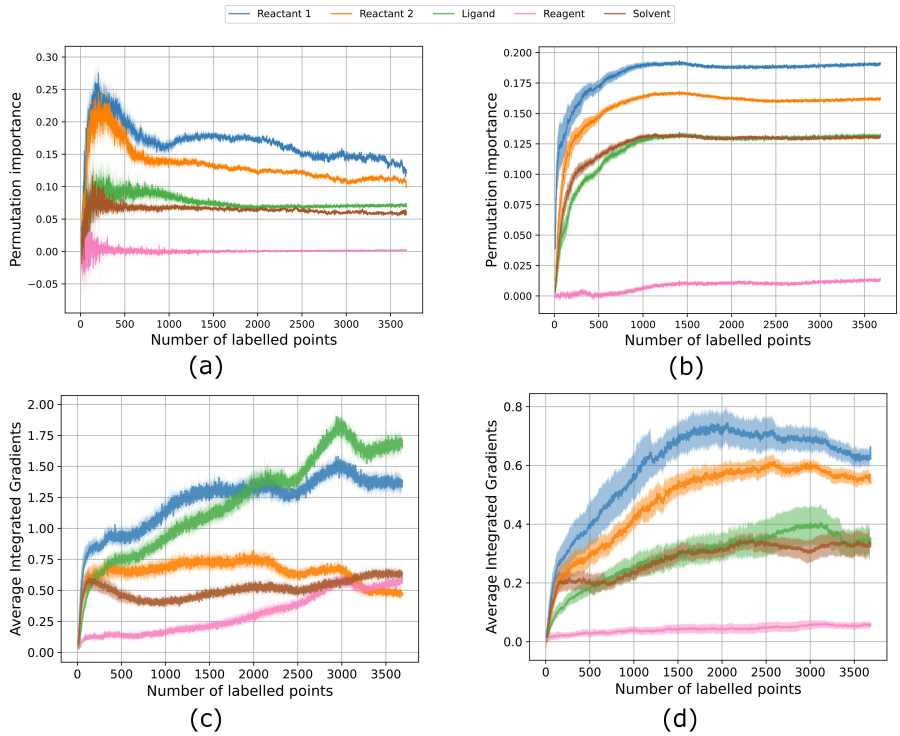


Figure 5.3: Permutation importance of (a) matrix factorization and (b) random forest, and Average Integrated Gradients of (c) complex and (d) simple neural network on the Suzuki reaction data. These were computed for the setting of 10 initial labels and using Margin query function. All show the average over all 25 runs and the corresponding 95% approximate confidence interval.

5.2 Paper 2: Autonomous Drug Design with Multi-Armed Bandits

Paper 1, summarized above, seeks to address the problem of how to synthesize a molecule to experimentally test its properties, where an unknown function provides these ground-truth properties during the Test step of the DMTA cycle. Thus, the DMTA cycle can be seen as sequential optimization of this unknown function. During the design step, the goal is to optimize this unknown function based on the current belief about the ground truth function, incorporating the current belief into the scoring function to guide the design. Thus, before the make step, it must be decided which molecules are most informative for updating the current belief, which, in this paper, is achieved by updating the parameters of a QSAR model that defines the scoring function. Nowadays, the generative models in the design step can produce a vast amount of molecules with promising properties, but experimental data is both costly and time-consuming to acquire, and therefore it is not possible to make, test and analyze

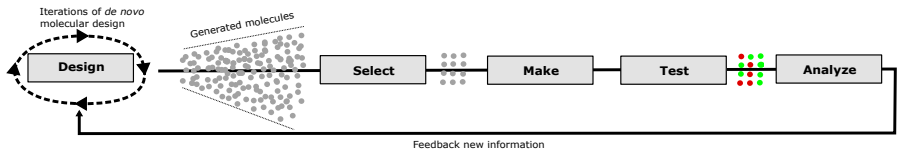


Figure 5.4: A schematic illustration of the *Select* step incorporated into the original DMTA cycle.

all of the generated molecules. Therefore, in Paper 2, we propose adding a step between the design and make steps, focusing on deciding which molecules to make. This is illustrated in Figure 5.4 by incorporating a *Select* step into the original DMTA cycle. Naïvely, one could acquire experimental data for the top-scoring molecules, but this is not necessarily the best approach. Instead, we formulate this as a stochastic multi-armed bandit problem to handle the inherent exploration-exploitation trade-off.

Algorithm 6 MAB formulation of the problem.

Input: Dissimilarity space $(\mathcal{X}, \mathcal{D}_{\mathcal{X}})$
1: **for** each round $t = 1, \ldots, T$ **do**
2: $M^t > K$ base arms, indexed by the set $\mathcal{M}^t$, arrive with corresponding feature vectors $\mathcal{X}^t \subset \mathcal{X}$ and scores $\mathcal{F}^t \in [0,1]^{M^t}$
3: Choose super arm $\mathcal{S}^t \subset \mathcal{M}^t$ of K distinct base arms
4: Observe reward $r(x_m^t) \in \{0,1\}$, $\forall m \in \mathcal{S}^t$
5: **end for**

To formulate this as a stochastic multi-armed bandit (MAB) problem, we consider a setting with multiple-plays, volatile arms and (dis-)similarity information, as illustrated in Algorithm 6. The multiple-play setting is appropriate since the algorithm should select several molecules to make, test and analyze in parallel before designing new molecules. A MAB problem with volatile arms is considered because an entirely new set of molecules can be generated in every design step, due to the randomness in the generation and the iterative update of the scoring function. Moreover, similarity information is available to assess the structural differences between the molecules.

To solve this MAB problem, we propose a Zooming algorithm with multiple plays and volatile arms, extending the contextual Zooming algorithm of Slivkins [Sli11] to our problem. The algorithm partitions the dissimilarity space of each base arm's feature vectors into balls of different radii, with the initial partition consisting of a ball covering the entire dissimilarity space. Given observed rewards for base arms covered by a ball, the empirical mean reward and corresponding confidence radius are computed. If the confidence radius of the empirical mean reward is less than or equal to the radius of the ball, the partition is refined by creating a new ball with half the radius. The radius of the ball is fixed, while the confidence radius is updated when rewards for base arms covered by the ball are observed. A set of available base arms and corresponding feature vectors is observed at the beginning of each round, and

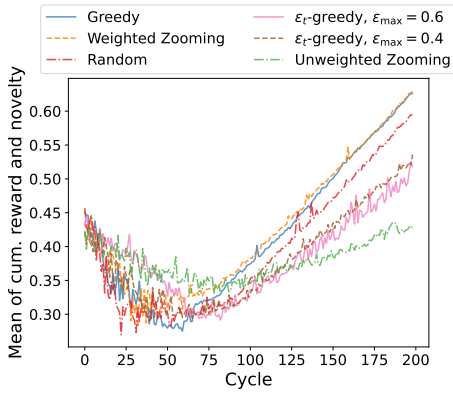


Figure 5.5: Average of cumulative reward and novelty.

subsequently, a super arm of multiple available base arms is chosen. Each base arm is chosen based on its index, which is computed from the empirical mean reward, radius, and confidence radius of the ball covering the accompanying feature vector.

For a fixed budget of molecules to be selected, we evaluate the proposed algorithm by comparing it with random selection, greedy selection (selecting the top-scoring molecules with respect to the current scoring function), and a combination of the two (ϵ-greedy selection). We use a dissimilarity space consisting of Morgan fingerprints, with dissimilarity measured by the Jaccard distance. For the proposed bandit algorithm, to investigate the effect of distinguishing arms covered by the same ball, we use a weighted index that accounts for the current score of the corresponding molecule. The results in Figure 5.5 show the mean of the cumulative reward and novelty per cycle. In this paper, we measure the novelty of the selected molecules in each cycle by their average pair-wise distance to active molecules selected in previous cycles. We find that the unweighted variant of our proposed algorithm performs among the best in the early cycles, while the weighted variant performs better in the later cycles. This suggests that using the benefits of both the unweighted and weighted variants can yield the best overall performance.

Contributions Hampus Gummesson Svensson performed the main work, and Morteza Haghir Chehreghani, Ola Engkvist, Esben Jannik Bjerrum, and Christian Tyrchan jointly supervised the work.

5.3 Paper 3: Utilizing Reinforcement Learning for Drug Design

The selection of what to make next in Paper 2 depends on the molecular design in the Design step of the DMTA cycle. Ideally, a structurally diverse set of

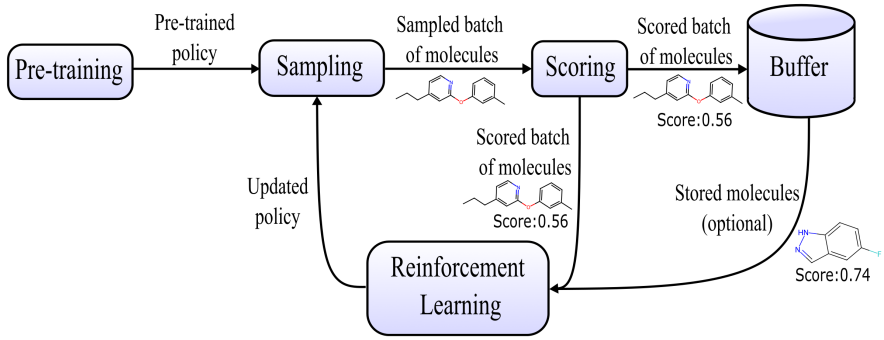


Figure 5.6: Schematic illustration of the reinforcement learning-based *de novo* drug design process with a replay buffer.

promising molecules should be generated to enable sufficient exploration and exploitation in selecting what to make next. This should be achieved with sample efficiency in mind, as practical constraints limit the number of molecules the scoring function can evaluate. Previous work has shown promising results using reinforcement learning for molecular *de novo* design, compared to other approaches such as variational autoencoders [Gao+22; Tho+22b]. Moreover, several works have proposed combining reinforcement learning with a hill climbing algorithm, which learns from the k top-scoring sequences [Tho+22a; Nei+18; Bro+19].

In Paper 3, we investigate various reinforcement learning algorithms for iteratively generating batches of molecules via a SMILES-based generative model. The policy optimization reinforcement learning algorithms that we investigate in the paper are Proximal Policy Optimization (PPO) [Sch+17], Advantage Actor-Critic (A2C) [Mni+16b], Soft Actor-Critic (SAC) [Haa+18], Actor-Critic with Experience Replay (ACER) [Wan+16], and Regularized Maximum Likelihood Estimation (MLE) [Oli+17]. SAC and ACER are off-policy algorithms, whereas the others are on-policy. The aim is to iteratively update a policy pre-trained on the ChEMBL dataset [Zdr+24] that provides probabilities for the next character to append to a SMILES string. The next character (action) is sampled from a multinomial distribution where the probabilities are given by the current policy and the SMILES string is finalized when the stop token is chosen as the next character. When a batch of finalized SMILES strings has been generated, these are scored to provide a reward signal to the RL algorithm. This *de novo* drug design process is illustrated in Figure 5.6. In this paper, we define an episode as the process of generating a batch of molecules.

Besides exploring different popular RL algorithms, we compare seven different ways to learn from molecules generated in the current iteration and previous iterations: i) learn from the batch of molecules generated in the current iteration (AC); ii) learn from the batch of current molecules and previously generated molecules with diverse rewards (BH); iii) learn from a subset of the current batch with diverse rewards (BC); iv) learn from the current batch,

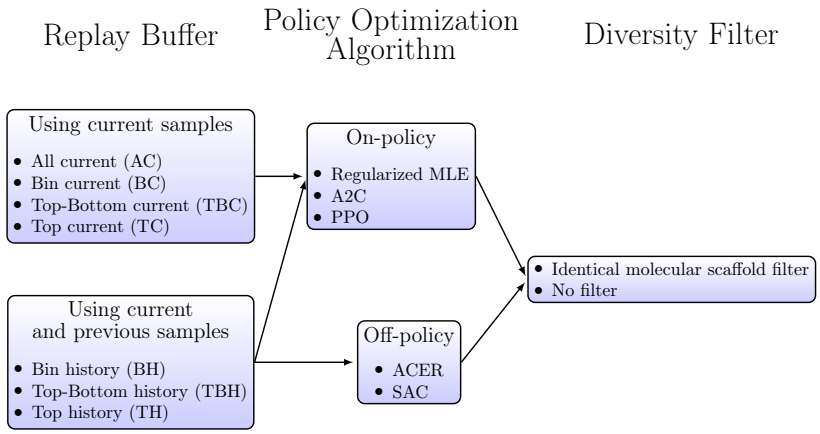
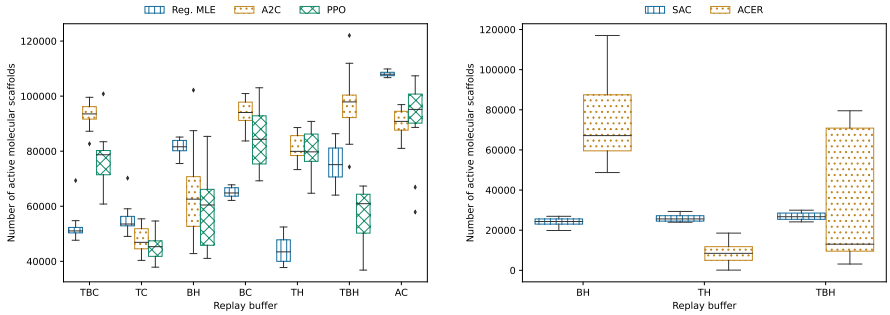


Figure 5.7: Illustration of the different combinations of replay buffer, policy optimization algorithm and diversity filter investigated in Paper 3.

and high- and low-rewarding molecules from previous iterations (TBH); v) learn from the highest and lowest rewarding molecules of the current batch (TBC); vi) learn from the current batch and high-rewarding molecules from previous batches (TH); vii) learn from the highest rewarding molecules of the current batch (TC). We collect all these approaches under the term *replay buffer*, given their ability to store and provide both current and previously generated molecules. These approaches are inspired by previous work that suggests combining reinforcement learning with a hill climbing algorithm, which learns from the k top-scoring sequences. Since the off-policy algorithms SAC and ACER already include an on-policy update step that uses the whole current batch, they were compared using only the BH, TBH, or TH replay buffers.

To evaluate the quality and diversity of the generated molecules, for a predefined budget of generated molecules, we compare the average episodic reward, number of active molecules and the corresponding number of unique scaffolds. We investigate different combinations of policy optimization algorithms and replay buffers for generating molecules predicted to be active against the dopamine receptor D2 (DRD2). This is evaluated both with and without a diversity filter, which penalizes the generation of molecules with similar scaffolds across iterations, as introduced by [Bla+20]. The various combinations of the replay buffer, policy gradient algorithm and diversity filter are illustrated in Figure 5.7. As part of our evaluation, Figure 5.8 shows the number of active molecular scaffolds for the different combinations. We find that using at least both high- and low-rewarding molecules is advantageous for generating a large number of active compounds with diverse scaffolds. Also, we observed that using off-policy algorithms with multiple off-policy updates does not necessarily yield more active molecules or scaffolds.



(a) On-policy algorithms.

(b) Off-policy algorithms.

Figure 5.8: Box plots of the number of active molecular scaffolds for the on-policy and off-policy algorithms (higher is better) when utilizing diversity filter.

Contributions Hampus Gummesson Svensson performed the main work, and the work was jointly supervised by Morteza Haghir Chehreghani, Ola Engkvist, and Christian Tyrchan.

5.4 Paper 4: Diversity-Aware Reinforcement Learning for *de novo* Drug Design

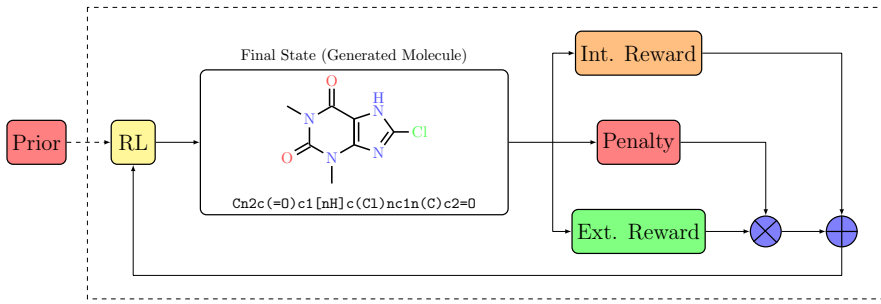
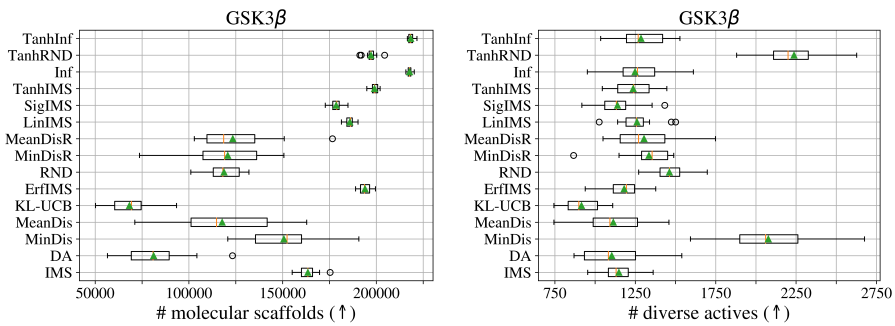


Figure 5.9: The proposed diversity-aware RL framework for *de novo* drug design utilizing extrinsic reward penalty and intrinsic reward to improve the diversity.

In Paper 4, we continue our study of RL-based *de novo* drug design, aiming to fine-tune a pre-trained generative model. We focus on the chemical exploration conducted during the fine-tuning process to identify a diverse set of promising molecules. Especially, we aim to improve the search for novel, active molecular structures across the vast chemical space. In previous work in *de novo* drug design, a popular approach to address this problem is to retain a memory of previously generated molecular scaffolds and let the agent observe a reduced reward signal for them [Bla+20]. We refer to this as (extrinsic)

reward penalization, in which a predefined function reduces the environment's original reward, known as the *extrinsic reward*, during the learning process, thereby discouraging the agent from generating these structures. The reward signal is often reduced by a binary function that transitions from 1 to 0 when a certain number of molecules with the same molecular scaffold have been generated. Alternatively, in previous work on RL, a common strategy to enhance exploration is to provide the agent with an intrinsic reward to yield novel behaviours.

In the paper, we propose a framework that combines an extrinsic reward penalty and an intrinsic reward to enhance exploration. We introduce and study new application-specific methods that, to the best of our knowledge, have not been previously explored. We also argue that reducing the reward by using a monotonic function is more beneficial than using a step function. The RL agent generates molecules by following its policy, e.g., in the SMILES representation used in this paper, and subsequently uses the penalty and/or intrinsic reward to modify the extrinsic reward. Each extrinsic reward is multiplied by the corresponding penalty term (equal to one if no penalty is used), while the intrinsic reward (equal to zero if no intrinsic reward is used) is added to the product. The modified rewards are observed by the RL agent and used to update its policy. This diversity-aware RL framework is illustrated in Figure 5.9. Intrinsic motivation encourages the agent to find new solutions without considering the extrinsic reward, while the reward penalty determines the importance of the extrinsic reward. We claim that combining these techniques enhances the chemical exploration in RL-based *de novo* drug design.



(a) Molecular Scaffolds. (b) Diverse Actives.

Figure 5.10: Evaluation of diversity on the GSK3β-based reward function. The boxplots compare the total number of molecular scaffolds and the total number of diverse actives after 2000 generative steps. Each boxplot shows the evaluation across 20 independent runs of each method, with the orange line and green triangle displaying the median and mean, respectively. Only active molecules are considered when computing the diversity metrics.

To support this claim, we evaluate our proposed framework on two well-established QSAR models targeting the GSK3β enzyme and JNK3 protein,

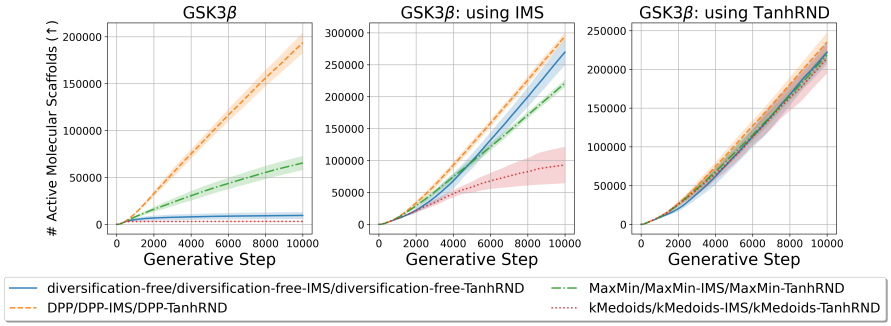
respectively, and focus on how it can enhance the structural diversity of the generated active molecules. To study the diversity of the active molecules, we utilize two reference-based and one distance-based diversity metric for the active molecules, namely the number of molecular scaffolds, topological scaffolds and diverse actives. For the different strategies explored in this paper, Figure 5.10 displays the number of molecular scaffolds and diverse actives on the GSK3β-based reward function. We consistently observe that "TanhRND", which combines a non-binary penalty function with a prediction-based intrinsic reward, yields solutions with the highest chemical diversity. This strategy also reaches similar rewards as the other investigated methods. Hence, we can maintain high solution quality while increasing diversity within the solution set.

Contributions Hampus Gummesson Svensson performed the main work, and the work was jointly supervised by Morteza Haghir Chehreghani, Ola Engkvist, and Christian Tyrchan.

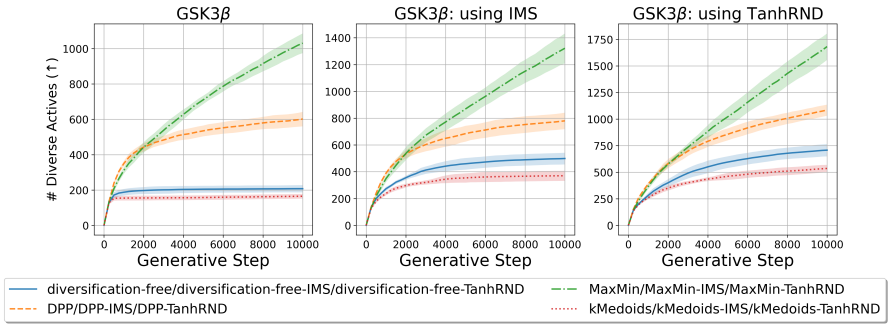
5.5 Paper 5: Diverse Mini-Batch Selection in Reinforcement Learning for Efficient Chemical Exploration in *de novo* Drug Design

In Paper 5, we continue to study chemical exploration in RL-based *de novo* drug design. We provide a novel perspective on exploration in on-policy RL, namely, mini-batch diversification for efficient exploration. We argue that selecting a diverse mini-batch for policy updates enhances exploration. The updates focus on learning from diverse and promising instances (i.e., interactions with the environment), thereby improving both quality and diversity without spending resources evaluating similar instances. This is particularly important in many real-world applications where it is costly and time-consuming to evaluate an instance to obtain a reward for reinforcement learning, e.g., in *de novo* drug design. Then, there is a limited number of evaluations that can be performed and, therefore, effective exploration for updating the policy is essential.

Based on this perspective, we propose a framework for diverse mini-batch selection in on-policy reinforcement learning. While we seek to minimize the number of costly, time-consuming evaluations to obtain rewards, we assume that a large set of interactions with the environment (without rewards) can be readily obtained by the current policy. Thus, we first follow the agent's policy to generate B instances, yielding a large set in which the expected return is optimized. Only considering quality when obtaining this set can be critical in real-world applications where safety and validity are at stake. For instance, the validity of SMILES representations (and many other molecular representations) is sensitive to noise. Given B high-quality instances, we propose selecting k diverse instances to summarize the larger set, assuming $B \gg k$ so that selecting a diverse set is meaningful. We argue that this



(a) Molecular Scaffold.



(b) Diverse Actives.

Figure 5.11: Total number of molecular scaffolds and diverse activities, after each generative step, evaluated on GSK3β-based reward function. The total number of diverse actives is plotted for every 250th generative step. We investigate four different approaches: 1) **diversification-free**: no mini-batch diversification; 2) **DPP**: mini-batch diversification via Determinantal Point Processes (DPPs); 3) **MaxMin**: mini-batch diversification via the MaxMin algorithm; 4) **kMedoids**: mini-batch diversification via k-medoids clustering.

makes policy updates more effective by considering only high-quality, diverse instances while accounting for the evaluation budget. We explore the use of determinantal point processes [KT12], the MaxMin algorithm [Ash+02] and k-medoids clustering [RK87] to select a diverse batch. Given a kernel matrix (i.e., a similarity function), determinantal point processes (DPPs) are known for their *repulsive* behaviour, where similar instances are less likely to occur together. Specifically, k-DPPs sample exactly k diverse instances from a larger set, enabling mini-batch selection when there is a budget on the total number of evaluations. The MaxMin algorithm iteratively selects the most dissimilar instance to the already chosen points until k points have been selected, while k-medoids clustering seeks to find k clusters with medoids that minimize the average dissimilarity to all the other items in the cluster.

We evaluate the proposed framework on RL-based *de novo* drug design, where live deployment often requires costly, time-consuming evaluations of

generated molecules to obtain corresponding rewards from a scoring function. We examine reward functions based on three well-established QSAR models targeting the DRD2, GSK3β, and JNK3 proteins/enzymes. The mini-batch selection depends on the kernel used to measure the similarity between instances and, therefore, it is important to incorporate relevant information for the task at hand. We study two different kernels and their additive and multiplicative combinations. One kernel computes the Tanimoto similarity between Morgan fingerprints, while the other computes the Dice similarity between atom-pair fingerprints of the molecular scaffolds. To evaluate the effects on chemical exploration, we assess the diversity of generated molecules by computing both molecular scaffolds and diverse actives. Figure 5.11 displayes the evaluation on the GSK3β-based reward function. To assess the chemical exploration over time, we report the number of molecular scaffolds and diverse actives over 10 000 generative steps. Each generative step corresponds to updating the policy on $k = 64$ evaluate molecules, selected from $B = 640$ generated molecules. We explore whether the proposed framework can be combined with the framework in Paper 4, where we utilize the binary extrinsic reward penalty (IMS) proposed by Blaschke et al. [Bla+20] and TanhRND proposed in Paper 4. To study the effectiveness of mini-batch diversification, we also run experiments where the policy directly generates k instances (*diversification-free*). We observe that mini-batch diversification via k-DPP sampling can effectively increase the number of diverse actives while delivering a similar or larger number of scaffolds. Thus, we find that DPP-based mini-batch diversification provides the best overall chemical exploration.

Contributions Hampus Gummesson Svensson performed the main work, and the work was jointly supervised by Morteza Haghir Chehreghani, Ola Engkvist, Jon Paul Janet, and Christian Tyrchan.

Chapter 6

Concluding Remarks and Future Directions

In this thesis, we investigate three decision-making problems in computer-aided drug design, focusing on both the design and make steps of the Design-Make-Test-Analyze cycle. This includes active data selection for training a reaction yield prediction model, deciding on what to make next, and efficient training and chemical exploration in *de novo* drug design. In particular, we study these problems in the context of three decision-making problems in machine learning: active learning, multi-armed bandit and reinforcement learning problems. This thesis demonstrates that techniques for solving these problems can support informed decision-making and presents novel approaches to drug design.

The main research result of Paper 1 is that increasing the training data by utilizing active learning can enhance the predictive ability of reaction yield prediction, compared to training on the same amount of random data. We study this in the fundamental setting of using one-hot encodings and increasing the training data by one instance at a time. There may be a larger gain from using active learning in settings with more elaborate feature vectors per reaction and when a batch of instances is added to the training data at once. However, this problem is more complex and is likely to require more elaborate active learning methods, e.g., by accounting for batch diversity and quantifying feature vector uncertainty. In Paper 2, the main research outcomes include formulating the problem of which molecules to make and test next as a multi-armed bandit problem. In addition, an algorithm for solving this problem is proposed. The proposed algorithm shows promising performance in balancing exploration and exploitation in the formulated problem. In Paper 3, the main research outcomes include the systematic study of on- and off-policy policy optimization algorithms for SMILES-based *de novo* drug design. This also includes comparing different ways to learn from both current and previously generated samples. The paper concludes that both high- and low-rewarding molecules should be used to update the policy, to improve the quality and diversity of the output. In Paper 4, the main research outcome is a proposed framework that incorporates an extrinsic reward penalty and an intrinsic reward to enhance diversity in *de*

novo drug design when using reinforcement learning to fine-tune a generative model. This also includes a comparison of different extrinsic reward penalties and intrinsic rewards, evaluated on the structural diversity, which concludes that a combination of these techniques is advantageous. In Paper 5, the main outcome is a framework for mini-batch diversification in on-policy reinforcement learning to enable efficient exploration, especially when rewards are expensive to obtain. Moreover, our extensive experiments show that this framework can significantly enhance chemical exploration in *de novo* drug design, leading to distinct output behaviours during fine-tuning of a generative model.

The work in this thesis demonstrates the potential of machine learning approaches to address impactful, real-world decision-making problems in drug design. However, this thesis studies these problems in simplified settings to enable controlled, reproducible comparisons. Therefore, real-world deployment and assessment are required to study the robustness of the approaches suggested in this work.

6.1 Future Directions

This thesis studies some of the complex decisions required to enhance computer-aided drug design, potentially enabling autonomous drug design. Still, greater focus is needed on robust decision-making throughout the entire drug design process. This also includes further studies of the machine learning approaches used in this thesis, which are needed; hence, we suggest possible future directions below.

To further investigate the advantages of active learning (AL) in drug design, conducting prospective studies to predict reaction yields is essential for assessing real-world effects, i.e., performing experiments in real time rather than relying on a fixed dataset. This opens the possibility of a larger set of unlabelled data, potentially leading to greater performance gains from AL than from random sampling. Also, the effects of using more complex features should be studied to determine how they affect the model's uncertainty and generalization. When using more complex features, it can be beneficial to employ deep learning models, which are already common in drug design. For instance, in Paper 1, we observe a performance gain when combining active learning with neural networks. However, the one-by-one sample query strategy in traditional AL is inefficient and not applicable to AL for deep learning models [SS18; Zhd19]. Batch-based AL query strategies in drug design should further investigate how to balance selecting instances from high-uncertainty and promising areas, e.g., as in [Kha+22], based on disagreement among an ensemble of models. Extending the framework proposed in Paper 5 can help find a balance between exploration and exploitation.

Advances in various aspects of reinforcement learning can open new directions for its application in drug design. For instance, previous work in *de novo* drug design has focused on sampling efficiency, for which reinforcement learning has demonstrated promising performance [Gao+22; Tho+22b]. Hence, it would be interesting to further investigate how to improve the sample efficiency in

deep reinforcement learning for *de novo* drug design. For this purpose, different sampling strategies could be studied. This work considers only multinomial sampling without temperature; future work can compare temperature sampling and other sampling strategies, e.g., explicit diversity-driven sampling that does not require DPPs to promote diversity. Moreover, a future direction for investigating sample efficiency is to analyze the convergence rate to the optimal policy of current deep learning-based algorithms. For instance, the neural tangent kernel provides a technique for analyzing convergence in neural networks [JGH18]. To improve sample efficiency, and because experimental data is both costly and time-consuming to obtain, a future direction could be to investigate further offline reinforcement learning, which learns from stored data [Lev+20]. Moreover, the existing literature on continual reinforcement learning [Abe+23] is limited. Thus, it could be impactful to study this topic, as many real-world problems can be formulated as open-ended problems whose problems and tasks can change over time.

Another interesting problem is the design of the scoring function (i.e., the objective function), which comprises several scoring components. Inverse reinforcement learning considers the problem of learning an agent's objectives, e.g., by observing a human expert [AD21]. Also, instead of having a single agent trying to maximize a reward function composed of several components, one could investigate using multiple agents, each optimizing a specific reward component while globally maximizing the cumulative reward by sharing knowledge.

To conclude, this thesis concerns learning to make decisions from experiences of interaction with an unknown environment. Learning to make decisions from experience is still waiting for its "big break" and for widespread adoption to solve real-world problems. However, as argued by Silver and Sutton [SS25], to continue advancing artificial intelligence, agents need to learn from experience through autonomous interaction with environments. This is a first step towards this in drug design. Still, algorithmic advancements are needed, and it is crucial to consider real-world problems to guide these advancements in a valuable direction.

References

[Abe+23] David Abel et al. 'A definition of continual reinforcement learning'. In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 50377–50407 (cit. on pp. 27, 49).

[ABL03] Naoki Abe, Alan W Biermann and Philip M Long. 'Reinforcement learning with immediate rewards and linear hypotheses'. In: *Algorithmica* 37.4 (2003), pp. 263–293 (cit. on p. 20).

[ÅCC20] Niklas Åkerblom, Yuxin Chen and Morteza Haghir Chehreghani. 'An Online Learning Framework for Energy-Efficient Navigation of Electric Vehicles'. In: *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI*. Ed. by Christian Bessiere. 2020, pp. 2051–2057 (cit. on p. 19).

[ACF02] Peter Auer, Nicolo Cesa-Bianchi and Paul Fischer. 'Finite-time analysis of the multiarmed bandit problem'. In: *Machine learning* 47.2 (2002), pp. 235–256 (cit. on pp. 22, 23).

[AD21] Saurabh Arora and Prashant Doshi. 'A survey of inverse reinforcement learning: Challenges, methods and progress'. In: *Artificial Intelligence* 297 (2021), p. 103500 (cit. on p. 49).

[AG13] Shipra Agrawal and Navin Goyal. 'Thompson sampling for contextual bandits with linear payoffs'. In: *International conference on machine learning*. PMLR. 2013, pp. 127–135 (cit. on p. 20).

[ÅHC22] Niklas Åkerblom, Fazeleh Sadat Hoseini and Morteza Haghir Chehreghani. 'Online Learning of Network Bottlenecks via Minimax Paths'. In: *Mach. Learn.* (2022). DOI: 10.1007/s10994-022-06270-0 (cit. on p. 19).

[Ahn+18] Derek T Ahneman et al. 'Predicting reaction performance in C–N cross-coupling using machine learning'. In: *Science* 360.6385 (2018), pp. 186–190 (cit. on p. 34).

[AHT+90] R Agrawal, M Hegde, D Teneketzis et al. 'Multi-armed bandit problems with multiple plays and switching cost'. In: *Stochastics and Stochastic reports* 29.4 (1990), pp. 437–459 (cit. on p. 21).

[AMH19] Arthur Aubret, Laetitia Matignon and Salima Hassas. 'A survey on intrinsic motivation in reinforcement learning'. In: *arXiv preprint arXiv:1908.06976* (2019) (cit. on p. 27).

[APW24] Amira Alakhdar, Barnabas Poczos and Newell Washburn. 'Diffusion models in de novo drug design'. In: *Journal of Chemical Information and Modeling* 64.19 (2024), pp. 7238–7256 (cit. on p. 15).

[Aru+17] Kai Arulkumaran et al. 'Deep reinforcement learning: A brief survey'. In: *IEEE Signal Processing Magazine* 34.6 (2017), pp. 26–38 (cit. on p. 26).

[Arú+19] Josep Arús-Pous et al. 'Randomized SMILES strings improve the quality of molecular generative models'. In: *ChemRxiv* (2019). DOI: 10.26434/chemrxiv.8639942.v2 (cit. on p. 8).

[Ash+02] Mark Ashton et al. 'Identification of diverse database subsets using property-based and fragment-based molecular descriptions'. In: *Quantitative Structure-Activity Relationships* 21.6 (2002), pp. 598–604 (cit. on p. 44).

[Ata+22] Sara Romeo Atance et al. 'De Novo Drug Design Using Reinforcement Learning with Graph-Based Deep Generative Models'. In: *Journal of Chemical Information and Modeling* 62.20 (2022), pp. 4863–4872. DOI: 10.1021/acs.jcim.2c00838 (cit. on p. 15).

[ATB17] Thomas Anthony, Zheng Tian and David Barber. *Thinking Fast and Slow with Deep Learning and Tree Search*. 2017. arXiv: 1705.08439 [cs.AI] (cit. on p. 25).

[Aue02] Peter Auer. 'Using confidence bounds for exploitation-exploration trade-offs'. In: *Journal of Machine Learning Research* 3.Nov (2002), pp. 397–422 (cit. on p. 20).

[AVW87] Venkatachalam Anantharam, Pravin Varaiya and Jean Walrand. 'Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays-Part I: IID rewards'. In: *IEEE Transactions on Automatic Control* 32.11 (1987), pp. 968–976 (cit. on p. 21).

[Bel+16] Marc Bellemare et al. 'Unifying count-based exploration and intrinsic motivation'. In: *Advances in neural information processing systems*. NIPS'16 29 (2016). Ed. by Daniel D. Lee et al., pp. 1471–1479 (cit. on p. 27).

[BG04] Andreas Bender and Robert C Glen. 'Molecular similarity: a key technique in molecular informatics'. In: *Organic & biomolecular chemistry* 2.22 (2004), pp. 3204–3218 (cit. on p. 13).

[Bic+12] G Richard Bickerton et al. 'Quantifying the chemical beauty of drugs'. In: *Nature chemistry* 4.2 (2012), pp. 90–98 (cit. on p. 15).

[Bil+22] Camille Bilodeau et al. 'Generative models for molecular discovery: Recent advances and challenges'. In: *Wiley Interdisciplinary Reviews: Computational Molecular Science* 12.5 (2022), e1608 (cit. on pp. 4, 5).

[Bla+20] Thomas Blaschke et al. 'Memory-assisted reinforcement learning for diverse molecular de novo design'. In: *Journal of cheminformatics* 12.1 (2020), p. 68 (cit. on pp. 27, 30, 40, 41, 45).

[Bla+23] Alexandre Blanco-Gonzalez et al. 'The role of AI in drug discovery: challenges, opportunities, and strategies'. In: *Pharmaceuticals* 16.6 (2023), p. 891 (cit. on p. 29).

[BM96] Guy W Bemis and Mark A Murcko. 'The properties of known drugs. 1. Molecular frameworks'. In: *Journal of medicinal chemistry* 39.15 (1996), pp. 2887–2893 (cit. on p. 12).

[Bna+13] Zahy Bnaya et al. 'Volatile Multi-Armed Bandits for Guaranteed Targeted Social Crawling.' In: *AAAI (Late-Breaking Developments)* 2.2.3 (2013), pp. 16–21 (cit. on p. 22).

[BRA20] Djallel Bouneffouf, Irina Rish and Charu Aggarwal. 'Survey on applications of multi-armed and contextual bandits'. In: *2020 IEEE Congress on Evolutionary Computation (CEC)*. IEEE. 2020, pp. 1–8 (cit. on p. 21).

[Bre01] Leo Breiman. 'Random forests'. In: *Machine learning* 45.1 (2001), pp. 5–32 (cit. on p. 13).

[BRH15] Dávid Bajusz, Anita Rácz and Károly Héberger. 'Why is Tanimoto index an appropriate choice for fingerprint-based similarity calculations?' In: *Journal of cheminformatics* 7.1 (2015), p. 20 (cit. on p. 11).

[Bro+19] Nathan Brown et al. 'GuacaMol: benchmarking models for de novo molecular design'. In: *Journal of chemical information and modeling* 59.3 (2019), pp. 1096–1108 (cit. on pp. 15, 30, 39).

[BSC21] John Daniel Bossér, Erik Sörstadius and Morteza Haghir Chehreghani. 'Model-centric and data-centric aspects of active learning for deep neural networks'. In: *2021 IEEE International Conference on Big Data (IEEE Big Data 2021)*. IEEE. 2021, pp. 5053–5062 (cit. on p. 29).

[Bur+19] Yuri Burda et al. 'Exploration by random network distillation'. In: *International Conference on Learning Representations*. 2019 (cit. on p. 27).

[CGJ18] Connor W Coley, William H Green and Klavs F Jensen. 'Machine learning in computer-aided synthesis planning'. In: *Accounts of chemical research* 51.5 (2018), pp. 1281–1289 (cit. on p. 6).

[Che+18] Hongming Chen et al. 'The rise of deep learning in drug discovery'. In: *Drug discovery today* 23.6 (2018), pp. 1241–1250 (cit. on p. 4).

[Chu+11] Wei Chu et al. 'Contextual bandits with linear payoff functions'. In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings. 2011, pp. 208–214 (cit. on p. 20).

[Chu+18] Kurtland Chua et al. *Deep Reinforcement Learning in a Handful of Trials using Probabilistic Dynamics Models*. 2018. arXiv: `1805.12114 [cs.LG]` (cit. on p. 25).

[CSV85] Raymond E Carhart, Dennis H Smith and RENGACHARI Venkataraghavan. 'Atom pairs as molecular features in structure-activity studies: definition and applications'. In: *Journal of Chemical Information and Computer Sciences* 25.2 (1985), pp. 64–73 (cit. on p. 10).

[Dav+15] Mark Davies et al. 'ChEMBL web services: streamlining access to drug discovery data and utilities'. In: *Nucleic acids research* 43.W1 (2015), W612–W620 (cit. on p. 15).

[Dav+20] Laurianne David et al. 'Molecular representations in AI-driven drug discovery: a review and practical guide'. In: *Journal of Cheminformatics* 12.1 (2020), pp. 1–22 (cit. on pp. 8, 9).

[Dic45] Lee R Dice. 'Measures of the amount of ecologic association between species'. In: *Ecology* 26.3 (1945), pp. 297–302 (cit. on p. 10).

[DK16] Danishuddin and Asad U. Khan. 'Descriptors and their selection methods in QSAR analysis: paradigm for drug design'. In: *Drug Discovery Today* 21.8 (2016), pp. 1291–1302. ISSN: 1359-6446. DOI: 10.1016/j.drudis.2016.06.013 (cit. on p. 13).

[EGJ20] Natalie S Eyke, William H Green and Klavs F Jensen. 'Iterative experimental design based on active machine learning reduces the experimental burden associated with reaction screening'. In: *Reaction Chemistry & Engineering* 5.10 (2020), pp. 1963–1972 (cit. on p. 29).

[ES09] Peter Ertl and Ansgar Schuffenhauer. 'Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions'. In: *Journal of cheminformatics* 1.1 (2009), p. 8 (cit. on p. 15).

[Gao+22] Wenhao Gao et al. 'Sample Efficiency Matters: A Benchmark for Practical Molecular Optimization'. In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo et al. Vol. 35. Curran Associates, Inc., 2022, pp. 21342–21357 (cit. on pp. 39, 48).

[Góm+18] Rafael Gómez-Bombarelli et al. 'Automatic chemical design using a data-driven continuous representation of molecules'. In: *ACS central science* 4.2 (2018), pp. 268–276 (cit. on p. 15).

[Gre21] Daria Grechishnikova. 'Transformer neural network for protein-specific de novo drug generation as a machine translation problem'. In: *Scientific reports* 11.1 (2021), p. 321 (cit. on p. 15).

[Gum23] Hampus Gummesson Svensson. 'Sequential Decision-Making for Drug Design: Towards Closed-Loop Drug Design'. PhD thesis. Chalmers University of Technology and University of Gothenburg, 2023 (cit. on p. ii).

[Haa+18] Tuomas Haarnoja et al. 'Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor'. In: *International conference on machine learning*. PMLR. 2018, pp. 1861–1870 (cit. on pp. 26, 39).

[Hao+23] Jianye Hao et al. 'Exploration in deep reinforcement learning: From single-agent to multiagent domain'. In: *IEEE Transactions on Neural Networks and Learning Systems* 35.7 (2023), pp. 8762–8782 (cit. on pp. 26, 27).

[Hei+23] Esther Heid et al. 'Chemprop: a machine learning package for chemical property prediction'. In: *Journal of Chemical Information and Modeling* 64.1 (2023), pp. 9–17 (cit. on p. 14).

[Ho95] Tin Kam Ho. 'Random decision forests'. In: *Proceedings of 3rd international conference on document analysis and recognition*. Vol. 1. IEEE. 1995, pp. 278–282 (cit. on p. 13).

[HSB16] Ye Hu, Dagmar Stumpfe and Jürgen Bajorath. 'Computational exploration of molecular scaffolds in medicinal chemistry: Miniperspective'. In: *Journal of medicinal chemistry* 59.9 (2016), pp. 4062–4076. DOI: 10.1021/acs.jmedchem.6b01437 (cit. on p. 12).

[HSW89] Kurt Hornik, Maxwell Stinchcombe and Halbert White. 'Multilayer feedforward networks are universal approximators'. In: *Neural networks* 2.5 (1989), pp. 359–366 (cit. on p. 26).

[Hu+24] Xiuyuan Hu et al. 'Hamiltonian diversity: effectively measuring molecular diversity by shortest Hamiltonian circuits'. In: *Journal of Cheminformatics* 16.1 (2024), p. 94 (cit. on p. 11).

[Hug+11] James P Hughes et al. 'Principles of early drug discovery'. In: *British journal of pharmacology* 162.6 (2011), pp. 1239–1249 (cit. on p. 3).

[Irw+20] John J Irwin et al. 'ZINC20—a free ultralarge-scale chemical database for ligand discovery'. In: *Journal of chemical information and modeling* 60.12 (2020), pp. 6065–6073 (cit. on p. 15).

[IS05] John J Irwin and Brian K Shoichet. 'ZINC- a free database of commercially available compounds for virtual screening'. In: *Journal of chemical information and modeling* 45.1 (2005), pp. 177–182 (cit. on p. 15).

[Jac01] Paul Jaccard. 'Étude comparative de la distribution florale dans une portion des Alpes et des Jura'. In: *Bull Soc Vaudoise Sci Nat* 37 (1901), pp. 547–579 (cit. on p. 10).

[JBJ18] Wengong Jin, Regina Barzilay and Tommi Jaakkola. 'Junction tree variational autoencoder for molecular graph generation'. In: *International conference on machine learning*. PMLR. 2018, pp. 2323–2332 (cit. on p. 15).

[Jen19] Jan H Jensen. 'A graph-based genetic algorithm and generative model/Monte Carlo tree search for the exploration of chemical space'. In: *Chemical science* 10.12 (2019), pp. 3567–3572 (cit. on p. 15).

[JGH18] Arthur Jacot, Franck Gabriel and Clement Hongler. 'Neural Tangent Kernel: Convergence and Generalization in Neural Networks'. In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio et al. Vol. 31. Curran Associates, Inc., 2018, pp. 8571–8580 (cit. on p. 49).

[Joh+19] Simon Johansson et al. 'AI-assisted synthesis prediction'. In: *Drug Discovery Today: Technologies* 32 (2019), pp. 65–72 (cit. on p. 16).

[Kai+20] Lukasz Kaiser et al. *Model-Based Reinforcement Learning for Atari*. 2020. arXiv: 1903.00374 [cs.LG] (cit. on p. 25).

[Kea+16] Steven Kearnes et al. 'Molecular graph convolutions: moving beyond fingerprints'. In: *Journal of computer-aided molecular design* 30.8 (2016), pp. 595–608 (cit. on p. 14).

[Kha+22] Yuriy Khalak et al. 'Chemical space exploration with active learning and alchemical free energies'. In: *Journal of Chemical Theory and Computation* 18.10 (2022), pp. 6259–6270 (cit. on p. 48).

[KHN15] Junpei Komiyama, Junya Honda and Hiroshi Nakagawa. 'Optimal Regret Analysis of Thompson Sampling in Stochastic Multi-armed Bandit Problem with Multiple Plays'. In: *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by Francis Bach and David Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, July 2015, pp. 1152–1161 (cit. on p. 21).

[Kim+25] Sunghwan Kim et al. 'PubChem 2025 update'. In: *Nucleic acids research* 53.D1 (2025), pp. D1516–D1525 (cit. on p. 15).

[KNS10] Robert Kleinberg, Alexandru Niculescu-Mizil and Yogeshwer Sharma. 'Regret bounds for sleeping experts and bandits'. In: *Machine learning* 80.2 (2010), pp. 245–272 (cit. on p. 22).

[KSU08] Robert Kleinberg, Aleksandrs Slivkins and Eli Upfal. 'Multi-Armed Bandits in Metric Spaces'. In: *arXiv preprint arXiv:0809.4882* (2008). DOI: 10.48550/arXiv.0809.4882 (cit. on p. 22).

[KSU19] Robert Kleinberg, Aleksandrs Slivkins and Eli Upfal. 'Bandits and experts in metric spaces'. In: *Journal of the ACM (JACM)* 66.4 (2019), pp. 1–77 (cit. on p. 22).

[KT12] Alex Kulesza and Ben Taskar. 'Determinantal point processes for machine learning'. In: *Foundations and Trends® in Machine Learning* 5.2–3 (2012), pp. 123–286 (cit. on p. 44).

[Lan06] Greg Landrum. *RDKit: Open-source cheminformatics*. 2006. URL: http://www.rdkit.org (cit. on pp. 9, 11).

[Lev+20] Sergey Levine et al. 'Offline reinforcement learning: Tutorial, review, and perspectives on open problems'. In: *arXiv preprint arXiv:2005.01643* (2020) (cit. on p. 49).

[Li+10a] Lihong Li et al. 'A contextual-bandit approach to personalized news article recommendation'. In: *Proceedings of the 19th International Conference on World Wide Web, WWW*. ACM, 2010, pp. 661–670 (cit. on p. 19).

[Li+10b] Lihong Li et al. 'A contextual-bandit approach to personalized news article recommendation'. In: *Proceedings of the 19th international conference on World wide web*. 2010, pp. 661–670 (cit. on p. 20).

[Lil+15] Timothy P Lillicrap et al. 'Continuous control with deep reinforcement learning'. In: *arXiv preprint arXiv:1509.02971* (2015) (cit. on p. 26).

[Lio+14] Evanthia Lionta et al. 'Structure-based virtual screening for drug discovery: principles, applications and recent advances'. In: *Current topics in medicinal chemistry* 14.16 (2014), pp. 1923–1938 (cit. on p. 14).

[LR+85] Tze Leung Lai, Herbert Robbins et al. 'Asymptotically efficient adaptive allocation rules'. In: *Advances in applied mathematics* 6.1 (1985), pp. 4–22 (cit. on p. 18).

[LS20] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020 (cit. on p. 20).

[Mac+11] Ricardo Macarron et al. 'Impact of high-throughput screening in biomedical research'. In: *Nature reviews Drug discovery* 10.3 (2011), pp. 188–195 (cit. on p. 3).

[Mar09] Yvonne Connolly Martin. 'Let's not forget tautomers'. In: *Journal of computer-aided molecular design* 23 (2009), pp. 693–704 (cit. on p. 7).

[Mas+14] Vijay H Masand et al. 'Does tautomerism influence the outcome of QSAR modeling?' In: *Medicinal Chemistry Research* 23 (2014), pp. 1742–1757 (cit. on p. 7).

[McI07] Campbell McInnes. 'Virtual screening strategies in drug discovery'. In: *Current opinion in chemical biology* 11.5 (2007), pp. 494–502 (cit. on p. 14).

[MCT17] Andrea Mauri, Viviana Consonni and Roberto Todeschini. 'Molecular descriptors'. In: *Handbook of computational chemistry*. Springer, 2017, pp. 2065–2093 (cit. on pp. 9, 13).

[Men+19] Steven M Mennen et al. 'The evolution of high-throughput experimentation in pharmaceutical development and perspectives on the future'. In: *Organic Process Research & Development* 23.6 (2019), pp. 1213–1242 (cit. on p. 16).

[Mer+21] Rocío Mercado et al. 'Graph networks for molecular design'. In: *Machine Learning: Science and Technology* 2.2 (2021), p. 025023 (cit. on p. 15).

[MFB21] Joshua Meyers, Benedek Fabian and Nathan Brown. 'De novo molecular design and generative models'. In: *Drug Discovery Today* 26.11 (2021), pp. 2707–2715 (cit. on pp. 14, 30).

[Mni+13] Volodymyr Mnih et al. 'Playing atari with deep reinforcement learning'. In: *arXiv preprint arXiv:1312.5602* (2013) (cit. on p. 25).

[Mni+16a] Volodymyr Mnih et al. 'Asynchronous Methods for Deep Reinforcement Learning'. In: *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learn-

ing Research. New York, New York, USA: PMLR, June 2016, pp. 1928–1937 (cit. on p. 26).

[Mni+16b] Volodymyr Mnih et al. 'Asynchronous methods for deep reinforcement learning'. In: *International conference on machine learning*. PmLR. 2016, pp. 1928–1937 (cit. on p. 39).

[Mor65] Harry L Morgan. 'The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service.' In: *Journal of chemical documentation* 5.2 (1965), pp. 107–113 (cit. on p. 9).

[Mos+20] Henry Moss et al. 'Boss: Bayesian optimization over string spaces'. In: *Advances in neural information processing systems* 33 (2020), pp. 15476–15486 (cit. on p. 15).

[Mur+20] Eugene N Muratov et al. 'QSAR without borders'. In: *Chemical Society Reviews* 49.11 (2020), pp. 3525–3564 (cit. on p. 13).

[Nei+18] Daniel Neil et al. *Exploring deep recurrent models with reinforcement learning for molecule design*. In: 6th International Conference on Learning Representations. 2018 (cit. on pp. 30, 39).

[Nev+18] Bruno J Neves et al. 'QSAR-based virtual screening: advances and applications in drug discovery'. In: *Frontiers in pharmacology* 9 (2018), p. 1275 (cit. on p. 13).

[Nil+24] Hannes Nilsson et al. 'Tree Ensembles for Contextual Bandits'. In: *Transactions on Machine Learning Research* (2024). ISSN: 2835-8856 (cit. on p. 21).

[Oli+17] Marcus Olivecrona et al. 'Molecular de-novo design through deep reinforcement learning'. In: *Journal of cheminformatics* 9.1 (2017), pp. 1–14 (cit. on pp. 26, 39).

[Pan+23] Chao Pang et al. 'Deep generative models in de novo drug molecule generation'. In: *Journal of Chemical Information and Modeling* 64.7 (2023), pp. 2174–2194 (cit. on pp. 14, 15).

[Par+25] Jinyeong Park et al. 'Mol-AIR: Molecular Reinforcement Learning with Adaptive Intrinsic Rewards for Goal-Directed Molecular Generation'. In: *Journal of Chemical Information and Modeling* 65.5 (2025), pp. 2283–2296 (cit. on p. 30).

[Pau+10] Steven M Paul et al. 'How to improve R&D productivity: the pharmaceutical industry's grand challenge'. In: *Nature reviews Drug discovery* 9.3 (2010), pp. 203–214 (cit. on p. 3).

[Per+18] Damith Perera et al. 'A platform for automated nanomole-scale reaction screening and micromole-scale synthesis in flow'. In: *Science* 359.6374 (2018), pp. 429–434 (cit. on p. 34).

[Pit+25] Will R. Pitt et al. 'Real-World Applications and Experiences of AI/ML Deployment for Drug Discovery'. In: *Journal of Medicinal Chemistry* (2025). DOI: 10.1021/acs.jmedchem.4c03044 (cit. on p. 29).

[Plo+12] Alleyn T Plowright et al. 'Hypothesis driven drug design: improving quality and effectiveness of the design-make-test-analyse cycle'. In: *Drug discovery today* 17.1-2 (2012), pp. 56–62 (cit. on p. 4).

[PMV13] Pavel G Polishchuk, Timur I Madzhidov and Alexandre Varnek. 'Estimation of the size of drug-like chemical space based on GDB-17 data'. In: *Journal of computer-aided molecular design* 27 (2013), pp. 675–679 (cit. on p. 4).

[Pre09] William H. Press. 'Bandit solutions provide unified ethical models for randomized clinical trials and comparative effectiveness research'. In: *Proc. Natl. Acad. Sci. USA* 106.52 (2009), pp. 22387–22392 (cit. on p. 19).

[RA12] Jean-Louis Reymond and Mahendra Awale. 'Exploring chemical space for drug discovery using the chemical universe database'. In: *ACS chemical neuroscience* 3.9 (2012), pp. 649–657 (cit. on p. 4).

[Ren+19] Philipp Renz et al. 'On failure modes in molecule generation and optimization'. In: *Drug Discovery Today: Technologies* 32 (2019), pp. 55–63 (cit. on p. 5).

[RH10] David Rogers and Mathew Hahn. 'Extended-connectivity fingerprints'. In: *Journal of chemical information and modeling* 50.5 (2010), pp. 742–754 (cit. on p. 9).

[RK87] LKPJ Rdusseeun and P Kaufman. 'Clustering by means of medoids'. In: *Proceedings of the statistical data analysis based on the L1 norm conference, neuchatel, switzerland*. Vol. 31. 1987, p. 28 (cit. on p. 44).

[RLK24] Philipp Renz, Sohvi Luukkonen and Günter Klambauer. 'Diverse hits in de novo molecule design: Diversity-based comparison of goal-directed generators'. In: *Journal of Chemical Information and Modeling* 64.15 (2024), pp. 5756–5761 (cit. on p. 13).

[Rob52] Herbert Robbins. 'Some aspects of the sequential design of experiments'. In: *Bulletin of the American Mathematical Society* 58.5 (1952), pp. 527–535 (cit. on p. 18).

[Sai+19] Semion K Saikin et al. 'Closed-loop discovery platform integration is needed for artificial intelligence to make an impact in drug discovery'. In: *Expert opinion on drug discovery* 14.1 (2019), pp. 1–4 (cit. on p. 5).

[SB18] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018 (cit. on pp. 5, 23).

[Sch+15] Tom Schaul et al. 'Prioritized experience replay'. In: *arXiv preprint arXiv:1511.05952* (2015) (cit. on p. 26).

[Sch+17] John Schulman et al. 'Proximal policy optimization algorithms'. In: *arXiv preprint arXiv:1707.06347* (2017) (cit. on pp. 26, 39).

[Sch+19] Philippe Schwaller et al. 'Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction'. In: *ACS central science* 5.9 (2019), pp. 1572–1583 (cit. on p. 16).

[Sch+21] Philippe Schwaller et al. 'Prediction of chemical reaction yields using deep learning'. In: *Machine learning: science and technology* 2.1 (2021), p. 015016 (cit. on p. 16).

[Sch18] Gisbert Schneider. 'Automating drug discovery'. In: *Nature reviews drug discovery* 17.2 (2018), pp. 97–113 (cit. on p. 4).

[Sch91] Jürgen Schmidhuber. 'A possibility for implementing curiosity and boredom in model-building neural controllers'. In: *Proc. of the international conference on simulation of adaptive behavior: From animals to animats*. 1991, pp. 222–227 (cit. on p. 27).

[Seg+18] Marwin HS Segler et al. 'Generating focused molecule libraries for drug discovery with recurrent neural networks'. In: *ACS central science* 4.1 (2018), pp. 120–131 (cit. on p. 15).

[Set09] Burr Settles. 'Active learning literature survey'. In: (2009) (cit. on p. 6).

[Set12] Burr Settles. *Active learning*. Synthesis lectures on artificial intelligence and machine learning 18. Morgan & Clay Pool, 2012. ISBN: 9781608457267. DOI: 10.2200/S00429ED1V01Y201207AIM018 (cit. on p. 17).

[She+15] Weiwei Shen et al. 'Portfolio Choices with Orthogonal Bandit Learning'. In: *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI*. Ed. by Qiang Yang and Michael J. Wooldridge. 2015, p. 974 (cit. on p. 19).

[Sil+17] David Silver et al. *Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm*. 2017. arXiv: 1712.01815 [cs.AI] (cit. on pp. 25, 26).

[Sli11] Aleksandrs Slivkins. 'Contextual Bandits with Similarity Information'. In: *Proceedings of the 24th Annual Conference on Learning Theory*. Ed. by Sham M. Kakade and Ulrike von Luxburg. Vol. 19. Proceedings of Machine Learning Research. Budapest, Hungary: PMLR, June 2011, pp. 679–702 (cit. on pp. 23, 31, 37).

[Sli24] Aleksandrs Slivkins. 'Introduction to Multi-Armed Bandits'. In: *arXiv* (2024). DOI: 10.48550/arXiv.1904.07272. arXiv: 1904.07272 [cs.LG] (cit. on pp. 6, 20, 22, 23).

[Sör48] Thorvald Sörensen. 'A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons'. In: *Biologiske skrifter* 5 (1948), pp. 1–34 (cit. on p. 10).

[SS18] Ozan Sener and Silvio Savarese. 'Active Learning for Convolutional Neural Networks: A Core-Set Approach'. In: *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. 2018 (cit. on p. 48).

[SS25] David Silver and Richard S Sutton. 'Welcome to the era of experience'. In: *Google AI* 1 (2025) (cit. on p. 49).

[SU07] Andrew I Schein and Lyle H Ungar. 'Active learning for logistic regression: an evaluation'. In: (2007) (cit. on p. 18).

[Sut+99] Richard S Sutton et al. 'Policy gradient methods for reinforcement learning with function approximation'. In: *Advances in neural information processing systems* 12 (1999) (cit. on p. 25).

[Sze10] Csaba Szepesvári. 'Algorithms for reinforcement learning'. In: *Synthesis lectures on artificial intelligence and machine learning* 9 (2010), Algorithms for reinforcement learning. DOI: 10.2200/S00268ED1V01Y201005AIM009 (cit. on p. 23).

[Tan58] Taffee T Tanimoto. *An elementary mathematical theory of classification and prediction*. International Business Machines Corporation, 1958 (cit. on p. 10).

[Tho+22a] Morgan Thomas et al. 'Augmented Hill-Climb increases reinforcement learning efficiency for language-based de novo molecule generation'. In: *Journal of Cheminformatics* 14.1 (2022), pp. 1–22 (cit. on pp. 30, 39).

[Tho+22b] Morgan Thomas et al. 'Re-evaluating sample efficiency in de novo molecule generation'. In: *arXiv preprint arXiv:2212.01385* (2022). arXiv: 2212.01385 [cs.CE] (cit. on pp. 39, 48).

[Tod+12] Roberto Todeschini et al. 'Similarity coefficients for binary chemoinformatics data: overview and extended comparison using simulated and real data sets'. In: *Journal of chemical information and modeling* 52.11 (2012), pp. 2884–2901 (cit. on p. 11).

[Tro10] Alexander Tropsha. 'Best practices for QSAR model development, validation, and exploitation'. In: *Molecular informatics* 29.6-7 (2010), pp. 476–488 (cit. on pp. 7, 14).

[Tyr+22] Christian Tyrchan et al. 'Chapter 4 - Approaches using AI in medicinal chemistry'. In: *Computational and Data-Driven Chemistry Using Artificial Intelligence*. Ed. by Takashiro Akitsu. Elsevier, 2022, pp. 111–159. ISBN: 978-0-12-822249-2. DOI: 10.1016/B978-0-12-822249-2.00002-5 (cit. on pp. 13, 14).

[Vam+19] Jessica Vamathevan et al. 'Applications of machine learning in drug discovery and development'. In: *Nature reviews Drug discovery* 18.6 (2019), pp. 463–477 (cit. on p. 4).

[VBW15] Sofía S Villar, Jack Bowden and James Wason. 'Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges'. In: *Statistical science: a review journal of the Institute of Mathematical Statistics* 30.2 (2015), p. 199 (cit. on p. 19).

[VS17] Divya Vohora and Gursharan Singh. *Pharmaceutical medicine and translational clinical research*. London, United Kingdom: Academic Press, 2017. ISBN: 978-0-12-802103-3 (cit. on p. 3).

[Wan+16] Ziyu Wang et al. 'Sample efficient actor-critic with experience replay'. In: *arXiv preprint arXiv:1611.01224* (2016) (cit. on pp. 26, 39).

[Wei88] David Weininger. 'SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules'. In: *Journal of chemical information and computer sciences* 28.1 (1988), pp. 31–36 (cit. on p. 8).

[Wil+17] Mary Jo Wildey et al. 'Chapter Five - High-Throughput Screening'. In: *Platform Technologies in Drug Discovery and Validation*. Ed. by Robert A. Goodnow. Vol. 50. Annual Reports in Medicinal Chemistry. Academic Press, 2017, pp. 149–195. DOI: 10.1016/bs.armc.2017.08.004 (cit. on p. 3).

[WML20] Olivier J Wouters, Martin McKee and Jeroen Luyten. 'Estimated research and development investment needed to bring a new medicine to market, 2009-2018'. In: *Jama* 323.9 (2020), pp. 844–853 (cit. on p. 3).

[Xie+23] Yutong Xie et al. 'How Much Space Has Been Explored? Measuring the Chemical Space Covered by Databases and Machine-Generated Molecules'. In: *The Eleventh International Conference on Learning Representations (ICLR)*. 2023 (cit. on pp. 12, 13).

[Yan+15] Yi Yang et al. 'Multi-class active learning by uncertainty sampling with diversity maximization'. In: *International Journal of Computer Vision* 113 (2015), pp. 113–127 (cit. on p. 18).

[Yos+18] Naruki Yoshikawa et al. 'Population-based de novo molecule generation, using grammatical evolution'. In: *Chemistry Letters* 47.11 (2018), pp. 1431–1434 (cit. on p. 15).

[You+08] Douglas Young et al. 'Are the chemical structures in your QSAR correct?' In: *QSAR & combinatorial science* 27.11-12 (2008), pp. 1337–1345 (cit. on p. 14).

[Zdr+24] Barbara Zdrazil et al. 'The ChEMBL Database in 2023: a drug discovery platform spanning multiple bioactivity data types and time periods'. In: *Nucleic acids research* 52.D1 (2024), pp. D1180–D1192 (cit. on pp. 7, 15, 39).

[Zhd19] Fedor Zhdanov. 'Diverse mini-batch active learning'. In: *arXiv preprint arXiv:1901.05954* (2019) (cit. on p. 48).

[ZY20] Hongming Zhang and Tianyang Yu. 'Taxonomy of reinforcement learning algorithms'. In: *Deep reinforcement learning: Fundamentals, research and applications*. Springer, 2020, pp. 125–133 (cit. on p. 25).