Università degli Studi di Palermo

AREA QUALITÀ, PROGRAMMAZIONE E SUPPORTO STRATEGICO
SETTORE STRATEGIA PER LA RICERCA
U. O. DOTTORATI

Dottorato in Ingegneria dell'innovazione tecnologica- Ciclo XXXIV
Dipartimento dell'Innovazione Industriale e Digitale
Laboratorio di Interazione Uomo-macchina

Study and identification of new molecular descriptors, finalized to the development of Virtual Screening techniques through the use of deep neural networks.

Ph.D. Candidate:
Salvatore Contino

Ph.D. Coordinator:
Prof. **Salvatore Gaglio**

Supervisor:
Prof. **Roberto Pirrone**

Co-Supervisor:
Dr. **Ugo Perricone**

CICLO XXXIV
ANNO CONSEGUIMENTO TITOLO 2022

Abstract

Technological advancement in the field of artificial intelligence has allowed the world to leverage new technologies to make improvements in the application world. In particular, one of the areas that has benefited the most is the field of Cheminformatics and Drug Discovery. Until recently, this field was based on a "trial-and-error" approach that has now been abandoned in favor of more accurate and, above all, less time consuming methods. In this dissertation the molecular descriptors that allow the conversion of chemical information into machine-readable data are discussed. Virtual Screening and Drug Repurposing are the domains of Drug Discovery within which the research activity of this thesis was conducted in order to evaluate the molecular descriptors, on classification tasks of bioactivity of small molecules on specific protein targets, through Deep Learning algorithms. Specifically, the CDKs family has been used as a protein target, for the fundamental role they play in cell cycle regulation. A first phase of theoretical study on the various cheminformatic representations (Molecular Graphs, Canonical SMILES, InChI, Fingerprints) allowed me to identify the Molecular Fingerprints, a vector representation of fixed length, as the most suitable descriptor for the research task. In fact, the Fingeprints encode structural information in a hashed bitmap proving to be the most efficient embedding computationally. Different families of Molecular Fingeprints (each one encoding structural information in a different way) have been tested to identify the best embedding size. Thereafter, each family of Molecular Fingeprints was examined for the impact of each molecular structure encoding on bioactivity classification performance. These experimental steps were fundamental to identify the strengths and especially the weaknesses of the descriptor, directing the topic of research on the creation of innovative and efficient molecular descriptors, EMBER and NMR-Like. EMBER (embedding multiple molecular fingerprints) comes from the observation that the different families of Fingerprints encode complementary information for the representation of the molecule, facilitating the task of classification. EMBER is presented as a 3D pseudo-image that contrary to the parallel use of Molecular Fingerprints is more efficient, being able to perform multi-class multi-target classification with a very low computational cost. The second innovative descriptor presented in this work is NMR-Like, which aims to overcome the limitations of Fingeprints while maintaining a computationally efficient numerical embedding. NMR-Like is based on H-NMR spectra of small molecules and is the first molecular descriptor used in the Virtual Screening domain. Contrary to Molecular Fingeprints, despite being a numerical embedding of fixed size, it manages to keep readable the molecular features and structure, allowing the operator to be able to interpret the distinctive features of each active molecule. This feature, added to the low computational cost required for the training of neural networks that use it as input data, makes it an ideal embedding. In conclusion, the work of the thesis shows an analysis of the descriptors present at the state of the art, focusing on Molecular Fingeprints that are used as a starting point to generate two innovative representations able to improve the efficiency in classification tasks (EMBER) and interpretability (NMR-Like).

Contents

List of Figures

List of Tables

Acronyms

ADMET	Absorption, Distribution, Metabolism, Excretion and Toxicity.
AI	Artificial Intelligence.
AKT1	AKT Serine/Threonine Kinase 1.
AUC	Area Under the Curve.
CDK	Cycline-Dependent Kinases.
CETSA	Cellular ThermoStability Assay.
CLAIRE	Confederation of Laboratories for Artificial Intelligence in Europe.
CNN	Convolutional Neural Network.
Ctab	Connection tables.
DD	Drug Discovery.
DL	Deep Learning.
DNN	Deep Neural Network.
DR	Drug Repurposing.
EHR	Eletronic Health Records.
EMA	European Medicines Agency.
FDA	Food and Drug Administration.
FID	Free Induction Decay.
GWAS	Genome-Wide Association Studies.
HHIP	Host High Identity Protein.
HTS	High-Throughput Screening.
IFP	Interaction Fingerprint Pattern.
LB	Ligand-Based.
LBVS	Ligand-Based Virtual Screening.
LIMK1	LIM Domani Kinase 1.
ML	Machine Learning.
NCBI	National Center of Biotechnology Information.

NMR	Nuclear Magnetic Resonance.
PDB	Protein Data Bank.
PDD	Phenotypic Drug Discovery.
QSAR	Quantitative Structure-Activity Relationship.
RF	Random Forest.
RIPK1	Receptor-Interacting Protein 1.
SB	Structure-Based.
SBVS	Structure-Based Virtual Screening.
SDF	Structure Data Format.
SK	Structural Keys.
SMILES	Simplified Molecular Input Line Entry System.
SVM	Support Vector Machine.
TDD	Target-based Drug Discovery.
TFs	Transcription Factors.
TRKA	Tropomysion receptor kinase A.
V-HTS	Virtual High-throughput Screening.
VHIP	Virus High Identity Protein.
VS	Virtual screening.

Chapter 1

Introduction

Various Artificial Intelligence (AI) concepts have been successfully used in recent years for Drug Discovery and chemical-pharmaceutical research more generally [7]. The expansion and increased accessibility of AI and the various related technologies have enabled these sectors of the scientific world to optimise many processes from the earliest to the most advanced stages of pharmaceutical research. Specifically, Machine Learning (ML) and Deep Learning (DL) algorithms are among the technologies that have played a key role in this technological progress. Specifically, these algorithms are based on the capacity of an AI system to acquire knowledge through the identification of patterns from raw data.

Deep Learning resolves this difficulty by breaking the desired complicated mapping into a series of nested simple mappings, each described by a different layer of the model. The input is presented at the visible layer, so named because it contains the variables that we are able to observe. Then a series of hidden layers extracts increasingly abstract features from the image. These layers are called "hidden" because their values are not given in the data; instead the model must determine which concepts are useful for explaining the relationships in the observed data. This characteristic of deep learning gives it great flexibility for different tasks, making it more suitable than machine learning techniques for tackling more complex, real-world problems. Drug Discovery (DD) is one such complex problem; it is the process of identifying a chemical entity that has the potential to become a new therapeutic agent for the treatment of one or more diseases. Traditional pharmaceutical research methodologies require a large investment in terms of both time and money; indeed, to develop a new drug, the average pre-tax expenditure is approximately US$2558 billion (for the year 2013) [8] and takes approximately 10-15 years [9, 10]. However, despite the huge investment of time and money, the success rate of clinical approval of small molecules is estimated at about 13%, implying a relatively high risk of failure. This rate of risk has prompted pharmaceutical companies to develop a new approach to research, beginning to integrate computer-based approaches such as those described above. This is why chemoinformatics was born, a term coined in 1998 by Frank Brown, an early founder of chemoinformatics, [11] who defined it as "mixing of information resources to transform data into information, and information into knowledge, for the intended purpose of making decisions faster in the arena of drug lead identification and optimisation" [12]. The latter has the objective of identifying a lead compound, decreasing production costs by halving the time to create a new drug using increasingly precise and efficient procedures, making the most of the great accessibility of data that the Big Data era makes more and more accessible.

1.1 Drug Discovery and Chemoinformatics

As a result of the many advances that have been made in the field of biotechnology, pharmacology has undergone a significant reshaping. There has been a shift from a trial-and-error approach (known as forward pharmacology) to more accurate methods, using the latest discoveries in molecular biology to discover new pharmaceutical molecules (known as reverse pharmacology) [13]. In Forward Pharmacology, also known as Phenotypic Drug Discovery (PDD), compounds are screened in cellular or animal disease models to identify molecules that have a beneficial effect: only after an active drug has been identified is an attempt made to identify the biological target of the drug [14]. In reverse pharmacology, also known as Target-based Drug Discovery (TDD), a biological target is hypothesised to be disease-modifying. A High-Throughput Screening (HTS) of compound libraries against the protein target is completed, identifying successful compounds that are then optimised and, unlike in reverse pharmacology, tested for in vivo efficacy in the final stages of drug discovery [15]. This new approach, in recent years has been fostered by increased knowledge of biological systems illuminated and new '*omics*' sciences, which have increased our ability to link diseases to their causes, leading to an exponential increase in drug targets. In light of new discoveries and approaches in pharmacology and molecular biology, biotechnology has become the driving force behind drug discovery and a major research area for new start-ups and companies worldwide [16].

The approaches to DD are therefore very dynamic, favouring the TDD approach in recent years, reducing lead identification time in the early stages of research. A schematic of the Drug Discovery process is shown in the figure below 1.1.

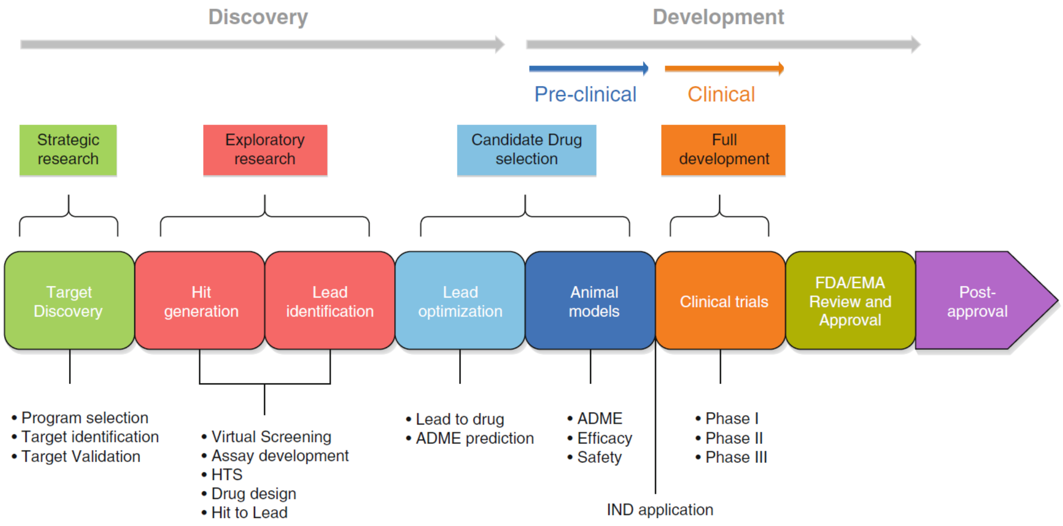


Figure 1.1: Schematic representation of the drug discovery and development process

As can be seen from the figure, Drug Discovery can be divided into two main phases: (i) discovery and (ii) development. The pharmaceutical development phases include the use of animal models and the first in vivo tests, to verify the information regarding the Absorption, Distribution, Metabolism, Excretion and Toxicity (ADMET) properties, each of which can influence the advancement to the next step of a drug candidate, should it not meet the necessary selection criteria in the in vivo models that present a set of variables that the computer model does not have. After the animal model phase, the drug progresses to clinical trials (divided into phase I, phase II and phase III) in which the drug candidate is tested on volunteers affected by the disease. If the drug is found to be suitable and succeeds in passing this scrupulous analysis, it will be further evaluated by a team of experts from the Food and Drug Administration (FDA) [17] and/or the European Medicines Agency (EMA) [18], and will eventually be marketed. As can be easily understood, this phase of pharmaceutical production cannot and must not be subject to reductions in the timescales that characterise it, so pharmaceutical companies in recent years have been trying to optimise the Discovery process, encouraging the use of new technologies provided by chemoinformatics, in an attempt to reduce costs and timescales.

The main applications of chemoinformatics in DD, include target selection, virtual library generation, Virtual High-throughput Screening (V-HTS), data mining, Quantitative Structure-Activity Relationship (QSAR) and in silico prediction of ADMET properties [12]. In particular, the phase that has gained the most benefit has been Virtual screening (VS), more precisely the Ligand-Based (LB) approach, i.e. screening based on the structural information of molecules whose biological activity is known on a specific target [19], in contrast to the Structure-Based (SB) approach, where reference is made to the 3D structure of the protein binding site obtained through X-Ray crystallography, Nuclear Magnetic Resonance (NMR) spectroscopy or homologous models. To date, SB models are the most accurate approach for identifying compounds in screening, but also the most computationally intensive. The LB approach is more efficient in the screening stages, mainly due to the contribution of ML and DL algorithms, which will be discussed in more detail in the next section.

1.1.1 Virtual Screening

Virtual screening can be used as a low-cost alternative to HTS or as a supplement to the HTS procedure. Virtual screening is used in the latter to screen vast libraries of compounds for possible actives, lowering the size of the library before moving on to more expensive HTS. Because virtual screening does not involve the actual manufacturing of compounds, unlike HTS, it is not constrained by the chemical space that can be tested. Virtual screening, on the other hand, necessitates experimental data, such as a protein structure for structure-based virtual screening (Docking methods) or a list of known actives for ligand-based virtual screening. These two are the two major approaches for virtual screening [20, 21] .

Structure-Based Virtual Screening (SBVS), which focuses on screening the database of compounds within the boundaries of a target active site, is perhaps the most dynamic branch of virtual screening. Such methods have the advantage that searches are no longer limited to the predicted binding modes of known ligands. Instead, entire receptor sites can be studied, allowing a wide range of potential receptor-ligand interactions to be sampled. As a result, new chemotypes can be discovered that bind to the receptor in previously unknown ways. This advantage comes with a substantial research time problem. In order to keep up with the ever-increasing speed of biological screening methods, searches must be fast (no more than 5-10 minutes per compound for even the most computationally equipped, substantially less for the least equipped) [1]. As in most disciplines, accuracy is compromised by speed. For high-throughput predictions, sophisticated energy functions, such as those using first-principles approaches for protein-ligand affinity calculations, take too long. Consequently, protein-ligand interactions have been described using simpler scoring algorithms. To produce a variety of SBVS tools, these are coupled with a number of rapid docking procedures in binding mode.

Docking is essentially a geometric research problem. In fact, the conformation of the binding site of a given protein is well known, whereas the conformational structure of its ligand is not. Consequently, most docking techniques focus on studying the flexibility of the ligand while keeping the structure of the protein rigid. The main source from which to obtain 3D structures of proteins is the Protein Data Bank (PDB) [22]; these are obtained mainly by X-ray crystallography or NMR spectrography, deposited by biologists and biochemists worldwide, are in the public domain and are freely accessible.

There are several SBVS techniques, each focusing on a different aspect of research and input data. Clique detection [23, 24] is a technique that is used on small fragments or sets of explicitly generated ligand conformations, which are well suited to this type of search. These techniques can be used to search for distance-compatible matches of protein and ligand features, for example complementary hydrogen bonding interactions, distances or volume segments. The most widely used tool for clique search docking algorithms is the DOCK programme [25, 26, 27]. DOCK applies distance-compatible matching searches that incorporate clique detection algorithms for rigid-body docking (see figure 1.2). Site points mapping the molecular features of the binding site are matched to the atomic centres of the ligand. The initial orientations of the ligand in the receptor site are generated using distance-compatible matches of user-definable size (usually 3-4). The final position of the ligand is then determined through optimisation against the selected scoring function.

A further approach for structure-based virtual screening is based on stochastic techniques and in particular one of the most widely used is Monte Carlo simulation. This, in contrast to the combinatorial technique described above, starts with a configuration of the initial molecule that will vary towards conformations with a favourable energy. Monte Carlo methods refer, in a very general sense, to any simulation of an arbitrary system using a computer algorithm explicitly dependent on a set of (pseudo) random numbers. The name, derived from the famous casino in Monaco, emphasises the importance of chance in

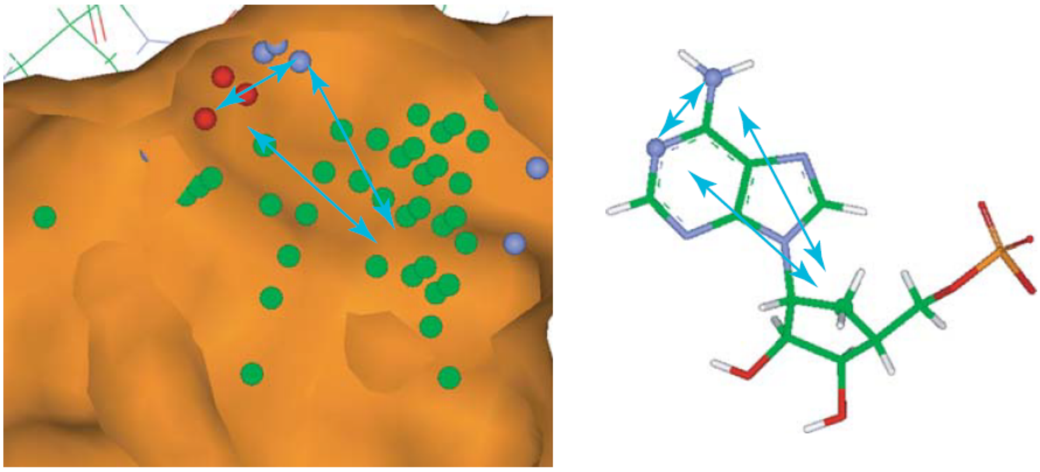


Figure 1.2: Clique detection as applied to the receptor site point/ligand atom paradigm employed in DOCK is depicted in this diagram. To make a match, distances and, optionally, atom chemistries must match. Furthermore, crucial sections that must be matched during clique detection can be allocated to atom clusters. It should be emphasized that, while this diagram focuses on receptor–ligand site points, it nevertheless serves as a useful graphic illustration of clique detection in general. Image taken from [1]

.

the method. In a system with D degrees of freedom, for example, the thermal average of a quantity A associated with each microstate of the system in equilibrium at absolute temperature T is given by

$$\langle A\rangle = \frac{1}{Z}\int A(x)\mathrm{e}^{-\frac{E(x)}{T}}\,\mathrm{d}x$$

where x is a point in D-dimensional space representing the state of the system, $E(x)$ is the energy of state x, and $Z = \int \mathrm{e}^{-E(x)/T}\,\mathrm{d}x$ the partition function (units set so that the Boltzmann constant (K_B), which establishes the correspondence between quantities of statistical mechanics and quantities of thermodynamics, is set to 1).

Monte Carlo simulations choose conformations in such a way that the selection is biased towards conformations that are significantly populated at equilibrium. This is typically achieved by weighing the probability of occurrence of a given conformation to its Boltzmann factor through the application of the Metropolis criterion [28]. If the difference between the energy of the resulting conformation and the energy of the current conformation, ΔE, is negative (i.e., the energy of the resulting conformation is less than the energy of the current conformation), then the resulting conformation is accepted and becomes the new conformation in the chain. If ΔE is positive, on the other hand, a (pseudo) random number R with a value between 0 and 1 is generated. The resulting conformation is only accepted if $\mathrm{e}^{-\Delta E/T} > R$. If the resulting conformation from the move attempt is rejected, then the current conformation becomes the new conformation of the chain.

Abayagan et al. combined efficient internal coordinate representations of the protein and ligand with a Monte Carlo optimisation protocol in their ICM program [29], software that uses the Monte Carlo algorithm to minimise energy functions in torsional space. A similar approach is used by the QXP software by applying it to small molecule superimposition [30]. An alternative search technique called 'tabu search' is used in the PRO-LEADS programme [31]. Starting from a random structure, new structures are created by random moves. A tabu list is maintained containing the best and most recent binding configurations found. New configurations generated that resemble those in the tabu list are rejected unless they show higher scores than the configurations in the tabu region. Consequently, sampling performance is improved as previously sampled configurations are avoided. AUTODOCK [32, 33] uses a variant of the Monte Carlo approach called simulated annealing to identify ligand docking poses. An alternative to the stochastic approach in AUTODOCK and GOLD [34] is provided by genetic algorithms. In GOLD two bit strings are used to represent the docking configuration. The first contains information about the conformation of the ligand by defining the twist of each corner of each of the freely rotating bonds, while the second contains information about the map of the relevant hydrogens and atoms in the molecule. A fitness function is used to provide the scoring function taking into account the evaluation of the hydrogen bonds, the internal energy of the ligand and the vdW energy of the protein-ligand complex.

Other approaches to SBVS are based on algorithms that exploit ligand and protein flexibility, respectively. Lorber and Shoichet [35] introduced the DOCK approach, based on protein flexibility, in which a database of 300 conformations with predefined substructures are tested each time, generating a ranking with rapid results compared to single-molecule docking. Thomas et al. [36] introduced a variant of this approach in which substructures are filtered according to similarity calculated on predetermined pharmacophoric fingerprints. Cavasotto and Abagyan created an IFREDA programme [37] that uses the flexibility protein in virtual screening. By docking flexible ligands to a flexible receptor, IFREDA generates a discrete set of receptor conformations, which are then used to perform subsequent docking and scoring of flexible ligands and rigid receptors. This is followed by a fusion and shrinkage step, in which the results of multiple virtual screenings are condensed to improve the enrichment factor. Techniques based on protein flexibility are computationally expensive, especially considering the large number of molecules to be tested in the screening steps. In fact, rigid protein-based approaches have been favoured over the years. Structure-based techniques are certainly the most reliable for the characterisation of a ligand, but as can be deduced from the techniques described above, they are not efficient for research on large datasets, in contrast to ligand-based approaches which have proved to be less expensive and faster in screening large datasets.

Ligand-Based Virtual Screening abandons knowledge of the structure of the target protein, focusing exclusively on the similarity of structure of the ligands. The initial phase of LBVS requires converting chemical structures into a machine-readable format. The three-dimensional molecular structure must be converted into an embedding that is able to retain all structural information. Over the years, several ways have been proposed to represent the structure of molecules but connection tables have emerged as the most important representation for two-dimensional (2D) information. The structure data (SD) file is probably the most widely used of these. This file was created to allow a huge number of molecules and their accompanying data to be moved through databases. A connection table stores the x and y coordinates of the atoms according to bond length (z coordinates can be included when three-dimensional (3D) data is to be saved), as well as the associated atom type, chirality and bond connection data. This file format has proved excellent for transporting chemical information, but has proved inflexible for the various screening algorithms developed over the years, so other representations such as SMILES and Molecular Fingerprints have been created. Simplified Molecular Input Line Entry System (SMILES), is a line notation (a typographical method using printable characters) for entering and representing molecules. While a SMILES string contains the same information as an extended connection table, it is in essence a chemical language with a vocabulary (symbols for atoms and bonds) and grammatical rules (e.g. for recognising substitution patterns). SMILES representations of structure can in turn be used as ’words’ in the vocabulary of other languages designed for chemical storage. Molecular Fingerprints are another linear representation of molecules that does not use letters to describe the atoms within the molecule, but uses the 0 and 1 bits to indicate the presence or absence of a specific chemical group within the molecule.

These descriptors form the basis of all LBVS methodologies, starting with chemical substructure searching (SS), which has been successfully applied with 2D and 3D data. 2D substructure analysis (or subgraph isomorph matching; see Figure 1) is still perhaps the most routinely used of all screening techniques with a history dating back to the 1950s with the development of the backtracking algorithm [38, 39]. Backtracking, also known as atom-by-atom searching, operates by matching an initial substructure query node with a database structure, before exploring nodes outside it. The algorithm moves from atom to atom until the last node is successfully matched, or until a mismatch is found, in which case it returns to the last correct choice point. To speed up the backtracking process, refinements have been devised such as the early pruning of unfruitful branches [40, 41]. Even with these modifications, the size of the search space of the recursive matching algorithm can prove to be very large. However, the technique is one of the few that can guarantee finding the answer for any query-target combination, so much so that it is a common fallback for difficult searches.

The search for 3D substructures abandons backtracking and is dominated almost entirely by pharmacophore mapping [42, 43, 44]. A pharmacophore is commonly defined as a 3D geometric arrangement of molecular features or fragments that form a necessary but not sufficient condition for biological activity. These features most typically represent functional groups (donor, acceptor, hydrophobic, etc.), but may also include larger 2D substructures, planes, vectors, exclusion volumes and other features. The use of pharmacophore for substructure search was first investigated by van Drie et al. using ALADDIN [45], the first 3D substructure search programme. Marriott et al. employed the DISCO program [46, 47] together with 3D conformation models of known M3 muscarinic antagonists to determine the necessary pharmacophore model. DISCO takes a set of low-energy conformations for each active molecule as input, and for each conformation, the likely locations of the pharmacophore site are automatically generated. DISCO's ability to define the positions of the protein's likely hydrogen bond donors and acceptors is a useful feature. This is significant because ligands may approach the same polar site location from different directions, which is difficult to account for when using simple atomic superposition. The search for substructures, both two- and three-dimensional, brings significant complexity to the VS process. One way in which this has been mitigated is through the application of whole-molecule similarity comparisons that remove the need for specific feature selection, introducing new 1D descriptors such as Fingerprints that have nowadays proven to be among the best embeddings for LBVS applications, especially with the advent of new artificial intelligence-based methodologies, which will be discussed in subsequent chapters.

1.1.2 Drug Repurposing

Drug Repurposing (DR) (also called Drug Repositioning, Reprofiling or Retasking) is a strategy used to identify new uses for approved or late-stage drugs that are outside the scope of the original indication. This approach has several advantages over developing an entirely new drug for a specific indication. Firstly, and perhaps most importantly, the risk of failure is reduced; since the repurposed drug has already been shown to be sufficiently safe in preclinical models and in humans, if preliminary studies have been completed, it is less likely to fail in subsequent efficacy studies, at least from a safety perspective. Secondly, since most of the preclinical testing, safety evaluation and, in some cases, formulation development will already have been completed, the drug development time can be reduced. Thirdly, less investment is required, although this will vary considerably depending on the stage and process of development of the repurposing candidate. Regulatory and phase III costs for a repurposed drug may be similar to those for a new drug in the same indication, but there may be significant savings in preclinical and phase I and II costs. These advantages, when combined, have the potential to result in a less risky and faster return on investment in repurposed drug development, with lower average associated costs once failures are taken into account (in fact, the cost of bringing a repurposed drug to market has been estimated at $300 million on average, compared to an estimate of $2-3 billion for a new chemical entity [48]). Finally, repurposed drugs may uncover new targets and pathways that can be explored further [2]. Drug repurposing typically consists of 3 steps, before being able to identify a drug that can be reused on another disease. The first and most critical step is to identify a candidate molecule. The second step is to evaluate the drug's efficacy in preclinical models (in vivo or in vitro tests) which, if successful, lead to step 3, i.e. evaluation in phase II clinical trials. Since the first step is the most important, but also the most expensive, new approaches have been developed over the years to speed up this phase, and in particular computational methods, thanks to the possibility of analysing data of different nature (e.g. gene expression data, chemical structure, genotypic or proteomic data or Eletronic Health Records (EHR)) have proved to be the strong point in the approach to this task. One of the most popular computational approaches to date is *signature matching*, which is based on the comparison of a unique characteristic between drugs or the comparison of a drug with a diseased phenotype. The signature can be derived from three different types of data: transcriptomic, proteomic and metabolomic. Transcriptomic data can be used to perform a drug-disease or drug-drug comparison, in both cases to identify similarity. For two drugs, sharing a transcriptomic signature may imply a shared therapeutic application regardless of their structural similarity or sharing of similar chemical structures. Because of the effectiveness of this approach, the cMap (Connectivity Map) was created in 2006, which contains expression profiles generated by dosing more than 1,300 compounds in a range of cell lines [49].*Molecular docking* is a further computational strategy used for drug repurposing to predict the complementary binding site between ligands and targets. In contrast to the traditional approach described in the previous chapter, inverse docking is used in DR, where multiple receptor sites are interrogated against a specific drug in order to identify new interactions. A further approach is based on *Genome-Wide Association Studies (GWAS)*, which is based on the search for variations associated with common diseases; this has been made possible by the technological leap forward that has been achieved with new genotyping techniques and the completion of the Human Genome Project [50, 51], the worldwide project for the complete sequencing and mapping of the human genome. The results obtained from this research are not always easy to interpret, especially in DD, which is why they are often associated with pathway analysis or network mapping, which provides information on the proteins involved in the signal cascade, also clarifying the result obtained through GWAS. A further method is the Retrospective clinical analysis which is based on a systemic analysis of data that can be obtained from various sources, including EHR data and clinical trial data. EHRs contain a considerable amount of structured data such as diagnostic and pathophysiological data, including data obtained from laboratory results and unstructured data such as clinical descriptions of patient signs and symptoms or data imaging. Data belonging to this category are not always open access, often bound by ethical constraints and legal restrictions. In 2016, the EMA [18] started to give free access to data obtained from clinical trials submitted by pharmaceutical companies for the free use of the academic community.

In addition to computational methods, experimental approaches are also used in drug repurposing, the two most widely used being: i) Binding assays to identify target interaction using proteomic techniques such as chromatographic affinity and mass spectrometry. The Cellular ThermoStability Assay (CETSA) technique, for example, has been introduced as a method of mapping target engagement in cells using biophysical principles involving the thermal stabilisation of target proteins by drug-like ligands with the appropriate cellular affinity; ii) phenotypic screening can identify compounds that show disease-relevant effects in model systems without prior knowledge of the target(s) involved. In the context of drug repurposing, if the compounds screened are approved or in the process of being approved, this may indicate repurposing opportunities that can be readily seized.

A summary of all approaches to Drug Repurposing is shown in the figure 1.3.

The heterogeneity of the data used in the various approaches described above fits well with the advent of new artificial intelligence technologies such as machine learning and deep learning. Within my research activity, I have focused on the application of the latter technologies for Drug Repurposing for a current task such as SARS-Cov-2 infection within the CLAIRE (Confederation of Laboratories for Artificial Intelligence in Europe) Task Force, testing the chemical descriptors developed over the last 3 years.

1.2 Machine Learning and Deep Learning applications in Drug Discovery

In recent years, the field of Artificial Intelligence (AI) has moved from largely theoretical studies to real-world applications. Credit for this explosive growth has come with the increased availability of increasingly high-performance computer hardware such as new GPUs that make parallel processing faster, especially in numerically intensive computations [52]. The goal of a good ML model is to generalize well from the training data to the available test data. Generalisation refers to how well the concepts learned by the model apply to data not seen by the model during the training phases. There are several algorithms available today, but they are typically grouped into two broad categories: *Supervised* and *Unsupervised* algorithms. These models vary in their prediction accuracy, training speed and the number of variables they can handle. Algorithms must be chosen carefully to ensure that they are suitable for

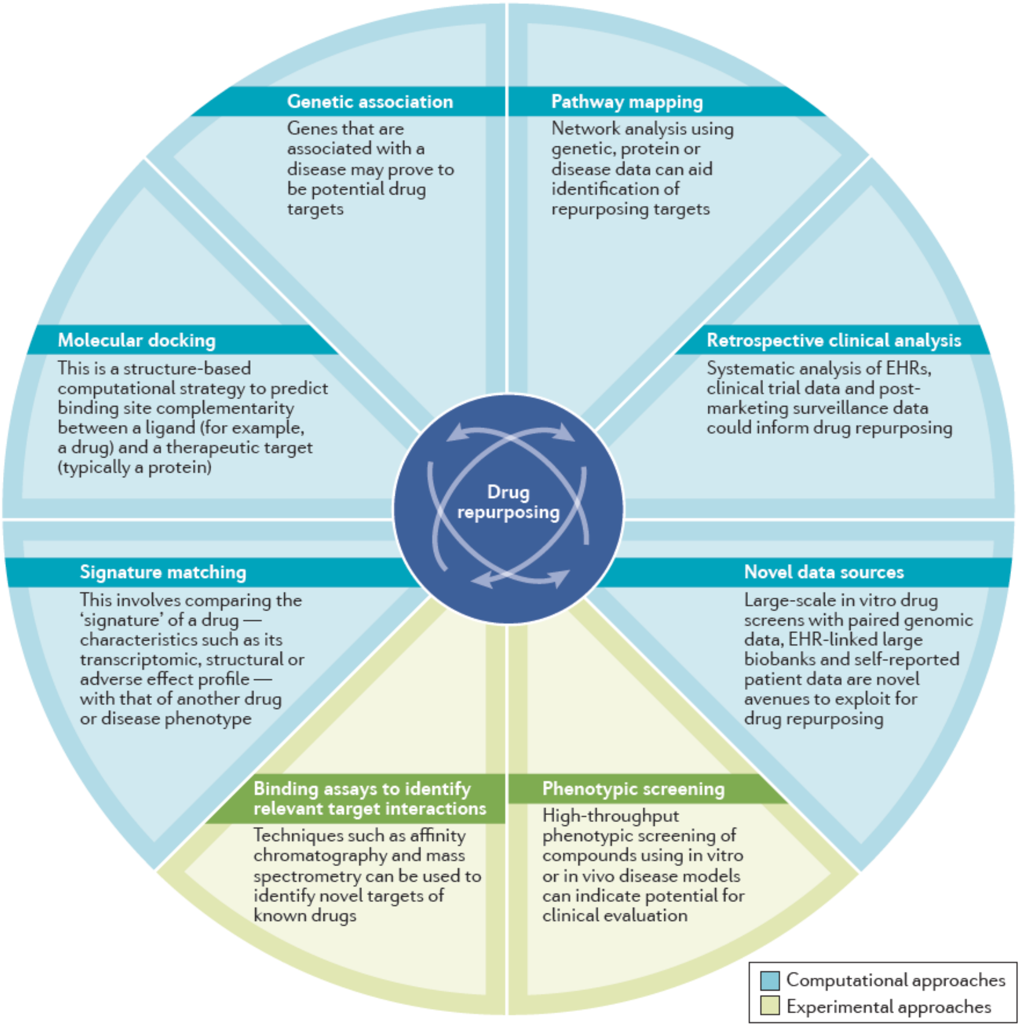

Genetic association
Genes that are associated with a disease may prove to be potential drug targets
Pathway mapping
Network analysis using genetic, protein or disease data can aid identification of repurposing targets
Molecular docking
This is a structure-based computational strategy to predict binding site complementarity between a ligand (for example, a drug) and a therapeutic target (typically a protein)
Retrospective clinical analysis
Systematic analysis of EHRs, clinical trial data and post-marketing surveillance data could inform drug repurposing
Drug repurposing
Signature matching
This involves comparing the 'signature' of a drug — characteristics such as its transcriptomic, structural or adverse effect profile — with that of another drug or disease phenotype
Novel data sources
Large-scale in vitro drug screens with paired genomic data, EHR-linked large biobanks and self-reported patient data are novel avenues to exploit for drug repurposing
Binding assays to identify relevant target interactions
Techniques such as affinity chromatography and mass spectrometry can be used to identify novel targets of known drugs
Phenotypic screening
High-throughput phenotypic screening of compounds using in vitro or in vivo disease models can indicate potential for clinical evaluation
🟦 Computational approaches
🟨 Experimental approaches

Figure 1.3: Approaches used in drug repurposing. Various computational approaches can be used individually or in combination to systematically analyse different types of large-scale data to obtain meaningful interpretations for repurposing hypotheses. Challenges for such analyses are discussed in BOX 5. Experimental approaches can also be used to identify repurposing opportunities. EHR, electronic health record. Image taken from [2]

.

the problem at hand and the amount and type of data available. A summary of the main known state-of-the-art algorithms and models is shown in figure 1.4.

The initiation of a drug development programme requires the identification of a target with a plausible therapeutic hypothesis. The selection of this target on the basis of the available evidence is called target identification and prioritisation. Having made this preliminary choice, the next step is to validate the role of the chosen target in the disease using physiologically relevant ex

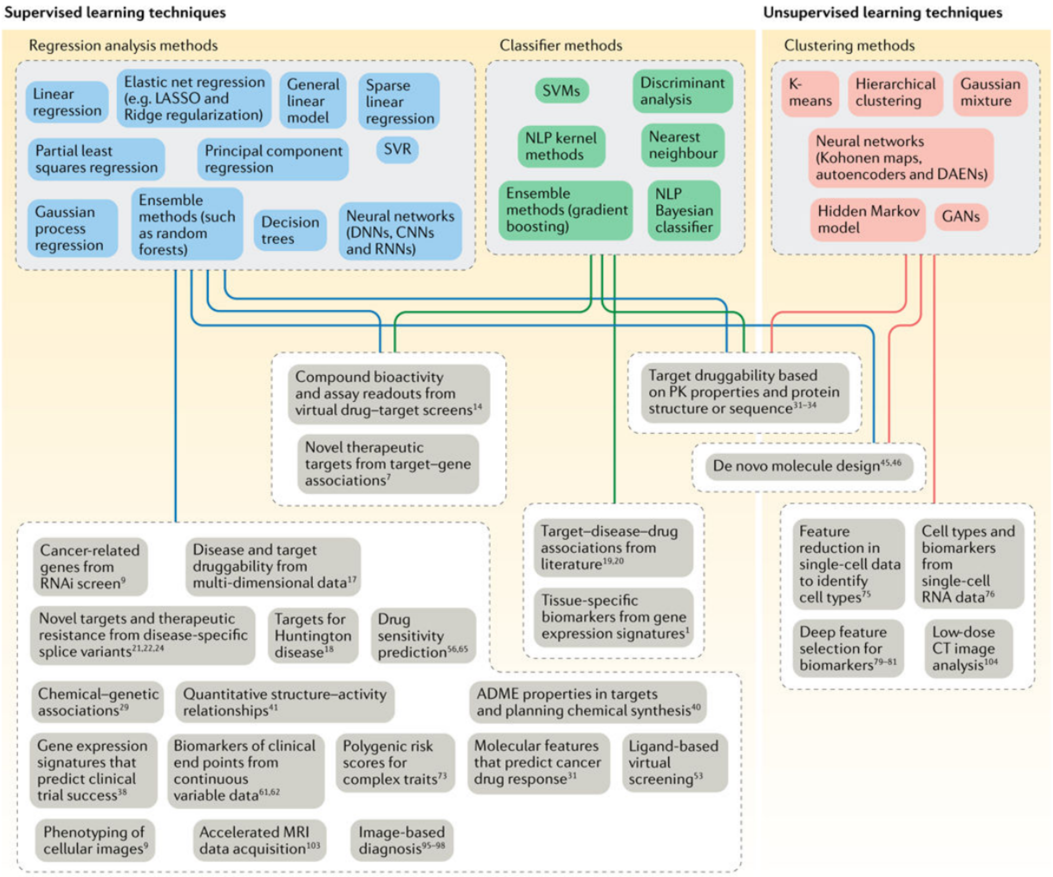

Supervised learning techniques
Unsupervised learning techniques
Regression analysis methods
Linear regression
Elastic net regression (e.g. LASSO and Ridge regularization)
General linear model
Sparse linear regression
Partial least squares regression
Principal component regression
SVR
Gaussian process regression
Ensemble methods (such as random forests)
Decision trees
Neural networks (DNNs, CNNs and RNNs)
Classifier methods
SVMs
Discriminant analysis
NLP kernel methods
Nearest neighbour
Ensemble methods (gradient boosting)
NLP Bayesian classifier
Clustering methods
K-means
Hierarchical clustering
Gaussian mixture
Neural networks (Kohonen maps, autoencoders and DAENs)
Hidden Markov model
GANs
Compound bioactivity and assay readouts from virtual drug–target screens[14]
Novel therapeutic targets from target–gene associations[7]
Target druggability based on PK properties and protein structure or sequence[31–34]
De novo molecule design[45,46]
Cancer-related genes from RNAi screen[9]
Disease and target druggability from multi-dimensional data[17]
Novel targets and therapeutic resistance from disease-specific splice variants[21,22,24]
Targets for Huntington disease[18]
Drug sensitivity prediction[56,65]
Target–disease–drug associations from literature[19,20]
Tissue-specific biomarkers from gene expression signatures[1]
Feature reduction in single-cell data to identify cell types[75]
Cell types and biomarkers from single-cell RNA data[76]
Deep feature selection for biomarkers[79–81]
Low-dose CT image analysis[104]
Chemical–genetic associations[29]
Quantitative structure–activity relationships[41]
ADME properties in targets and planning chemical synthesis[40]
Gene expression signatures that predict clinical trial success[38]
Biomarkers of clinical end points from continuous variable data[61,62]
Polygenic risk scores for complex traits[73]
Molecular features that predict cancer drug response[31]
Ligand-based virtual screening[53]
Phenotyping of cellular images[9]
Accelerated MRI data acquisition[103]
Image-based diagnosis[95–98]

Figure 1.4: Machine learning tools and their use in drug discovery The machine learning techniques that have been employed to answer the drug discovery questions mentioned in this Review are depicted in this diagram. Unsupervised techniques are used to construct models that enable data grouping, whereas supervised learning techniques (regression and classifier methods) are used to solve queries that demand prediction of data categories or continuous variables.

vivo and in vivo models (target validation). This approach, however, requires a considerable amount of time. Modern biology is increasingly rich in data. This includes human genetic information in large populations, transcriptomic, proteomic and metabolomic profiles of healthy individuals and those with specific diseases, and highly clinical images. The ability to capture these large data sets and reuse them through public databases presents new opportunities for early target identification and validation. However, these multidimensional datasets require appropriate analytical methods to produce statistically valid models that can make predictions for target identification, and this is where ML can be exploited [52]. For example, Costa et al. [53] constructed a Random Forest based meta-classifier to identify candidate genes as targets, using topological networks of protein-protein interaction, and transcriptional and metabolic interaction. A further example of the application of ML models can be found in the work of Ament et al. [54] where a regression model and LASSO regularization was used to identify Transcription Factors (TFs) involved in Huntington's disorder, resulting in a set of TFs to be used as a starting point for disease therapies.

The prediction of the bioactivity of new ligands is another aspect of drug research where machine learning is routinely applied. Carpenter and Huang [55] propose several uses of machine learning techniques for virtual screening to find compounds with a possible key role against proteins implicated in Alzheimer's disease. In order to evaluate the toxicity of 3486 Per- and Polyfluoroalkyl Substances, Cheng and Carla A. Ng [56] employed Machine Learning techniques to categorize their bioactivity. As can be seen, Machine Learning algorithms have proven to be very versatile in the early stages of drug discovery, with Support Vector Machine (SVM) and Random Forest (RF) being the two best performing algorithms in the state of the art [57, 58, 59, 60, 61].

In spite of the flexibility demonstrated, ML models present a great limitation when the operator is not able to consciously isolate the set of features to be used for training. This limitation has been overcome by the advent of deep neural architectures that have proved capable of using complex data (such as, for example, images) whose peculiar features will be extracted directly from the network. In terms of performance, DNNs have proved to be outperforming 'shallow' machine learning models on many of the Drug Discovery tasks and in particular LBVS. To date, DL has been employed in all areas of life science research: Angermueller et al. [62] report on DL approaches in computational biology, while Anwar et al. [63] present an in-depth report on DL for medical imaging. Deep Learning solutions have been proposed for all stages of the drug design workflow [64], and AI-based techniques such as decision support systems and robotic platforms are likely to work in tandem with human medicinal chemists to undertake drug discovery in the near future [65]. In the area of Drug Discovery the applications are manifold, for example,

Wallach in [66] presented *AtomNet*, which is considered the first CNN for structure-based screening. In [67] a CNN for learning circular fingerprints [68] from molecular graphs is proposed, and some experiments are performed to demonstrate their effectiveness in both solubility and drug efficacy prediction. In [69] *DeepVS* is presented: this CNN makes use of the notion of *context* of an atom in the protein-compound complex that is a vector representation of the structural properties of its neighborhood. In [70] the SMILES notation [71] describing the compound is used to create a *feature* matrix where each column is a one-hot encoding of the presence of a particular SMILES symbol at a certain position. This representation is fed to a CNN to detect the "chemical motifs" that are relevant to the binding substructures. In Jimenez-Carretero et. al. [72] research in 2018, they used a deep Convolutional Neural Network (CNN) to train the model to predict the toxicity of images of DAPI-stained cells pretreated with a group of drugs with different toxic mechanisms. Goh et. al. [73] developed "Chemception", a deep CNN for predicting chemistry, using only two-dimensional drawings of molecules. Although Chemception is slightly inadequate in terms of predicting toxicity, there is still room for improvement. All these DL models should be iteratively refined with new experimental data to increase model reliability and predictive power. In recent years, deep learning techniques, and in particular CNNs have gained increasing impact on drug and VS design due to the tremendous increase in prediction accuracy at any stage of this process [64, 66]. DNNs have also been used to predict biological activity, ADMET characteristics, and physicochemical factors, displaying consistent and robust prediction skills with high sensitivity when applied to a variety of targets [74, 75]. CNNs have also been used to predict features like hydrocarbon kinetic energy as a function of electron density[76].

In spite of everything, Virtual Screening remains, without doubt, one of the most studied topics in the field of DL applications. The reader is referred in particular to the work of Kimber et al. [77] for structure-based approaches, and to the paper by Sydow et al [78] for ligand-based ones.

1.3 Motivation and goals

In recent years, the pharmaceutical industry has been investing resources and time in the search for innovative methodologies capable of raising the success rate in the field of Drug Discovery. As described in the previous sections, the DD domain is very complex and is continuously evolving following the technological advancement that has been taking place in recent years. The approaches and input data available to the scientific community are manifold and it is the task of researchers to try to improve the efficiency of the process. In my research activity, I have been engaged in studying the most commonly used molecular descriptors in the state of the art, in order to fully understand their structure and be able to create new, more efficient ones.

The state of the art approaches that provide advantageous results are based on computational methods; in particular those based on Machine Learning (ML) and Deep Learning (DL) [79], have shown great potential, thanks to the greater accessibility to large data sets each containing chemical information. The latter are very heterogeneous and need to be converted into suitable input data for the training phases of neural networks. The variety of chemical data requires the researcher to know its origin, structure and properties in order to maximise its effectiveness for the task to be solved, in the different application domains. In fact, Drug Discovery is composed of different phases with problems that require different information and technologies in continuous evolution. The objective of this work focuses on one of the first phases of DD, namely Virtual screening (VS), and in particular on the descriptors that are used in this domain at the state of the art. Indeed, in VS, structural descriptors referring in some way to the potential biological activity on a specific target are used to identify a set of compounds eligible as lead compounds [1] from various online databases such as ChEMBL [80, 81]. The main goal of my research is to extend the knowledge of molecular descriptors present in the state of the art in the above mentioned field, trying to provide a contribution not only theoretical but above all concrete, integrating knowledge of biological, chemical and computer science developed during this research path, providing two new embeddings usable in the field of VS:

- **EMBER - embedding multiple molecular fingerprints**: a 3D embedding consisting of 7 molecular fingerprints, RDKit, Morgan, AtomPair, Torsion, Layered, FeatMorgan and ECFP4, each of which contains different structural information for each molecule [82].

- **NMR-Like**: an innovative descriptor based on H-NMR spectra that is proposed as a tool capable of both screening and characterising the chemical groups directly involved in the interaction with the protein.

Both presented descriptors were compared with molecular fingerprints, in terms of performance, using Deep Neural Network (DNN) and Machine Learning (ML) algorithms in the target-specific bioactivity classification task. To further deepen the level of understanding, Explainability experiments were conducted, starting with the architectures with the best classification performance, in order to obtain the contributions of the molecular features within each descriptor. Thanks to this possibility, NMR-Like, is proposed not only as embedding for the classification of bioactivity but also as a tool for the prediction of the chemical groups directly involved in the interaction between the active molecule and the protein, further decreasing the time needed to identify the lead compound, assisting the studies of molecular docking necessary to study the correct key-lock coupling between the molecule and the protein.

1.4 Dissertation outlines

The remainder of the dissertation is organized as follows. **Chapter 2** contains a detailed description of the molecular descriptors used at the state of the art in drug discovery applications. **Chapter 3** deals with the selection of the biological target used in the study and describes in detail the preprocessing performed in order to obtain the various datasets used in the various phases of experimentation. In **Chapter 4**, the approach based on molecular fingerprints is described in detail, starting from the dimensional exploration up to the creation of EMBER. **Chapter 5** shows in detail the potentialities of the NMR-Like descriptor in the bioactivity classification task and the possible applications in conjunction with molecular docking. In **Chapter 6** we show the applications of EMBER and NMR-Like in the Drug Repurposing domain. In **Chapter 7** some conclusions are drawn.

Chapter 2

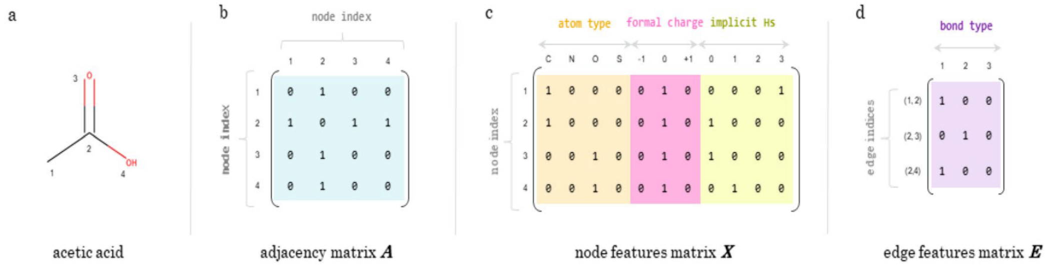
Molecular Descriptors

This chapter provides a detailed description of the molecular descriptors that are used in ML and DL approaches in virtual screening and drug repurposing. Converting experimentally obtained chemical information into data that can be used as input in chemoinformatics is a complex and important process. Over the years, several forms of machine-readable data have been developed, but today the most widely used are: Graph representation (such as MolFiles, Structure Data Format (SDF)), SMILES (Simplified Molecular Input Line Entry System), InChI (and InChIKey) and Fingerprints, which are the most widely used molecular representation in the domains I have explored during my research activity. Each type of data will be treated and described in detail in this chapter.

2.1 Graph representation

The graph is the first form of representation one associates with molecules, and in the world of computational chemistry it is also the starting point for the construction of the other chemical descriptors discussed below. The idea behind the molecular graph representation lies in mapping the atoms and bonds that make up the molecule into a set of nodes and arcs, typically in a 2D structure which can be extended using 3D information (e.g. atomic coordinates, bond angles and chirality).

A graph is formally defined as a tuple $G = (V, E)$ of a set of nodes V and a set of arcs E, where each of the arcs $e \in E$ links a pair of nodes in V. In a molecule intuitively the nodes represent the atoms while the arcs represent the bonds which in this case are undirected. This information has to be described in a way that can be handled by the computer while retaining fundamental chemical information such as how the atoms are connected to each other, the identity of each atom and the identity of each bond. Adjacency matrices A are typically used to describe the connection between atoms; since aij is an element of A, a$_{ij}$ = 1 indicates that there is a bond between nodes v$_i$ and v$_j$ in the molecular graph **G**, while a$_{ij}$ = 0 indicates the absence of a bond between these two nodes [4]. The identity of each atom, on the other hand, can be represented as a node features matrix **X**¸where each row of **X** corresponds to node v$_i$ (i.e. an atom of the molecule) in G which also refers to a node feature vector xi whose length corresponds to the number of features of the atom that are represented. The identity of the bonds is presented in the form of an edge feature matrix E, where each row of E corresponds to an arc e$_{ij}$ = (v$_i$, v$_j$) in **G**, and refers to an edge feature vector e$_{ij}$ for that specific arc, representing the features assigned to that bond. An example of the graph representation is shown in figure 2.1.



$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 1 & 0 & 0 \\ 2 & 1 & 0 & 1 & 1 \\ 3 & 0 & 1 & 0 & 0 \\ 4 & 0 & 1 & 0 & 0 \end{array}$$

$$\begin{array}{c|ccccccccccc} & C & N & O & S & -1 & 0 & +1 & 0 & 1 & 2 & 3 \\ \hline 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 2 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 3 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 4 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{array}$$

$$\begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline (1,2) & 1 & 0 & 0 \\ (2,3) & 0 & 1 & 0 \\ (2,4) & 1 & 0 & 0 \end{array}$$

Figure 2.1: Example graph representation for acetic acid. a Graph representation of acetic acid with nodes numbered from one to four. b Example adjacency matrix, A, for an acetic acid graph with the corresponding node ordering on the left. c Example node features matrix, X, which one-hot encodes a few selected properties. d Example edge features matrix, E, where each edge feature vector is a one-hot encoding of single, double, or triple bonds. “Implicit Hs” stands for the number of implicit hydrogens on a given node

Two of the closest forms to the graph representation just described are connection tables and MDL file formats.

2.2 Connection tables

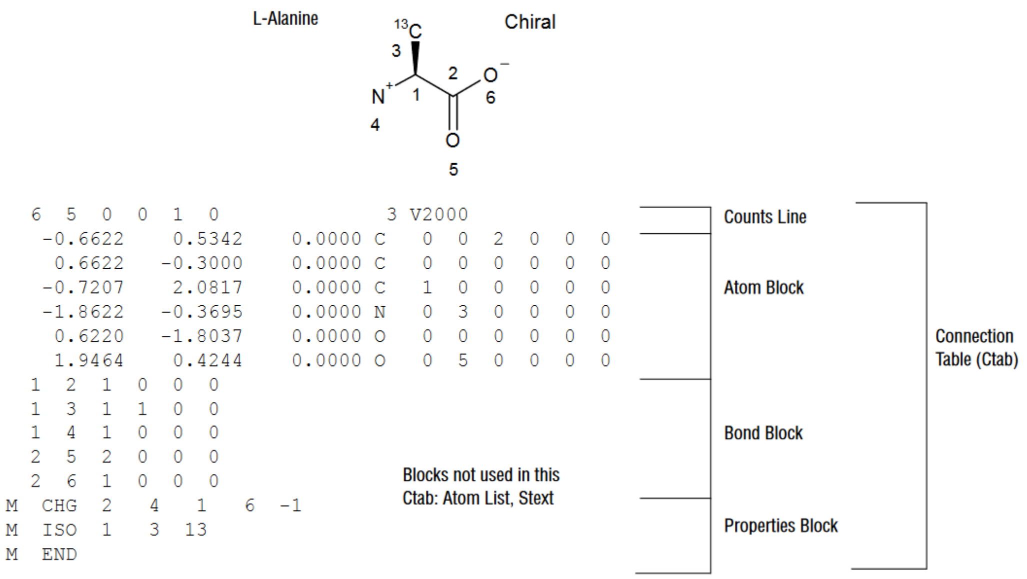
Connection tables (Ctab) are the basic element for all chemical tables files (CT-file) [83] and contain information describing the structural relationships and properties of a set of atoms. The latter may be totally or partially connected by bonds. Each collection of atoms can for example describe molecules, molecular fragments, substructures, substituent groups and polymers. Typically a Ctab consists of 6 different parts:

- **Counts line**: this is typically the first line in the table and gives an overview of the structure by specifying the atoms and bonds present.
- **Atom block**: This block describes the identity of all atoms present, such as atomic symbol, charge, stereochemistry and associated hydrogens as a list with arbitrary indices.
- **Bond block**: This block contains information on the bonds characterising the molecule or, in the case of molecular fragments or disconnected atoms, contains information on how to identify them. The order of the bonds is represented by an additional column. Together with the Atom block, it forms the core of Ctab.
- **Atom list block**: This is a list containing information about atoms, such as their atomic number, the number of bonds each one makes and the space they occupy.
- **Stext block**: This is a block that is used by desktop programmes and describes structural information in the form of text.
- **Properties block**: This is the block that describes a set of additional properties of the molecule. For example, the charge, radicals, isotopes, number of rings, count of substituents present, unsaturated atoms, the attachment point of each atom and the order in which they are reported are all reported in this block.

An example of Ctab is shown in the figure 2.2.

Ctab have been used as a core to develop other representations of chemical molecules, both 2D and 3D. An example of this is the Molfile format, which was developed by MDL and is commonly known as CTfiles (Chemical Table files). In addition to the Ctab features, Molfiles are very extensible formats and are for this reason widely used for the transfer of chemical information. Molfiles can then be structured into the most common Structure Data Format (SDF) files, which are currently widely used to describe macromolecules such as proteins.

As can be seen from the structure of the CTab shown in the figure and the description of the various blocks, this class of descriptors has a complex structure that varies according to the information contained within it. The extensibility that characterises them is useful in the transfer of chemical information, much less so in applications in the VS and DR domains. For this



```
  6  5  0  0  1  0            3 V2000
   -0.6622    0.5342    0.0000 C   0  0  2  0  0  0
    0.6622   -0.3000    0.0000 C   0  0  0  0  0  0
   -0.7207    2.0817    0.0000 C   1  0  0  0  0  0
   -1.8622   -0.3695    0.0000 N   0  3  0  0  0  0
    0.6220   -1.8037    0.0000 O   0  0  0  0  0  0
    1.9464    0.4244    0.0000 O   0  5  0  0  0  0
  1  2  1  0  0  0
  1  3  1  1  0  0
  1  4  1  0  0  0
  2  5  2  0  0  0
  2  6  1  0  0  0
M  CHG  2   4   1   6  -1
M  ISO  1   3  13
M  END
```

Figure 2.2: An example of the organisation of a CTab using Alanine.

reason, different linear representations have been developed from these graph representations, whose embedding is more suitable for the task. These linear descriptors will be discussed in the following sections.

2.3 SMILES

SMILES (The **S**implified **M**olecular **I**nput **L**ine **E**ntry **S**ystem), was developed in 1988 by Weininger et al. [71]. It has since become the most popular linear notation because it can represent molecules in a simple way. To explain how SMILES was generated, it is necessary to introduce the concept of a molecular graph. The idea that guides the representation of the molecular graph is to map the atoms and bonds that make up a molecule into sets of nodes and arcs, imagining to treat the atoms of a molecule as nodes and the bonds as arcs. In typical graph representations, nodes are represented using circles or spheres, and edges using lines. In the case of molecular graphs, the nodes are often represented using letters indicating the type of atom (as on the periodic table), or simply using points where the bonds meet (for carbon atoms). The representation of a molecular graph is formally a 2D object which can be expanded to represent 3D information (e.g. atomic coordinates, bond angles, chirality). However, any spatial relationships between nodes must be encoded as node and/or arc attributes, since nodes in a graph do not formally have spatial positions but only pairwise relationships. The SMILES notation system was later incorporated into the Daylight Chemical Information Systems [84] toolkit; the company is still maintaining it. The SMILES representation, which is non-unique and unambiguous, is obtained by assigning a number to each atom of the molecule that will represent the crossing order of the graph, where 1 is the initial node (user defined). The assignment of the initial node and the differential analysis of the graph can generate *canonical* and *randomized* SMILES as shown in Figure 2.3.

Initially, SMILES did not encode for stereochemistry, but an isomeric SMILES was later introduced, capable of describing it, and is now the default SMILES in many software programs. SMILES can therefore encode isomeric specifications, configurations around double bonds (Z or E), and configurations around tetrahedral centres as well as many other types of chiral centres that are rarely supported (e.g. allene, octahedral). However, a problematic set of structures to describe using SMILES notation are those that cannot be easily described using molecular graphs, such as organometallic compounds and ionic salts. There are many types of molecules that cannot be described by the graph model. These are any structure that contains any form of delocalised bonds, such as coordination compounds, as well as any molecule that contains one of the following: polycentric bonds, ionic bonds or and ionic salts. For example, organometallic compounds such as metallocenes or metal carbonyl complexes are difficult to describe using molecular graphs because their bonding pattern cannot be explained by valence bond theory. In other words, it would be difficult to describe bonds using only pairwise relationships between atoms. Solutions for dealing with plurivalent bonds have been introduced through the use of hypergraphs; in a hypergraph, the edges are sets of at least two atoms (hyperedges) instead of tuples of [85] atoms. However, the use of hypergraphs is not further discussed here as they are not currently popular in the field. For molecules in which the arrangement of atoms is constantly changing in 3D space, the graphical representation may not be meaningful, especially if pair bonds are broken and formed or if the structure is frequently reorganised. That is, for applications where one is limited to using a single static representation for a molecule that is in fact reorganising on the time scale of the problem (e.g. tautomers), then a single molecular graph representation would not be appropriate and may even be detrimental to solving the problem.

At the state of the art, despite being among the first molecular descriptors, SMILES are widely used with DL techniques in both the VS and general chemoinformatics domains [86]. For example, another deep learning model for

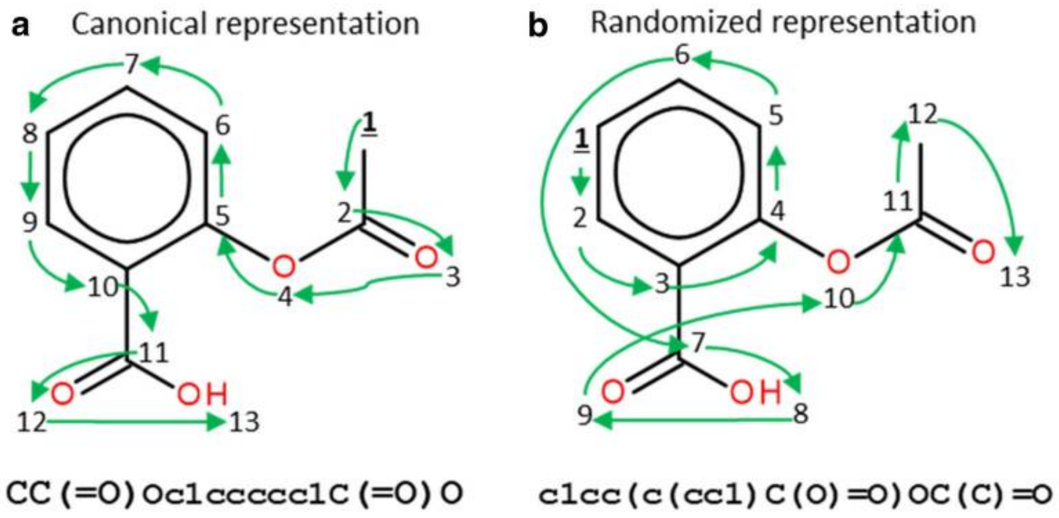


Figure 2.3: Canonical (a) and randomized (b) SMILES representations of aspirin. Both SMILES strings shown represent the same molecule but, as the atom numberings are different, the generated SMILES strings are, too. The original figure can be found in [3]

compound classification was developed by Hirohara et al. They created a distributed representation of compounds based on the SMILES notation, which linearly represents the structure of a compound, and applied the SMILES-based representation to a convolutional neural network in this method (CNN). They can process all types of compounds by incorporating a wide range of structure information thanks to SMILES, and learning the CNN representation automatically acquires a two-dimensional representation of the input features. The obtained model was tested on the TOX21 dataset, turning out to be better than the one that had won the TOX21 challenge [70]. Yadav and Jujjavarapu in 2021 presented a neural network-based methodology for classifying lipopeptides based on SMILES [87], while in 2021 Habib et al. [88] presented TarDict, a Random Forest-based classifier for predicting drug-target interaction, again using SMILES as input data.

2.4 InChI (and InChIKey)

InChI is the open source International Chemical Identifier introduced in 2006 and developed at the request of IUPAC. [89], the International Union of Pure and Applied Chemistry, with main contributions from NIST [90] and the InChI Trust [91]. The descriptor was constructed by trying to meet several requirements that are fully met by InChI:

- **Structure-Based Approach**: Anyone should be able to produce InChI from the structural formula of a chemical alone.
- **Strict uniqueness of the descriptor**: The same label indicates the same compound, so there can be no differently labelled compounds. This is achieved by following a well-defined canonical procedure of numbering the atoms.
- **Free accessibility of the descriptor**: It must be available to anyone wishing to develop a programme or new code using the identifier.
- **Applicability to the entire domain of organic chemistry**: this concept can be extended to inorganic chemistry.
- **Hierarchical approach**: Approach that allows the encoding of molecular structure with different levels allowing the inclusion of stereochemical, isotopic and tautomeric information.
- **Interoperability between large amounts of data**: Ability to produce a descriptor that can be used with large amounts of data.

InChI is based on the "classical chemical structure model" with some significant modifications and additions. The following principles form the basis of the InChI approach, or the "InChI model of chemical structure". A molecule is composed of atoms. Atoms are either skeletal (non-hydrogen atoms, as well as bridging hydrogen, as in diborane) or terminal hydrogen atoms (simply called "hydrogens"). Skeleton atoms are connected two by two by bonds and are characterised by their chemical element, whole formal charge, radical state, isotopic mass, associated implicit hydrogens and bonds with other skeleton atoms. The hydrogens may be linked to skeletal atoms or shared by a group of skeletal atoms (such groups may also share a negative charge). All bonds are simple bonds (connections). That is, they have no double, triple or other attributes. Bonds are formed in pairs; therefore, no bond can involve three or more atoms (except hydrogen(i) shared by a group of skeletal atoms). A molecule is without co-ordinates. However, the identifier represents the configuration of the stereogenic elements, as captured by the amplified structural source data with 2-D or 3-D coordinates. The most important aspect of InChI is its hierarchical nature and in particular it is linked to the concept of core parent structure, a common archetype for source structures and other related structures such as tautomers and stereoisomers. In fact, the descriptor provides a layer organisation that allows feature blocks to be added. Each layer is a sequence of characters starting with "/" followed by a letter denoting the identity of the layer (e.g. the molecular skeleton connection layer is preceded by "/*c*", the hydrogen layer by "/*h*", the charge layer is preceded by "/*q*" and the protonation/deprotonation layer by "/*p*", etc.). See figure 2.4 for an example of InChI annotation.

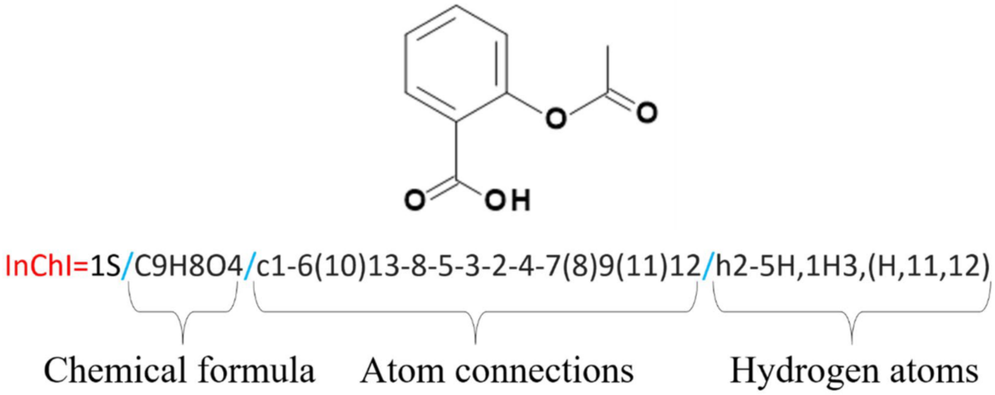
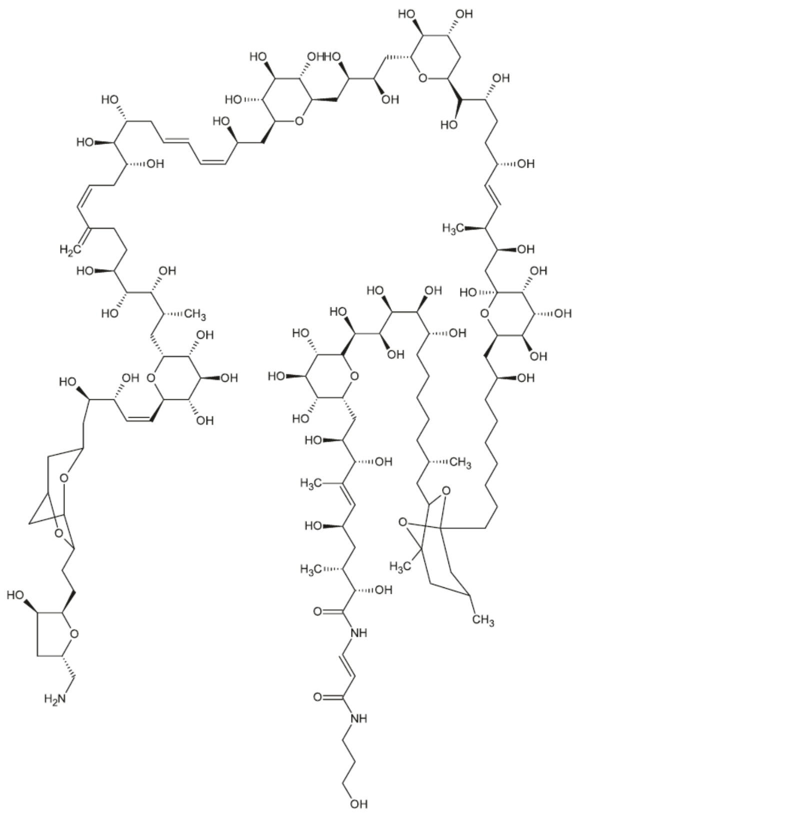

O
OH
O
O
InChI=1S/C9H8O4/c1-6(10)13-8-5-3-2-4-7(8)9(11)12/h2-5H,1H3,(H,11,12)
Chemical formula
Atom connections
Hydrogen atoms

Figure 2.4: InChI notation of aspirin. Red letters are the standard beginning of the notation. The following 1 corresponds to the InChI version number, and S states that the notation is a standard InChI. Slashes (blue) are delimiters. Image taken from [4].

The presence of layers, although convenient from a descriptive point of view, creates very long strings and above all of variable length, which make their management in databases and various computer applications complex. For this reason, an encoded form of InChI has been developed, called *InChIKey*, which has a fixed length of 27 characters, making internet searches and database indexing easier (an example of a molecular representation with InChI and InChIKey is shown in the figure 2.5).

The encoding with which InChIKey is generated is based on a hashing algorithm that is typically used in computer science to generate a fixed-length string from a larger source string, creating a highly compacted version of the string, which makes it inevitable to obtain the same hash code for two different inputs (collision), causing the uniqueness of the identifier to be lost. InChIKey has a hierarchical structure like that of InChI. The first block of 14 characters encodes the constituent core of the molecule as described by the sub-layers of formula, connectivity, hydrogen position and charge in InChI and is very often the same for the same molecular structures. All other chemical characteristics such as isotopic substitution, changes in stereoconfiguration, tautomeric state and bond coordination are described in the second character block. An exam-



IUPAC Name:
(E,2S,3R,5R,8R,9S)-10-[(2R,3R,4R,5S,6R)-6-[(1S,2R,3S,4S,5R,11S)-12-[(1R,3S,5S,7R)-5-[(8S)-9-[(2R,3R,4R,5R,6S)-6-[(E,2S,3S,6S,9R,10R)-10-[(2S,4R,5S,6R)-6-[(2R,3R)-4-[(2R,3S,4R,5R,6S)-6-[(2S,3Z,5E,8R,9S,10R,12Z,17S,18R,19R,20R)-21-[(2R,3R,4R,5S,6R)-6-[(Z,3R,4R)-5-[(1R,3R,5S,6R)-6-[2-[(2R,3R,5S)-5-(aminomethyl)-3-hydroxyoxolan-2-yl]ethyl]-4,7-dioxabicyclo[3.2.1]octan-3-yl]-3,4-dihydroxypent-1-enyl]-3,4,5-trihydroxyoxan-2-yl]-2,8,9,10,17,18,19-heptahydroxy-20-methyl-14-methylidenehenicosa-3,5,12-trienyl]-3,4,5-trihydroxyoxan-2-yl]-2,3-dihydroxybutyl]-4,5-dihydroxyoxan-2-yl]-2,6,9,10-tetrahydroxy-3-methyldec-4-enyl]-3,4,5,6-tetrahydroxyoxan-2-yl]-8-hydroxynonyl]-1,3-dimethyl-6,8-dioxabicyclo[3.2.1]octan-7-yl]-1,2,3,4,5-pentahydroxy-11-methyldodecyl]-3,4,5-trihydroxyoxan-2-yl]-2,5,8,9-tetrahydroxy-N-[(E)-3-(3-hydroxypropylamino)-3-oxoprop-1-enyl]-3,7-dimethyldec-6-enamide
InChIKey: CWODDUGJZSCNGB-HQNRRURTSA-N

Figure 2.5: Structure, IUPAC name and InChIKey for palytoxin

ple of an InChIKey obtained from the InChI source for the cafferine molecule is shown in figure 2.6.

This chemical representation, however detailed it may appear, especially in its extended version, has many shortcomings, both structural and applicative. Indeed, InChI only distinguishes certain types of stereochemistry (e.g. cis/trans-platinum structures have the same InChI), it does not handle mixtures such as positional isomers or polymers with variable bonds very well. Moreover, it is not a file format, which is why the conversion of structures into this descriptor may sometimes not take place as desired. In spite of these shortcomings, they are still a young chemical descriptor that has proved to be, thanks to the features described above, a very useful tool for linking different chemical information, even if nowadays they are not a good embedding for the Virtual screening and Drug Repurposing phases.

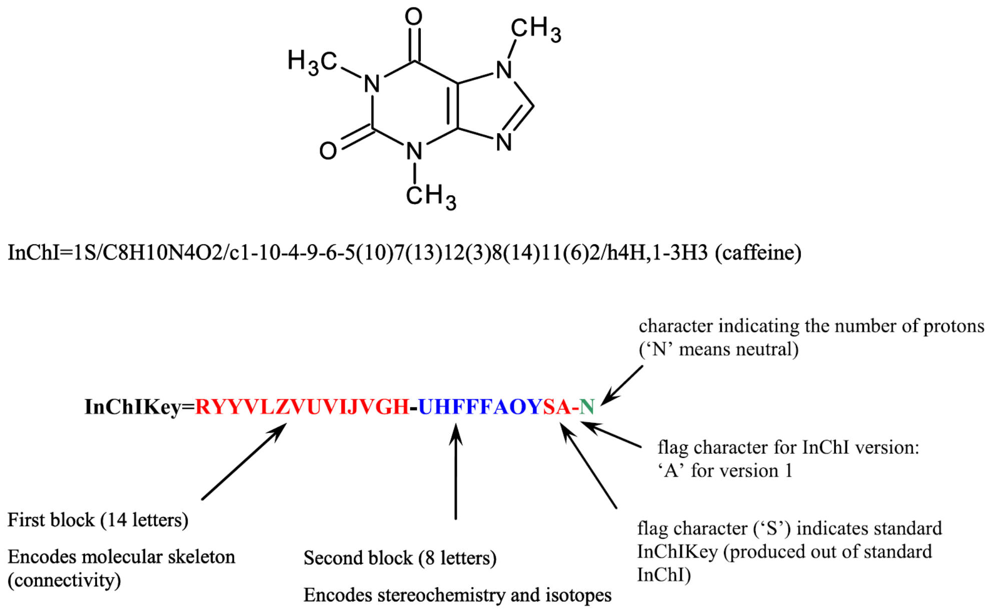

InChI=1S/C8H10N4O2/c1-10-4-9-6-5(10)7(13)12(3)8(14)11(6)2/h4H,1-3H3 (caffeine)
character indicating the number of protons ('N' means neutral)
InChIKey=RYYVLZVUVIJVGH-UHFFFAOYSA-N
flag character for InChI version: 'A' for version 1
First block (14 letters)
Encodes molecular skeleton (connectivity)
Second block (8 letters)
Encodes stereochemistry and isotopes
flag character ('S') indicates standard InChIKey (produced out of standard InChI)

Figure 2.6: InChIKey layout explained (using caffeine as an example)

2.5 Molecular Fingerprint

The last descriptor that will be discussed in this section is the Molecular Fingerprint, which is the most widely used state-of-the-art molecular descriptor for VS. This embedding stems from the evolution of Structural Keys (SK) [4], the first descriptor created for high throughput screening on large data sets. Structural keys are typically represented as Boolean arrays, where each element is *TRUE* or *FALSE* and represents the presence or absence of a chemical structure (pattern), respectively. SKs vary widely in size, ranging from a few tens to thousands of bits, which means that the larger they are, the greater the possibility of detecting a chemical substructure. Unfortunately, over the years, it has been observed that the Structural keys, due to the search for predefined patterns for the creation of the Boolean array, suffered from a lack of generality which made them not very versatile in the VS [84]. Molecular fingerprints were developed to overcome this limitation by eliminating the idea of a predefined pattern assigned to each bit while remaining a Boolean array or bitmap. The Molecular Fingerprints bitmap is created from the molecular graph from which pattern information is extracted and converted into bits using various kernels and algorithms. Since the number of patterns, including the type of atom, bond or the possible presence of aromatic rings, is very large, a bit is not assigned to each pattern, as is the case with structural keys, but hashing, is used to create an encoded version. After enumerating all of the patterns in the molecule, each one is used as a seed for a hashing function that outputs 4 to 5 index positions with their corresponding bits in the "pattern fingerprint" set to 1; this fingerprint is bit-wise OR-end to the molecular one. Because the hashing function can cause a bit collision, we can't be sure a pattern exists unless at least one of its bits is unique. A molecular substructure, on the other hand, is missing if all of its fingerprint bits are set to 0. An example of fingerprint generation is shown in figure 2.7.

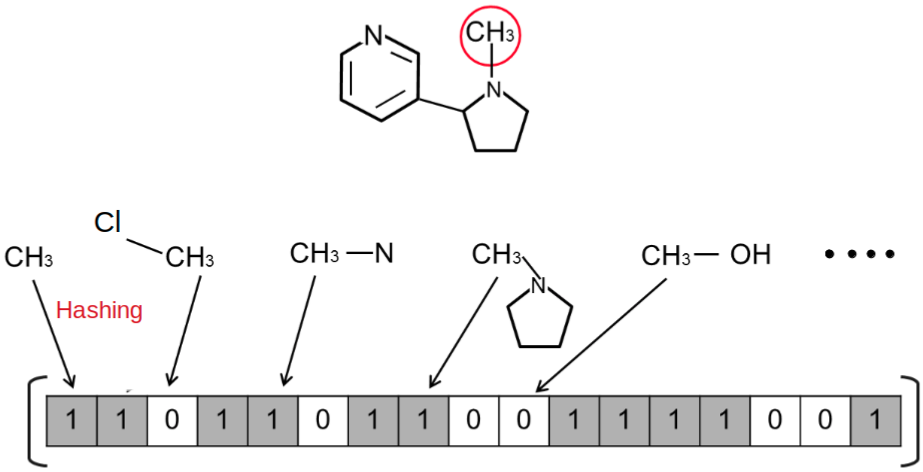


Figure 2.7: Simplified fingerprint generation: the hashing function sets just 1 bit per pattern.

Although the difference between the meaning of the bits belonging to the Fingerprint and those belonging to the structural key is evident, they share an important characteristic, in fact in both descriptors if a bit is set to 1 it indicates the presence of a certain pattern within the molecular structure. For this reason, both lend themselves to the search for substructures and the study of similarities between two molecules through the use of simple Boolean operations [84]. The study of similarity, in fact, is a widely used task even outside the field of Virtual Screening and Drug Discovery and can be calculated in different ways, the complexity of which varies according to the molecular descriptor used. In fact, the advent of Molecular Fingerprints has made it possible to optimise research methodologies by reducing computational difficulties thanks to the condensed encoding of information typical of this representation. The methodologies used for the study of similarity based on the Fingerprints are numerous and many of these are developed to quantify in a range of values the similarity between two bit vectors. In the chemoinformatics community the standard is represented by the Tanimoto Coefficient [92, 93, 94] which is calculated according to the equation

$$S_A B = \frac{c}{(a+b+c)}$$

Where c is the number of equal bits 1 for both vectors, where a is the number of bits 1 in vector A and b is the number of bits 1 in vector B. The Tanimoto coefficient is only one of the similarity metrics used in the world of chemoinformatics, other famous coefficients are given in table 2.1

The use of fingerprints in similarity studies has revealed to the chemoinformatics community their enormous potential to contain structural information in a sparse vector. This characteristic makes them the most widely used molecular descriptor in the domain of Virtual screening, especially in LBVS which is mainly based on the study of structural similarity between active molecules as described in detail in the section 1.1.1. Their increasing use over the years has allowed their development, reaching a great variety in the state of the art, both in terms of length and complexity of the chemical information they are able to carry. The most popular molecular fingerprints can be grouped into the following classes:

1. Pattern Fingerprints (e.g. RDKit, Layered [82])
2. Topological Fingeprints (e.g. Daylight [84], AtomPair[95], Torsion [96]
3. Circular Fingeprints (e.g. Morgan [97], FCFP, ECFP[97], FCFP)
4. Structural Keys (e.g. MACCS [98], BCI [99])
5. Pharmacophore fingerprints (e.g., CAT descriptors [100], 3D fingerprints)

Each of these fingerprints is a bitmap, the structure of which varies in length and complexity. Each of these fingerprints is generated using different kernels, so they contain different chemical information for having the same structure. Pattern and topological fingeprints are generated by analysing all the fragments of the molecule that follow a path (usually linear) for a number of bonds, and then hashing each of these paths to create the fingerprint as shown in the figure 2.7. This means that any molecule can produce a meaningful fingerprint, and its length can be adjusted. These are hashed fingerprints, which means that a single bit cannot be traced back to a given feature, although as described above, because the number of bits is limited, a bit collision can

Table 2.1: Summary of the most used similarity metrics.

Measure	Formula*	Range
Cosine	$\frac{c}{\sqrt{(a+b)*(b+c)}}$	$0.0, 1.0$
Dice	$\frac{2.0*c}{((a+c)+(b+c))}$	$0.0, 1.0$
Euclid	$\sqrt{\frac{c+d}{a+b+c+d}}$	$0.0, 1.0$
Forbes	$\frac{c*(a+b+c+d}{((a+c)*(b+c)}$	$0.0,$
Hamman	$\frac{(c+d)-(a+b)}{(a+b+c+d)}$	$-1.0, 1.0$
Jaccard	$\frac{c}{(a+b+c)}$	$0.0, 1.0$
Kulczynski	$0.5*(\frac{c}{a+c}+\frac{c}{b+c})$	$0.0, 1.0$
Manhattan	$\frac{(a+b}{a+b+c+d}$	$1.0, 0.0$
Matching	$\frac{c+d}{a+b+c+d}$	$0.0, 1.0$
Pearson	$\frac{(c*d)-(a*b)}{\sqrt{(a+c)*(b+c)*(a+d)*(b+d)}}$	$-1.0, 1.0$
Rogers-Tanimoto	$\frac{c+d}{(a+b)+(a+b+c+d)}$	$0.0, 1.0$
Russell-Rao	$\frac{c}{(a+b+c+d)}$	$0.0, 1.0$
Simpson	$\frac{c}{min((a+c),(b+c))}$	$0.0, 1.0$
Yule	$\frac{(c*d)-(a*b)}{(c*d)+(a*b)}$	$0.0, 1.0$

occur causing an overlap of the chemical information that bit 1 contains. The main difference between these two classes of fingerprints lies in the pattern and chemical information that is identified as 1. RDKit fingerprints have been designed to be used in substructure screening. The algorithm that generates them specifically searches for substructures, numbering them with a relatively small number identifier, in order to facilitate the search. These substructures are then hashed using atomic number for the atom, bond type and aromaticity for aromatic bonds. Layered fingerprints are another type of RDKit fingerprints and are based on a generation algorithm very similar to the one described above. They are also fingerprints designed to search for substructures, but unlike RDKit fingerprints, they organise the chemical information into layers that will then be hashed. Once a substructure has been found, it can be inserted into one of the following layers:

- Pure topology
- Bond order
- Atom types (atomic number)
- Presence of rings
- Ring sizes
- Aromaticity

Daylight fingerprints: This is the best known of the topological fingerprints. It consists of up to 2048 bits and encodes all possible connectivity paths through a molecule up to a given length, minimising bit collision. A representation of a hypothetical 17-bit topological fingerprint is shown in the figure 2.7. All fragments found from the starting atom (circled in red) are shown and the corresponding bit in the fingerprint is indicated. Only fragments and bits for a single starting atom are shown; for the complete fingerprint, this process would be performed for every atom in the molecule.

Circular fingerprints are generated using a similar approach, but constructing fragments within a radius of the starting atom instead of linear fragments. This type of fingerprint overcomes the problem of isomorphic molecules, i.e. when two molecules have different atom numbering but the same structure. The algorithm is based on a circular kernel that iterates the operation by varying the radius and increasing the number of patterns detected (An example of the application of the kernel just mentioned is found in figure 2.8).

The main algorithm for this class of fingerprints is Morgan's, which is based on an iterative process in which numerical identifiers are assigned to each atom, initially using a rule that encodes an invariant numbering to each atom starting

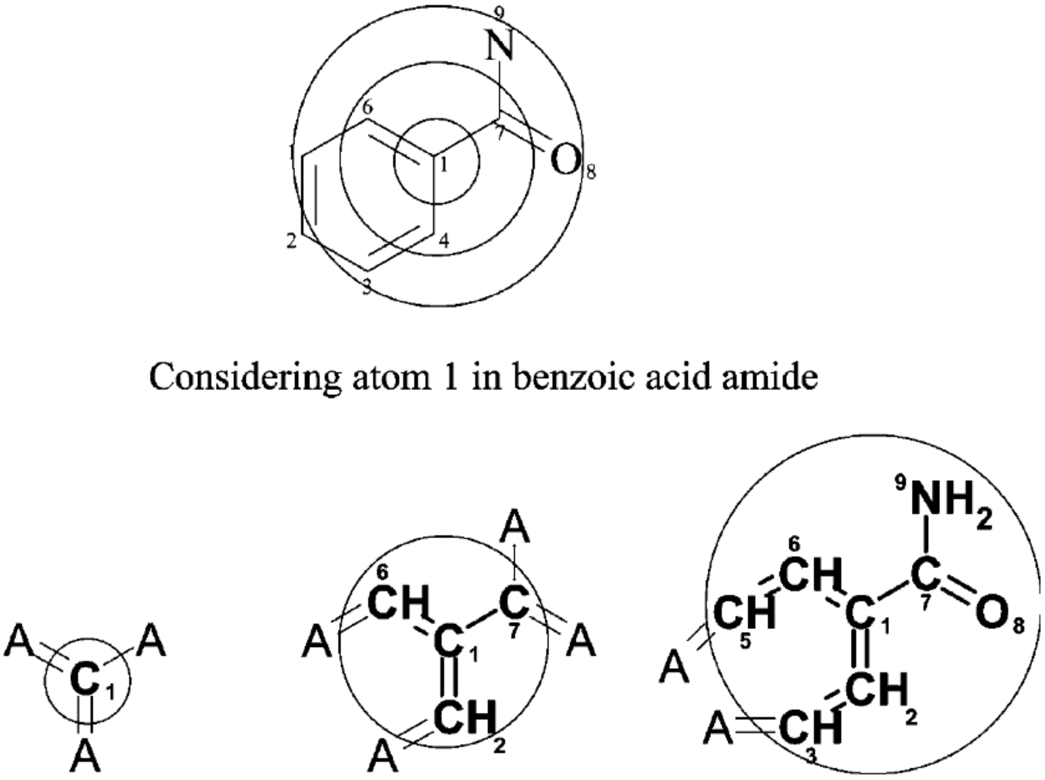
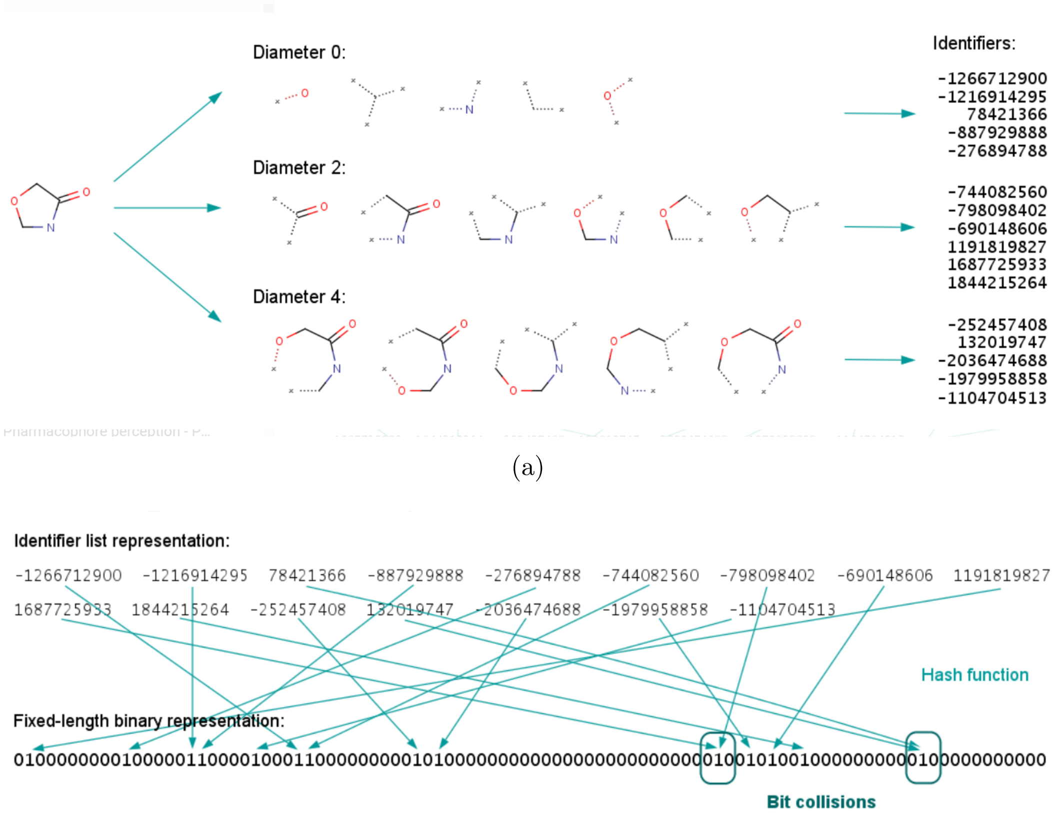


Figure 2.8: An example of the application of the circular kernel.

with the atom identifier at the centre of the kernel, and then proceeding to the next step using the identifiers from the previous iteration. Thus, the identifiers generated are independent of the original numbering of the atoms. The iterative process continues until the numerical identifier of each atom is unique. To safeguard unambiguity, the standard Morgan process is carefully recoded after each iteration to avoid mathematical overflows and possible bit collisions [97].

The algorithm for generating ECFPs is very similar to the Morgan algorithm, but applies several variations that make them different in terms of the chemical information they contain. In fact, ECFPs terminate after a predetermined number of iterations regardless of whether a unique identifier is obtained or not. The initial identifier, and all other numerical identifiers representing the other atoms in the chemical environment, are collected within a set, which will be referred to as an extended-connectivity fingerprint. ECFPs follow a much faster hashing scheme than the strict addition of bits used in Morgan, decreasing the computational cost necessary for their generation [97]. An example of ECFP generation is shown in figure 2.9.

FCFPs (Functional-Class Fingerprints) are a variation of ECFPs, which are



(a)

(b)

Figure 2.9: ECFP generation process. Generation of the fixed-length bit string ("folding").

further abstracted in that instead of indexing a particular atom in the environment, they index the role of that atom. Thus, several atoms or groups with the same or similar function are not distinguished by fingeprints. This also allows them to be used as pharmacophoric fingerprints. FeatMorgan is an example of FCFP and is the most widely used type in this class. Molecular Fingeprints are a very powerful tool available to the chemoinformatics community, especially for researchers working in the VS and similarity search domain. At present, there are many studies that use them as descriptors in these application fields. Modern approaches in chemoinformatics have focused on the use of ML and DL techniques applied to Fingeprints instead of classical molecular descriptors. The reason is that the latter contain information on chemical groups and pathways; they provide comprehensive information on molecular complexity, thus allowing a more robust comparison between two or more structures than molecular descriptors.

There are several state-of-the-art works using fingeprints, e.g. convolutional neural networks have been used to learn fingerprints directly from two-dimensional graphs of the molecule [67]. In the work of Duvenaud et al. a single convolutional layer with softmax activation is used instead of a hashing function to produce bit indexing of a neighbourhood of atoms collected in the same way as circular fingerprints. The fingerprints created were used to make classification and the authors report excellent performance in predicting both solubility and toxicity from two specially defined datasets. The approach was found to be very innovative, although it suffers from a high computational cost when compared to fingerprints generated using hashing functions. Yang et al. used an Extreme Gradient Boosting (XGBoost) approach to identify JAK2 kinase inhibitors using fingeprints as input data [101]. Zhong et al in 2020 combined a DNN with Molecular fingeprints to predict the degree of OH^+ radicals of 593 contaminants [102]. In the paper of Zagidullin et al. [103] 11 different fingerprints are compared in order to predict the sensitivity of drug combinations and synergy scores, evaluating their relationship through the use of clustering approaches and fingerprints similarity studies based on CKA (Centered Kernel Alignment). In the paper by Abbasi et al. [104], molecular Fingerprints or combinations thereof are used in combination with primary protein sequences to predict Drug-Target interaction.

As can be guessed, their main strength lies in their ability to store chemical information within a highly condensed structure. Their generation algorithm is very fast, despite the different kernels of chemical structure analysis, thanks to the use of hashing algorithms. The typical size of hashed fingerprints ranges from $1K$ to $4K$ bits, where 1024 is the most used size despite having the highest incidence of bit collision. These characteristics added together have motivated me to deepen the capacities, the potentialities and the deficiencies of the Molecular Fingeprints in the LBVS through the use of algorithms of ML and DL. These approaches will be discussed in detail in the next sections.

Chapter 3

Target selection

My research activity was based on the study of molecular descriptors, in particular Molecular Fingeprints and the descriptor proposed in this thesis, *NMR-Like*, in two distinct phases of Drug Discovery: Virtual screening and Drug Repurposing, following in both cases a Ligand-Based approach that, as already described in the section 1.1.1, does not require knowledge on the structural sequence of the target, but focuses on the similarity of structure shared by the ligands.

In a VS study, the selection of the target is a crucial step for the proper progress of the research. Indeed, it should not be forgotten that these preliminary stages aim to identify a molecule that has the potential to become a drug, which will be used to treat a pathology. The know-how of the researcher in the pharmaceutical field also includes knowledge of the biology underlying the processes that manifest the pathological phenotype, making it possible to select an appropriate target for chemoinformatics research.

In my research activity, the target selected was a protein from the Cycline-Dependent Kinases (CDK) family, namely $CDK1$, a protein directly involved in the regulation of the cell cycle. The correct functioning of this biological process is essential for the maintenance of regular tissue organisation, allowing cells that have ended their life to be replaced by healthy, differentiated cells, by means of mitosis from a mother cell. Failure of the delicate interlocking of biological interactions, regulated by the CDK family in cooperation with many other co-protagonists, leads to a cancer phenotype.

Cancer has been described by many scientists as the disease of the century, and occurs when cells begin to proliferate uncontrollably and their number exceeds the amount physiologically required. The disease is not hereditary, but is a disease with a strong genetic component can be affected by one or more genetic mutations that change the structure of the proteins involved [105, 106].

The proteins that are mutated can be multiple and often many of them are regulators or effectors that interact directly with CDK family proteins, so intervening in their targeted regulation can prevent the uncontrolled evolution of the cancer phenotype.

Nowadays, one of the major limitations encountered is the lack of good labeled datasets to use as benchmarks for descriptor testing. In the following sections, the target family is described in detail and most importantly, the pipelines that led me to the creation of the labeled datasets used in the different experimental phases are exposed.

3.1 CDK family

Members of the Cycline-Dependent Kinases (CDK) family were originally characterised by all serine/threonine-specific kinase proteins activated by interaction with cyclins in order to regulate the cell cycle of eukaryotic cells. With the advent of the CMGC (**C**yclin-dependent kinase [CDK], **M**itogen-activated protein kinase [$MAPK$], **G**lycogen synthase kinase [$GSK3$], **C**DC-like kinase [CLK]) division of the kinome [107] only 20 proteins are now considered to be part of the CDK family, and can be grouped in turn into various subsections. The first branch of kinases is represented by the kinases 'CDKs 1, 2, 4 and 6' which are directly involved in cell cycle regulation. A second branch is represented by 'CDKs 7, 8, 9, 12 and 13', which regulate transcriptional events through the phosphorylation of heptad repeat residues in the C-terminal tail of RNA polymerase II (CTD). CDK7 is an unusual protein because it indirectly regulates the cell cycle by mediating the activity of CDKs 1, 2, 4 and 6, while $CDK3$ has the Retinoblastoma protein (pRB) as a substrate to promote the end of quiescence and the transition from G0 to G1 phase in the cell cycle. Other proteins belonging to the CDK family, such as CDK5, 10, 11, 14-18 and 20 have various and tissue-specific functions. For example, $CDK5$ plays a crucial role in regulating microtubule activity during the developmental stages of the neuron and is one of the few kinases that does not require the presence of cyclins for its activation.

As the name of the family suggests, the kinases belonging to this class are bound to the presence of a cofactor, the cyclin, in order to be activated. A representation of the CDK and $CDK - Like$ ($CDKL$) family of proteins is shown in figure 3.1.

CDK1 is a protein that plays a key role in maintaining the wild-type pheno-

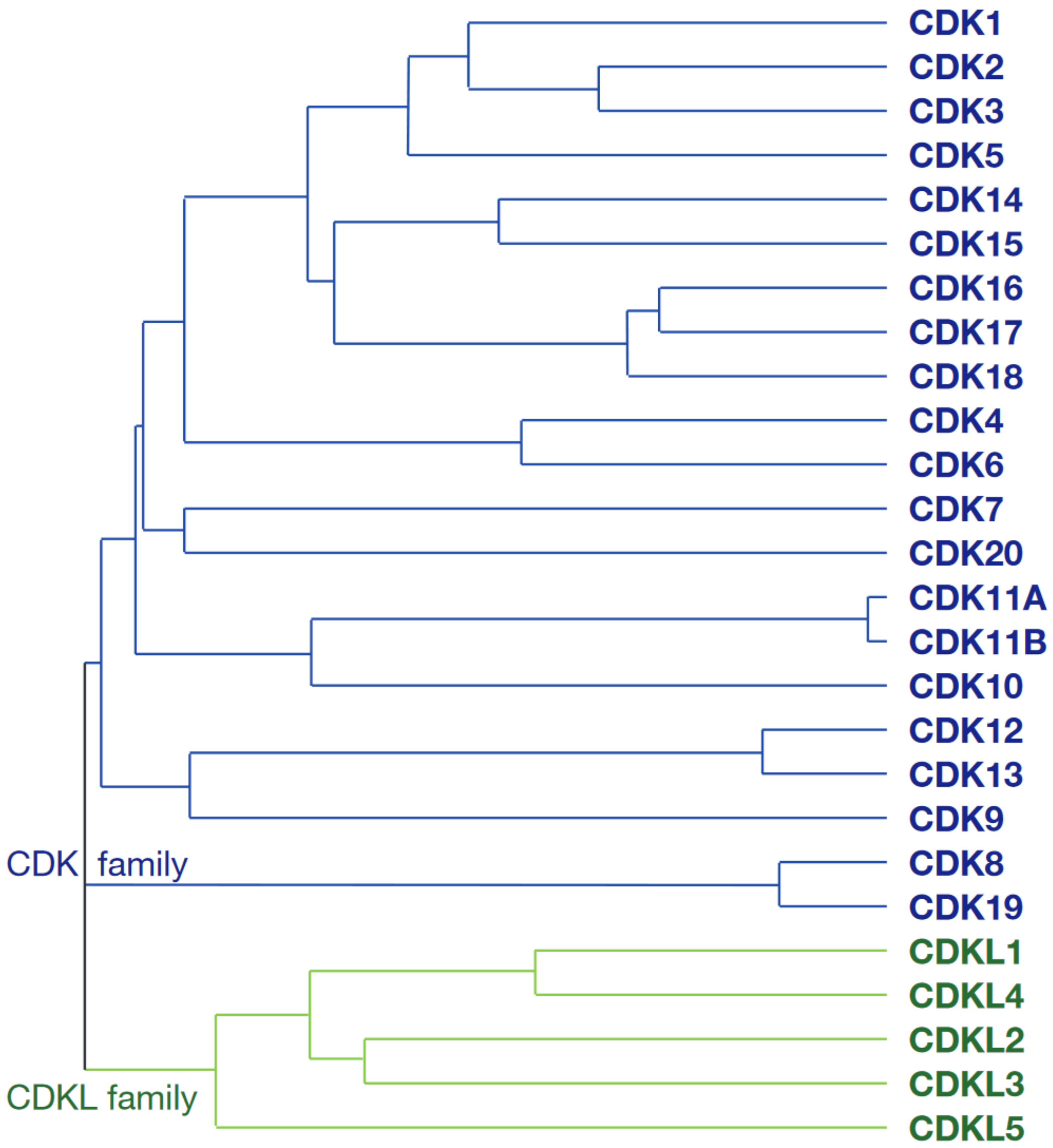


Figure 3.1: Human CDK e CDKL (CDK-Like) proteins [5].

type, because it is involved in the regulation of the G2 phase and the M [108] phase (as can be seen in figure 3.2) the mitotic phase of the cycle, i.e. the phase in which division signals are sent to the mother cell leading to the formation of the two daughter cells. Deficiencies in CDK1 activity have been associated with the onset of several forms of cancer in humans [109, 110, 111, 112], and it is therefore clear why I chose this protein as the target for my research activity.

It is important to remember that although the role of CDK1 is crucial for the G2/M transition [113], its inhibition is not able to completely terminate the onset of cancer, because tumorigenesis is a multifactorial process involving several proteins.

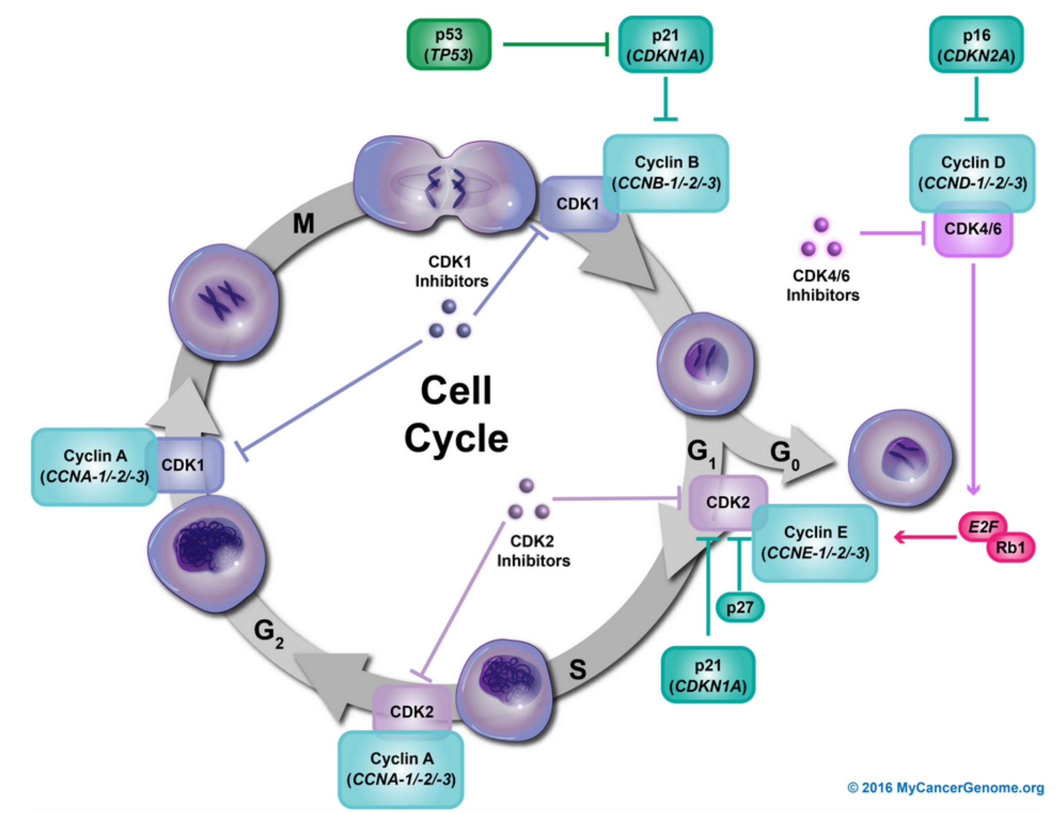


Figure 3.2: A diagram of the cell cycle. It is characterised by 4 different phases: G1 (gap phase 1), S (DNA synthesis), G2 (gap phase 2), and M (mitosis). CDK1 regulates the transition from G2 to M phase. Figure taken from My Cancer Genome [6].

3.2 Data preparation

The datasets used during my research activity were different and will be described in detail below. All datasets used to test the Molecular Fingerprint and the NMR-Like descriptor in LBVS applications, were created starting from CDK1 as a target, while for applications in the Drug Repurposing domain, a set of proteins involved in the infection of SARS-CoV-2 were selected.

Dataset 1. Single protein research. Phase I.

The following dataset was selected starting with the CDK1 protein. The well-known ChEMBL molecule database [80] was used to succeed in generating it. Specifically, two sets with known bioactivity on both the single protein, *CDK1*, from *CHEMBL308* ID and the *CDK1-Cyclin B1* protein complex from *CHEMBL1907602* ID were downloaded for a total of 1830 molecules. Incomplete data were removed, yielding a total of 1707 molecules that were partitioned into a training set consisting of 1432 molecules with a 1 : 1 ratio of active to inactive, equaling 716 molecules. The test set was created using the remaining molecules, for a total of 275 samples, 175 inactive and 100 active specifically.

The classification of the biological activity of the compounds was obtained through the use of the *half maximal inhibitory concentration parameter* (IC50), which is the amount of substance required to inhibit the target protein (in the specific case CDK1) by half [114]. The threshold used for labeling this dataset states that a molecule is considered active when its IC_{50} value $\leq 9\mu M$, alternatively it is considered inactive.

3.2.1 Dataset 2. Single protein research. Phase II.

This second dataset was generated to expand the number of molecules tested, especially considering the increase in performance that Machine Learning algorithms and in particular Deep Learning algorithms have when the number of data used increases. In order to optimize the performance of the networks, to obtain the most accurate information on the reliability of the descriptors, the dataset 3.2 was expanded. Specifically, to the 1707 compounds of the starting dataset, 13 compounds obtained from an update of the information present on ChEMBL for IDs *CHEMBL308* and *CHEMBL1907602* were added and then 2422 compounds active on TRKA (Tropomysion receptor kinase A, *CHEMBL2815*) were selected, 2825 compounds active on AKT1 (AKT Serine/Threonine Kinase 1, *CHEMBL4282*), 199 compounds active on LIMK1 (LIM Domani Kinase 1, *CHEMBL5932*) and 50 compounds active on RIPK1 (Receptor-Interacting Protein 1, *CHEMBL4282*) for a total of 5496 molecules that following removal of duplicates were reduced to a total of 5452. The molecules were divided into training, validation and test set according to a scheme 80% : 10% : 10% and with a ratio between active and inactive of 1 : 10 for each of them.

Labels for molecules active and inactive against CDK1 were defined using the IC_{50} value. In contrast to what was performed for the dataset 3.2, the threshold used was not set at $9\mu M$, but I redefined a new threshold based on the distribution of the data. In the literature, the IC50 values used are $IC_{50} \leq 1\mu M$ for strongly active molecules and $IC_{50} \geq 9 - 10\mu M$ for definitely inactive molecules. These two limits, however, cannot be considered universal and usable for all molecules, which is why in the preparation of this dataset, I took care to study the distribution of IC_{50} values for all selected molecules, using a clustering algorithm, the *K-Means* [115]. Initially, the method called *elbow method* was applied.This heuristics consists in clustering the data points $\boldsymbol{x}$ with a variable number of clusters k, while plotting the *Within-Cluster Sum of Squares*:

$$WCSS = \sum_{i=1}^{k} \sum_{\boldsymbol{x} \in C_i} (\boldsymbol{x} - \boldsymbol{\mu_i})^2 \tag{3.1}$$

where C_i is the i-th cluster, and μ_i is its centroid. The plot will exhibit an "elbow" in correspondence of the optimal value for k. In this way, we obtained $k = 2$ for each target as it was expected, and we were also able to evaluate the centroids, and the extent of each cluster. Analyzing the clustering results, we obtained the value $K_D = 7\mu M$ as a good threshold to separate data correctly for each target.

The plot of the elbow method is shown in figure 3.3.

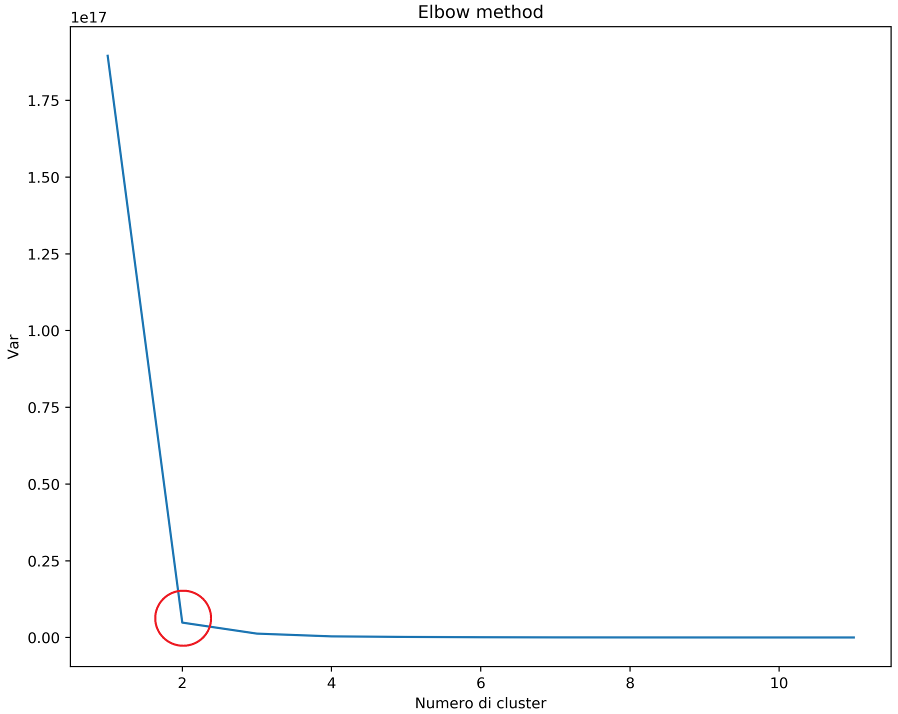


Figure 3.3: Graphical representation of the result obtained through the elbow method. Inside the red circle the number of clusters is highlighted.

In the next step, the K-Means algorithm was applied on the two clusters identified through the elbow method determining the two centroids in which the data were distributed and which were consistent with the values reported in the literature:

- Centroid 1 for active molecules at $IC_{50} = 0.91762\ \mu M$, upper bound at $IC_{50} = 0.971 \mu M$
- Centroid 2 for inactive molecules at $IC_{50} = 13.91221\ \mu M$, lower bound at $IC_{50} = 13.338 \mu M$

This approach allowed me to redefine a new threshold of activity equal to the average value of centroids with IC_{50} value $= 7.414\ \mu M$, so all molecules with $IC_{50} \leq 7.414\ \mu$ M were considered active. Also in this dataset, despite the increase in the number of molecules used the bioactivity refers to *CDK1*. The final dataset consisted of 7147 molecules (869 active and 6278 inactive).

3.2.2 Dataset 3. Multi-Class Multi-Target Search.

Deep Learning algorithms, and in particular deep neural networks have extended the ability to classify the bioactivity of a molecule to more than a single target at the same experimental stage, thus allowing the researcher to know if a molecule is active or inactive on more than one protein at the same time. It was precisely on the basis of this axiom that I based the creation of the third dataset to test the molecular descriptors. In collaboration with the group of computational chemistry of Fondazione Ri.MED, starting from *CDK1* as a protein target, I was involved in performing a study of similarity between binding site, ie the site of binding/interaction between the molecule and the protein that induces the conformational change through which biological activity is manifested. Specifically, I made use of the *Interaction Fingerprint Pattern (IFP)* [116], a type of fingerprint that contains chemical information inherent to the patterns involved in the interaction between the molecule and the binding site. This type of Molecular Fingerprints is used to perform similarity studies between the binding sites of proteins, which, according to the amino acid residues of which they are composed, can bind only molecules with a specific chemical structure. The similarity coefficient used to identify these proteins was the Tanimoto coefficient with a threshold ≥ 0.80, which allowed me to identify a total of 20 other target proteins.

For each of the selected targets, all molecules with known bioactivity were downloaded from *ChEMBL*, not using IC_{50} as the only parameter, but also using the inhibition constant K_i and the dissociation constant K_d. K_d measures the equilibrium between the ligand-protein complex and the dissociated components as shown in the chemical equation below

$$\mathrm{PL} \underset{K_\mathrm{d}}{\rightleftharpoons} \mathrm{P} + \mathrm{L}$$

$$K_\mathrm{d} = \frac{[P][L]}{[PL]} = \frac{k_{-1}}{k_1}$$

Where $[P]$ is the free protein concentration, $[L]$ is the free ligand concentration, $[PL]$ is the protein-ligand complex, k_{-1} is the dissociation rate constant for the complex and k_1 is the association rate constant. The K_i inhibition constant also represents a dissociation constant, but more narrowly for the binding of an inhibitor to an enzyme. That is, a ligand whose binding reduces the catalytic activity of the enzyme. The binding equilibrium described by the K_i value depends on the kinetic mechanism of inhibition.

As described in the previous section, the IC_{50} threshold for defining a strongly active molecule is $\leq 1\mu M$, a threshold that also applies canonically for K_i. K_d, in contrast, does not have a defined threshold in the literature and it was necessary to use the *K-Means* cluster again to be able to define a threshold that fit the data. Clustering was calculated on the individual data sets for each of the proteins, in order to first identify the ideal number of clusters for each set using the elbow method, allowing me to calculate the WCSS (see equation 3.1) which returned a value of $k = 2$ common to all proteins. Following evaluation of the 40 centroids identified (2 for each protein) a threshold of $K_d = 7\mu M$ was identified.

Once the thresholds for active and inactive classes were defined, the sets downloaded from *ChEMBL* were labeled. The distribution between active and inactive molecules was unbalanced towards the active molecules, therefore a further phase of data processing was conducted. This phase has been developed following two different approaches, creating workflows with *KNIME Analysis Platform* [117]. The first method, exploits IFP to identify a new set of proteins, with a binding site dissimilar to that of CDK1. This assumption is based on the fact that being the protein-ligand interaction very specific all molecules active on proteins with a binding site different from that of the target, are inactive on the latter. To perform this analysis was used the node "3D-e-Chem - KLIFS" (KLIFS - release version 2.4, developed by the Pharmaceutical Chemistry Division - VU University Amsterdam) that returns information on all human kinoma from the "Kinase-Ligand Interaction Fingerprints and Structures" database [118] that in fact contains all the information regarding the site of interaction of protein kinases and the catalytic domain deposited in the

Protein Data Bank [22] . All kinases were used as input to generate the IFPs, used the similarity search that specifically was performed using the "Similarity Search" node of KNIME, in order to select proteins with a Tanimoto coefficient in the range $[0 - 0.15]$ and therefore very dissimilar from the selected targets. These proteins were used to search for new inactive on ChEMBL, selecting all proteins with $IC_{50} \leq 1/muM$. Despite this first approach, the number of inactive was still unbalanced towards the active so with the group of foundation Ri.MED we thought it was necessary to deepen the research following a different approach, no longer related to dissimilarities between proteins, but to dissimilarities between ligands strongly active on the selected proteins.

The ligands were downloaded in SDF format (Ctab-based data structure that is used to deposit information in databases) from the Protein Data Bank (PDB) [22] (see ligands code in the table 3.1). Once the crystallized structures of these small molecules were obtained 601810 molecules were downloaded from ChEMBL DB v26 and used for a dissimilarity study using the following values as thresholds:

- molecular weight > 100
- number of carbon atoms > 10
- number of nitrogen atoms > 2
- number of oxygen atoms > 2
- at least one aromatic ring

The similarity study was conducted using ECFP4 fingeprints, a type of fingerprints suitable for comparison between small molecules, always using the Tanimoto coefficient in the range [0-0.1] as an evaluation parameter. By using these different approaches to select molecules, we minimized the possibility of analogues bias and artificial enrichment, typical of using uncurated datasets and Decoys [119, 120], artificially generated compounds for which the bioactivity is unknown, but which are arbitrarily labeled as inactive by the researcher using them. The general analyses allowed us to obtain a very large number of compounds for each target protein and a summary is shown in the table 3.1.

As can be seen from the table 3.1, the number of inactives for each kinase is very high compared to the number of actives. In order to obtain a balanced dataset with a ratio of 1:100 between active (of the less abundant class, CDK6) and inactive, a combined search between the various datasets was performed

Table 3.1: A summary of all proteins (active and inactive) obtained from pre-processing methods.

Target	PDB ID	Ligand Code*	Actives	Inactives
ACK	5ZXB	9KO	746	159775
ALK	6E0R	HKJ	1665	227247
CDK1	6GU2	F9Z	1241	124473
CDK2	6INL	AJR	1924	225087
CDK6	5L2S	6ZV	646	256561
INSR	5E1S	5JA	1423	195990
ITK	4RFM	3P6	1001	135007
JAK2	6M9H	J9D	5526	577409
JNK3	2B1P	AIZ	658	95252
MELK	6GVX	TAK	1215	246662
CHK1	6FC8	D4Q	2175	21763
CK2a1	6JWA	5ID	1053	10534
CLK2	6FYL	3NG	671	6800
DYRK1A	4YLK	4E2	1126	11274
EGFR	5GNK	80U	4757	47541
ERK2	6OPH	6QB	3525	35237
GSK3B	5F94	3UO	2578	25768
IRAK4	6EG9	OLI	2131	21282
MAPK2K1	4AN9	ACP; 2P7	1254	12508
PDK1	3NAX	MP7	1117	11166

* Most affine lingands

to rank the inactive that were more common among the proteins, creating a ranking of presence. Once the ranking of inactives was obtained, the 64600 collectively most present were selected. The final dataset consists of 89373 molecules that were separated with the classical 80% : 10% : 10% ratio into Training (68370 molecules), Test (13046 molecules) and Validation set (7597 molecules).

A summary pipeline of the steps followed for the creation of each dataset is shown in the figure 3.4.

3.2.3 Dataset 4. Drug Repurposing Research

The dataset described in this section is the result of the activity carried out within the Task Force CLAIRE (*Confederation of Laboratories for Artificial Intelligence in Europe*) against COVID-19, Topic Bioinformatics (protein and molecular data analysis) to which the Human-Computer Interaction Laboratory has joined on a voluntary basis, and was also conducted in collaboration with researchers of Fondazione Ri.MED.

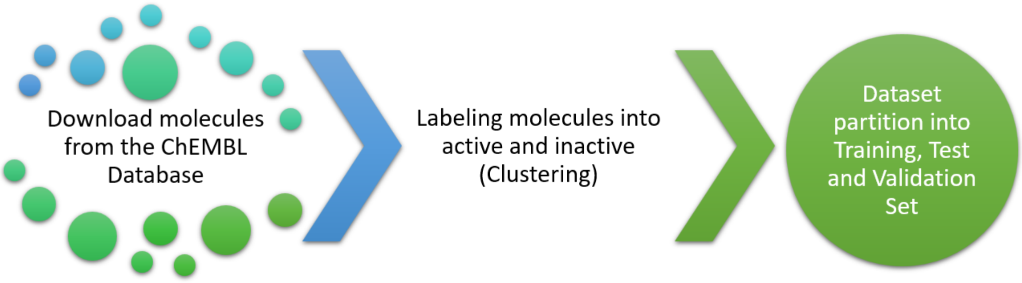

Download molecules from the ChEMBL Database
Labeling molecules into active and inactive (Clustering)
Dataset partition into Training, Test and Validation Set

Figure 3.4: Pipeline used for the generation of datasets 1, 2 and 3.

The first phases of the study were based on the creation of a dataset composed of viral and Host (human) proteins, directly involved in the infection or with similar sequence, and drugs, approved or in advanced trials, active or inactive with respect to these proteins. These first steps were crucial, since, given the topical nature of the research, there are no data regarding drugs that act directly on proteins implicated in the infection of COVID-19. The first phase of the study was based on a query, through the use of BLAST (Basic Local Alignment Search Tool) [121] the tool provided by National Center of Biotechnology Information (NCBI), to search for sequence similarity between a set of 41 proteins including host (human) and viral (SARS-CoV-2) proteins selected in collaboration with the group of Fondazione Ri.MED. The algorithm used is *blastp* and the search was performed on the database "nonredundant protein sequence (nr)" for the organisms: i) human (taxid:9606) and ii) virus (taxid:10239) excluding SARS-CoV-2 (taxid:2697049). The results of the alignment queries were filtered using the *Expect value* (E value) [122], a parameter that describes the number of random hits that can be detected when searching for hits within sequences of varying length. All alignments that had an E value < 0.01 were selected and sorted according to the percentage of Identity (see equation 3.2), allowing for two lists with the highest identity proteins with Host High Identity Protein (HHIP) and Virus High Identity Protein (VHIP).

$$Identity = \frac{\text{number of identical Aminoacids}}{\text{shortest sequence length}} \tag{3.2}$$

BLAST query results had different IDs (e.g., GeneBank, RefSeq, PBD, etc.)

Table 3.2: Cross-reference DrugBank database with the two high identity protein lists.

Drugs	HHIP	VHIP
Approved	5	0
Sperimenta	7	12

Table 3.3: DrugBank database cross-reference with interacting proteins.

	Approved	Sperimental
Drugs	85	128

reasoning that the NCBI Retrieve/ID mapping tool was used to obtain UniprotKB IDs. The filtered data were cross-referenced with approved and late-stage drugs in the DrugBank database (Wishart et al 2018) in order to identify compounds with proven activity on selected proteins. The results of this initial cross-reference are shown in table 3.2

As can be seen from the table 3.2, the number of active drugs is not sufficient to be able to efficiently train a neural network, reason for which a further cross-reference was performed, using data provided by the CLAIRE Task Force. Specifically, all proteins that interact directly with proteins on the HHIP list, i.e. Host proteins, were identified. This allowed me to expand the number of proteins that upon re-intersection with the DrugBank database returned a higher number of active drugs, as shown in Table 3.3.

The identified molecules were labeled as active. The final dataset included 1153 molecules a ratio of 1:4 between active and inactive, which were extrapolated by selecting drugs that act on strongly dissimilar proteins not involved in the biological mechanisms of SARS-CoV-2 infection.

A summary pipeline of the steps followed for the creation of dataset 3.2.3 is shown in the figure 3.5.

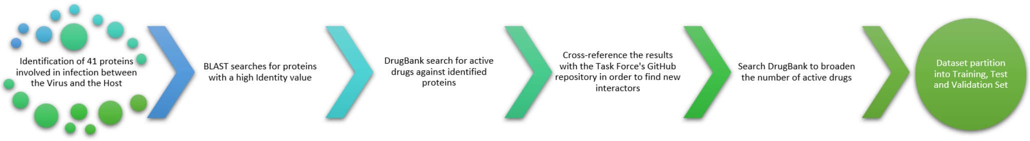

Identification of 41 proteins involved in infection between the Virus and the Host
BLAST searches for proteins with a high Identity value
DrugBank search for active drugs against identified proteins
Cross-reference the results with the Task Force's GitHub repository in order to find new interactors
Search DrugBank to broaden the number of active drugs
Dataset partition into Training, Test and Validation Set

Figure 3.5: Pipeline used for the generation of datasets 4.

Summary

The table 3.4 shows a summary of the active and inactive used for each dataset

Table 3.4: Summary of the active and inactive used for each dataset

Dataset	Molecules	
	Actives	Inactives
Dataset 1	716	716
Dataset 2	869	6278
Dataset 3	24773	64600
Dataset 4	213	940

Chapter 4

Molecular Fingerprints

This chapter discusses the studies performed on Molecular Fingerprints. The study started from the selection of 7 Molecular Fingerprints each of which presented a different information of the molecule, due to its generation algorithm. The first exploratory analysis was focused on the single fingerprints, to analyze the best version in terms of size (256, 512, 1024, 2048bit), followed by the analysis of the best combination of fingerprints, up to the creation of **EMBER** (**EMB**edding multipl**E** molecular finge**R**prints) the multispectral representation that included them all together.

4.1 Fingerprints generation

The Molecular Fingerprints used for my research activity, can be grouped into two major groups: the pattern-based fingerprints specifically RDKit, Atompair, Torsion and Layered generated through a linear kernel and the circular fingerprints, specifically, Morgan and FeatMorgan and ECFP, generated with a circular kernel. Within the same group, each of the fingerprints encodes the molecular structure differently, and it is because of this characteristic that they were selected. The generation of the various Molecular Fingerprints was performed using the same workflow for all experiments conducted, namely the KNIME framework ([117]), an open-source platform under GPLv3 license for data analysis, reporting and integration. The modules integrated within it are different, such as the "chemistry development kit" or R (software), allowing the user to manipulate the structure of the nodes that will constitute the workflow. The nodes used for the generation of fingerprints, are respectively "RDKit Fingerprint" which is part of the extension "RDKit Nodes Feature" and the node "Fingerprint" which is part of the extension KNIME-CDK. Both nodes, combined with each other, allow to generate all known state of the art

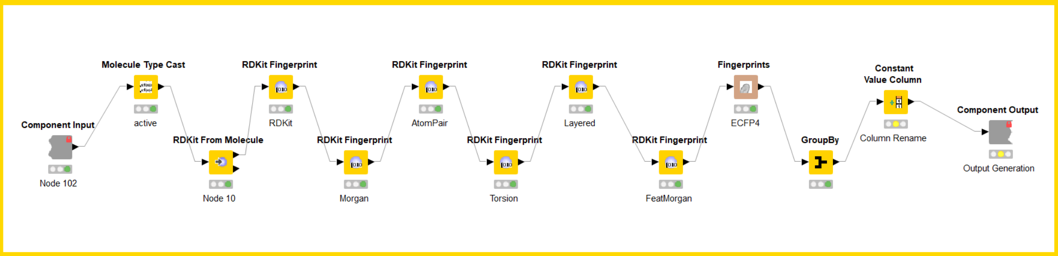


Figure 4.1: The KNIME workflow used for the generation of Molecular Fingeprints

Fingerprints. A graphical representation of the KNIME workflow is shown in figure 4.1.

The choice of this framework is also related to the efficiency of the generation process. In fact, for the Virtual Screening the number of molecules to be tested is very high, so the ability of KNIME to use computational operations on GPU instead of CPU has been found to be advantageous. The pipeline was run for each of the datasets described in chapter 3.

4.2 Dimensional Exploration of Molecular Fingerprints

Molecular Fingerprints are bitmaps whose size varies from 256 to 8192bits, and can be determined during their generation. For the same fingerprints, the generation algorithm is the same each time it is created, but if the size of the defined bitmap is changed it will not contain the same chemical information. As described in the [insert label section], bits 1 are added through a hashing function in "bit wise OR" developing bit collision phenomena. The probability of these events grows inversely proportional to the size of the fingerprints. At most, the fingerprints too large, have a low probability of bit collision, but are very scattered, making more complex mathematical operations applied to them. In order to find the right compromise between the size of the fingerprints, to obtain a computationally efficient embedding and with sufficient chemical information contained in them, I conducted experiments using a convolutional neural network (CNN) in order to identify the version best suited to the needs of the domain. The Molecular Fingerprints tested were: the pattern-based fingerprints specifically RDKit, Atompair, Torsion and Layered and the circular fingerprints Morgan and FeatMorgan. Each of them encodes the information of the molecular structure in a different way, so

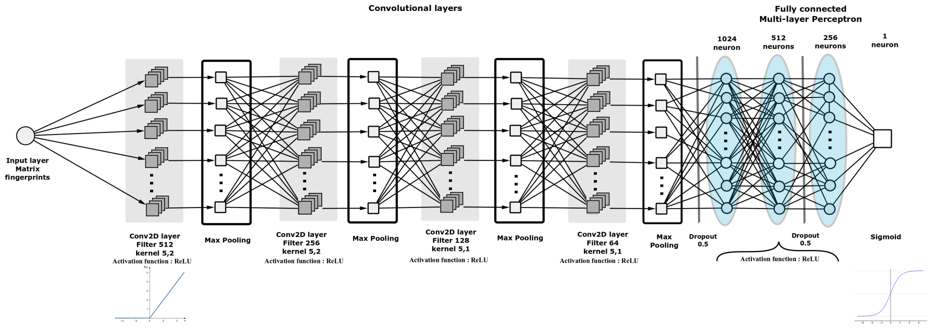


Figure 4.2: Bi-dimensional architecture of the network

in this first phase of the study I wanted to examine whether the performance in a classification task varied with the size of the fingerprints both when used individually and in combination with each other. Experiments were conducted using Dataset 1 (716 active molecules and 716 inactive molecules) described in section 3.2 to perform a binary classification task between active and inactive molecules on CDK1. Given the nature of the task and the numerical embedding that the fingerprints present, the use of a CNN appeared to be the best choice. The fingerprints were tested, individually and in various combinations for various dimensionality (256, 512 and 1024bit). Bitmaps of larger size were not used because of their high sparsity. Two different convolutional neural architectures were trained from scratch; the first is a 1D CNN trained on single fingerprints and the second is a 2D CNN trained on various combinations of fingerprints arranged as a two-dimensional matrix. This second representation was intended to fill in the chemical information gaps presented by the individual fingerprints. Both networks have the same structure for feature extraction formed by 4 convolutional layers with 512, 256, 128, 64 filters, respectively with ReLU activation, each followed by a 2x2 Max Pooling, differing only for the dimensionality of the convolutional kernel. The classification is achieved through MLPs (MultiLayer Perceptron) with 1024, 512 and 256 ReLUs, while the output of the architecture was represented by a single sigmoidal neuron to perform the binary classification. A schematic representation of the 2D architecture is shown in figure 4.2.

For both architectures an intensive tuning of the hyperparameters was conducted by performing a grid search with the following values; convolutional filters tested [1024, 512, 256, 128, 64] in combination with all available padding values of the keras framework; the 2D kernels tested were (20,2), (20,1), (15,2), (15,1), (5,2), (5,1), (4,2), (4,1), (3,2), (3,1) while the 1D ones 2, 3, 4. The learning rate was multiplied by 10 in the range [10^{-6}, 1; $2 * 10^{-5}$, 0.2]. the Dropout probabilities were in the range [0.2, 0.9] with steps of 0.1. The results obtained in this first phase of the experimentation are shown in tables 4.1 and 4.2, in which are reported, respectively, the best values obtained using single fingerprints and the best results obtained from the combination of the various fingerprints for all the various dimensions sampled. The metrics used to verify the performance of the model were the global accuracy of the model, the loss, the F1-score and the Area Under the Curve (AUC).

Table 4.1: Results of 1D CNN on the test set. Best/worst values for each column are in bold/italic

Length	Fingerprint	Accuracy	Loss	F1-score	AUC
1024	Layered	0.9100	0.54	0.8700	*0.9453*
512	**Layered**	**0.9272**	**0.4447**	**0.9000**	**0.9610**
256	*Torsion*	*0.8654*	*0.5456*	*0.8310*	0.9481

As shown in table 4.1 and 4.2 the best overall performance is achieved by the two-dimensional representation consisting of Morgan, Torsion and Layered at 512 bits (MTL-512, table 4.2). The Layered fingerprints is the fingerprints that reported the best results, both individually and in all combinations of fingerprints for the various dimensions. These results gave me an important food for thought, confirming the initial idea that the information contained within the fingerprints are complementary, because they take into account different aspects of the same molecule. The combined use of the various Molecular Fingerprints is therefore an important aspect that will be explored in the following sections. Once the results were obtained, I studied the characteristics of the individual embeddings, delving into the literature information shared by the cheminformatics community, which allowed me to identify the best embedding to use as input data for deep neural networks. Although the results obtained show better performance for 1D or 2D fingerprints with 512 bits, the final choice fell on the embedding of 1024bit. The 512bit fingerprints have a higher probability of bit collision, wasting much of the chemical information contained within the molecular structure. Consequently, considering the satisfactory results obtained by the fingerprints with 1024bit and evaluating a reduction of bit collision of almost 50% I selected this representation for the continuation

Table 4.2: Results of the 2D CNN on the test set with different fingerprint length. Fingerprint types: *(R)DKit*, *(M)organ*, *(A)tompair*, *(T)opological Torsion*, *(L)ayred*, and *(F)eatMorgan*. Best/worst values for each column are in bold/italic

(a) 1024 bit fingerprints

Fingerprints	Accuracy	Loss	F1-score	AUC
M,L	**0.9200**	**0.5600**	0.8800	**0.9563**
R,M,A	*0.9000*	*0.6800*	*0.8600*	0.9527
M,A,L,F	0.9200	0.6000	**0.8877**	*0.9444*
R,M,A,L,F	0.9163	0.6082	0.8820	0.9513
R,M,A,T,L,F	0.8945	0.6280	0.8557	0.9494

(b) 512 bit fingerprints

Fingerprints	Accuracy	Loss	F1-score	AUC
M,F	*0.8981*	*0.4463*	*0.8679*	0.9555
M,T,L	**0.9345**	**0.3900**	**0.9117**	**0.9685**
R,M,T,F	0.9418	0.4268	0.9001	*0.9400*
R,A,T,L,F	0.9127	0.4052	0.8867	0.9630
R,M,A,T,L,F	0.9236	0.3950	0.9004	0.9774

(c) 256 bit fingerprints

Fingerprints	Accuracy	Loss	F1-score	AUC
L,F	0.9090	**0.4087**	0.8792	0.9655
R,L,F	**0.9127**	0.4734	**0.8846**	**0.9606**
R,A,L,F	0.9054	0.4914	0.8749	0.9572
R,M,T,L,F	*0.8909*	0.5380	*0.8623*	0.9624
R,M,A,T,L,F	0.8981	*0.5982*	0.8679	*0.9537*

of my research activity, defining it as the best compromise between the characteristics required for a descriptor to be used in Drug Discovery. All the results obtained during this phase have been published as a contribution to a conference [123].

4.3 Combination of fingerprints

After defining the best embedding dimension, based on the results obtained from the two-dimensional representations, I hypothesized a combined use of the fingerprints testing them on a larger dataset, through the use of convolutional neural networks. The fingerprints used in this phase are 7 (RDKit, Morgan, AtomPair, Torsion, Layered, FeatMorgan, ECFP4), to describe from multiple points of view the candidate molecule. As regards of fingerprints types we selected, they can be grouped into two classes: pathway-based, also known as topological, and circular. Pathway-based fingerprints include RDKit, Atompair, Torsion, and Layered. In this case, the kernel is linear, and each fingerprint differs in the types of atoms and bonds. Circular fingerprints include Morgan, Featmorgan, and ECFP4.In this case the kernel is circular and takes into account the neighborhood of each atom based on the selected radius (usually 1 to 3). The algorithm for generating each of these fingerprints is described in detail in 2.5. The choice of these 7 fingerprints is related to the intention to represent all characteristic of the molecule in a complementary way, integrating all structural information.

Translated with www.DeepL.com/Translator (free version) The experiments were conducted using Dataset 2 (545 active molecules, 4907 inactive molecules, see section 3.2.1), deep neural networks and shallow Machine Learning algorithms were used in order to evaluate the overall performance on the task. To better study the performance of the tested models two training schemes were used. In Training scheme 1, a classical ML approach was followed for the training, where the number of molecules between training, validation and test set is strongly unbalanced towards the training, maintaining a constant ratio of $1 : 10$ between active and inactive among the 3 sets obtained. The second training procedure (Training scheme 2) was developed taking into account the general distribution of the population of molecules, which is strongly unbalanced towards the inactive. For this reason, the dataset was divided into a training set, with a balanced ratio of active/inactive and a test set with a ratio of $1 : 50$ active/inactive. All the models obtained were trained and tested on both training schemes. The most widely used state-of-the-art Machine Learning algorithms, specifically Support Vector Machine (SVM) and Random Forest (RF) were used as baselines for the experiments conducted with DNNs. The parameters selected for both models were obtained through a classical grid search. Particularly, a Radial Basis Function-SVM was trained and the best performances were obtained with a value of $\gamma = 1$ while regularization $C = 5$ for training scheme 1 and $\gamma = 0.1$ and $C = 1$ for training scheme 2. A grid search was also performed for the Random Forest model, obtaining two algorithms that performed better on training scheme 1 with 100 estimators and the *Gini index*, while for training scheme 2 the estimators were 2 again with the *Gini* index. CNN1D and 2D architectures have the same structure, consisting of 4 convolutional layers with 128, 64, 32, 16 filters followed by 2x2

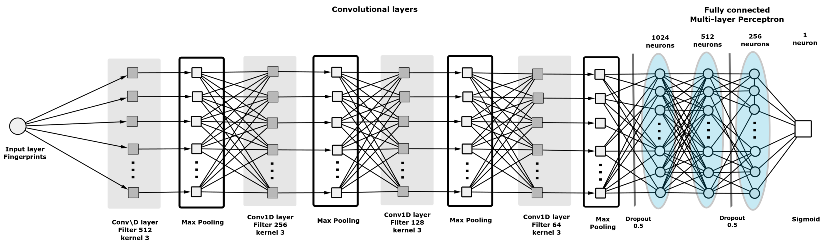


Figure 4.3: One-dimensional convolutional architecture.

Max Pooling and ReLU activation. Each network achieved classification using MLP with 1024, 512, 256 ReLUs per layer, while the output was sigmoidal in order to achieve binary classification. A schematic representation of the 1D architecture is shown in figure 4.3.

Two other architectures have been created, to allow a different approach to those tested so far. The idea behind these two new ensemble architectures is to use all the individual fingerprints simultaneously during the training phase, extracting features by integrating all the downstream information to perform the classification. The two architectures were formed by 7 1D CNN networks working in parallel on the 7 fingerprints with respectively 4 convolutional layers with 512, 256, 128, 64 filters, followed by a 2x2 Max Pooling. The classification block of each individual network was identical to that of CNN1D and CNN2D described above, with a sigmoidal unit output to all networks. The two NN Ensembles differed in the region that integrated the feature maps obtained from the individual 1D CNNs. The Voting architecture, in fact, used a classification based on the voting of the single networks, going to evaluate if the largest number of predicted labels was consistent with the expected one. The second ensemble architecture, Tuned-MLP-Out was a much more refined version that did not simply analyze the result obtained from the individual networks, but used an MLP layer of 3 ReLUs to integrate the various probabilities obtained from the sigmoids of each individual network. Classification was obtained with an additional sigmoidal neuron. The entire Tuned-MLP-Out architecture is shown in figure 4.4.

Also for these 4 architectures a tuning of the hyperparameters was performed, using the grid search approach, testing all possible configurations

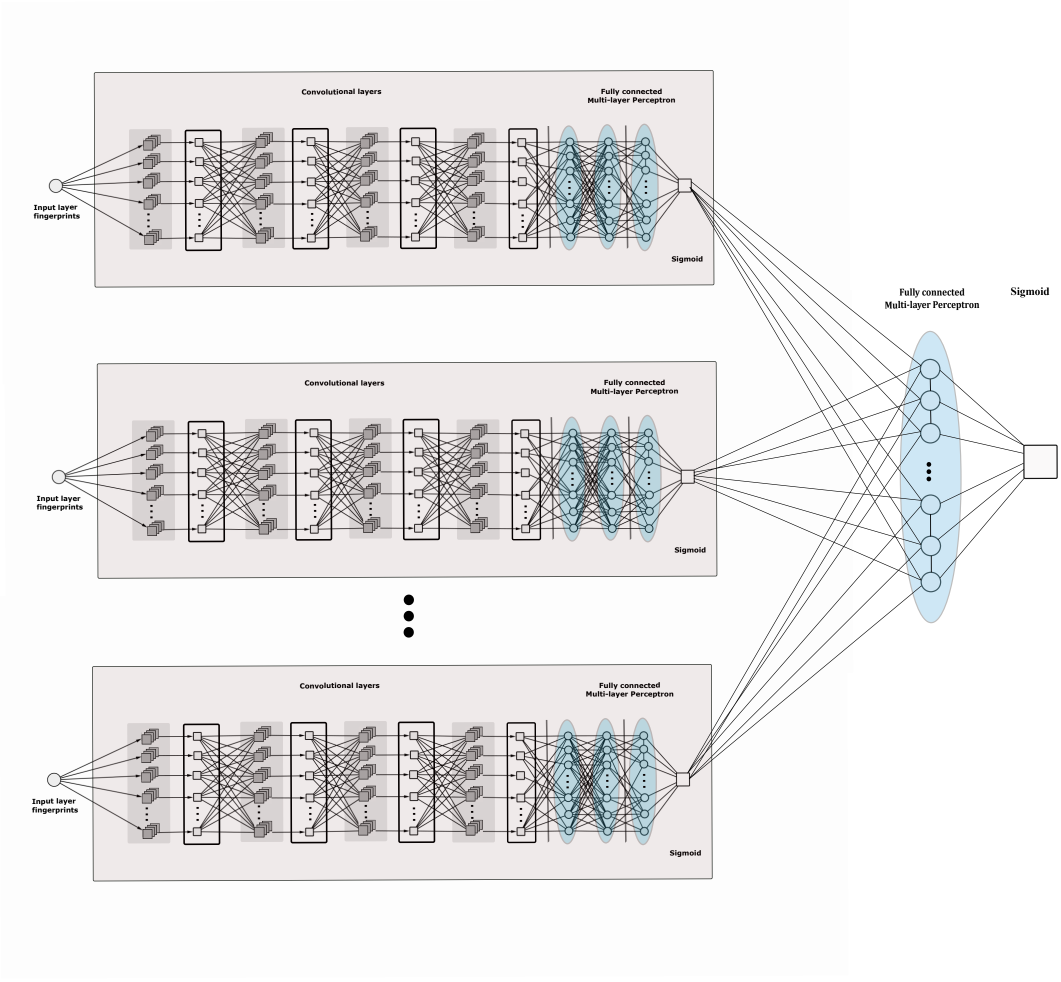


Figure 4.4: Tuned-MLP-Out. The complex architecture with MLP Classifier.

between convolutional filters, kernel sizes, optimizers and learning rate. All training procedures were conducted using K-Fold cross validation to compare and select only the models that performed best. In order to reduce the possibility of incurring in overfitting also the Early stopping procedure was used, with a patience of 50 training epochs.

Model performance was evaluated using several metrics. Balanced accuracy bACC $(TP/P + TN/N)/2$, or the average of *Sensitivity*, the number of true positives predicted over the total number of positives, and *True Negative*, or the ratio of predicted negatives to negatives in the entire test set, was used to evaluate active/inactive discrimination for both training schemes. On account of the application domain, Sensitivity was used as the main evaluation metric for model performance. In fact, being able to correctly detect the highest number of true positives is the basis of the screening experiments that are being carried out. Also the *Matthews correlation coefficient* (MCC) was used as a discrimination measure. MCC is a well known index used for binary classification, that returns a value in $[-1; 1]$; for a 2×2 contingency table, that is a binary classifier's confusion matrix, MCC is related to the chi-square statistic as $\|MCC\| = \sqrt{\chi^2/n}$ where n is the number of observations. MCC thus measures the dependency of the predictions from the true (i.e. expected) labels. On the other hand, just sensitivity has been used in the active only selection task because we want to maximize correct prediction of active compounds in spite of accepting a relevant number of false positives.

The other metrics shown in the table are Loss, F1-score, and AUC. The results obtained are shown in table 4.3 and 4.4.

Table 4.3: Results for the active/inactive discrimination task, and Training scheme 1. Best/worst values for each column are in bold/italic.

Architecture	Bal. accuracy	Sensitivity	Loss	AUC	F1-score	MCC
Tuned-MLP-Out	**0.9880**	**0.9855**	**0.0405**	**0.9979**	**0.9510**	**0.9462**
Voting	0.9768	0.9710	0.2093	0.9920	0.8965	0.9033
CNN 1D (F)	0.9687	0.9710	0.0688	0.9904	0.8979	0.8813
CNN 2D (R-M-F)	0.9679	0.9565	0.0770	0.9912	0.8918	0.8817
Random Forest (F)	0.9510	0.8985	0.6405	0.9837	*0.6065*	0.8962
SVM (F)	*0.9421*	*0.8985*	*0.7883*	*0.9868*	0.8857	*0.8731*

Fingerprint types: *(R)DKIT,(M)organ, (F)eatMorgan, (L)ayered*

As can be seen, the best performances are achieved by the Voting-MLP-Out ensemble architecture for both training schemes. Specifically, the best results are obtained with the classical training approach (*Training Scheme 1*, table 4.3), where the model outperforms all other models for all metrics

Table 4.4: Results for the active/inactive discrimination task, and Training scheme 2

Architecture	Bal. Accuracy	Sensitivity	Loss	AUC	F1-score	MCC
Tuned-MLP-Out	**0.9644**	**0.9625**	**0.0983**	0.9875	0.5519	0.5989
Voting	0.9639	0.9500	0.1523	**0.9889**	0.6379	**0.6694**
CNN 1D (F)	0.9579	0.9625	0.1398	0.9854	0.4709	*0.5336*
CNN 2D (T-L-E)	0.9525	0.9375	0.1054	0.9841	*0.5192*	0.5920
Random Forest (F)	*0.8789*	*0.7750*	*0.6221*	0.9541	0.6528	0.6540
SVM (F)	0.9208	0.8625	*0.6221*	0.9682	**0.6699**	0.6524

Fingerprint types: *(F)eatMorgan, (T)orsion, (L)ayered, (E)CFP4*

considered. Of note is the Sensitivity value that demonstrates the excellent ability to discriminate TP molecules within the test set. To demonstrate the classificatory capabilities of our models a further study using the TP/P value was conducted. In contrast to the Total Sensitivity, in this case a different percentage of the test was used to demonstrate that the molecules prioritized by the model were indeed active. For the two training schemes, the calculated TP/P percentages were different due to the number of actives present in the two test sets. For Training Scheme 1 a TP/P was calculated using 1%, 2%, 5% and 10% of the test set. For Training Scheme 2 only a TP/P value was calculated at 1% and 2% due to the small number of actives present. The results obtained are shown in table 4.5 and 4.6.

Table 4.5: True Positives versus Positives ratio computed on the test set 1 (70 active molecules out of 701 compounds).

Architecture	TP/P 1%	TP/P 2%	TP/P 5%	TP/P 10%
Tuned-MLP-Out	7/7	14/14	34/35	65/69
Voting	7/7	14/14	34/35	61/69
CNN 1D (M)	7/7	13/14	33/35	62/69
CNN 2D (R-M-F)	7/7	12/14	32/35	61/69
RF(F)	7/7	14/14	35/35	63/69
SVM(F)	7/7	14/14	35/35	61/69

Fingerprint types: *(R)DKIT,(M)organ, (F)eatMorgan,*

As can be seen from your tables, the *Tuned-MLP-OUT* architecture turns out to be the best for both training schemes managing to correctly prioritize all 1 and 2% active molecules with both training schemes.

The excellent performance achieved by *Tuned-MLP-OUT* on the training scheme 1 shows how with a classical approach the network is able to correctly prioritize the active compounds, satisfying the requirements of a classifier. In order to stress further the architecture, a further experimentation has been

Table 4.6: True Positives versus Positives ratio on the test set 2 (80 active molecules out of 3720 compounds).

Architecture(Training 2)	TP/P 1%	TP/P 2
Tuned-MLP-Out	37/37	65/74
Voting	32/37	57/74
CNN 1D (F)	31/37	52/74
CNN 2D (T-L-E)	31/37	52/74
RF(F)	37/37	62/74
SVM(F)	32/37	55/74

Fingerprint types: *(F)eatMorgan, (L)ayered, (T)orsion, (E)CFP4*

carried out in order to demonstrate the effectiveness of the approach using all the fingeprints in a complementary way. The dataset used in the Training scheme 1 has been resampled increasing the ratio between active and inactive in the training, validation and test set, to 1:20, 1:50 and 1:100. The idea behind this consideration is related to a use in Virtual screening experiments in vivo, where the ratio between active and inactive is not fixed at 1:10. The results obtained with the 3 active/inactive ratios are shown in Table 4.7 and 4.8 and use the same metrics as used for classification.

Table 4.7: Performance of the *Tuned-MLP-Out* network on three data sets with 1%, 2%, and 5% active/inactive proportion respectively. Best/worst values for each column are in bold/italic

A/I*	Bal.Accuracy	Sensitivity	Loss	AUC	F1-score	MCC
1%	*0.7475*	*0.5000*	*0.5116*	0.9700	*0.5333*	*0.5289*
2%	**0.9671**	**0.9375**	0.5114	0.9415	0.9009	0.8226
5%	0.9382	0.8780	**0.0565**	**0.9991**	**0.9230**	**0.9196**

* Active/inactive proportion

Table 4.8: True Positives versus Positives ratio computed on the test set with 1%, 2%, and 5% active/inactive proportion respectively.

A/I*	TP/P 1%	TP/P 2%	TP/P 5
1%	4/8	-	-
2%	7/8	7/16	-
5%	4/8	8/16	20/41

*Active/inactive proportion. 1% = 8 active molecules; 2% = 16 active molecules; 5% = 41 active molecules.

As expected, the bACC and Sensitivity values decreased from the results shown in Table 4.3. This is due to the extreme imbalance between the two classes that were assayed. Nevertheless, the results are positive if we analyze the discriminative abilities of the actives in Table 4.8 where it is shown that the model is able to prioritize the most active molecules even at a ratio of 1:100. Ultimately, the results obtained in the research phase have shown us the true potential in the task of classification of Molecular Fingeprints, but even more they have shown us how the information contained therein, although generated by the same molecule, are encoded differently. Consequently, their simultaneous use allows the neural architecture to extract all the most important features to correctly classify active molecules from inactive ones, integrating the various chemical information. Nevertheless, problems have been encountered with the use of this approach based on the Tuned-MLP-OUT ensemble architecture, especially from a computational point of view. The neural network as described above consists of 7 1D CNNs working in parallel with each other and subsequently connected by an MLP layer before classification for a total of $51,449,735$ parameters leading to a very high computational cost even for a relatively small dataset like the one used. For this reason, given the proven effectiveness of using the 7 Molecular Fingeprints simultaneously, and not being able to use further ensemble approaches given the high number of parameters, I assumed that creating a new descriptor representation was the best solution. From these results comes the idea of **EMBER** - embedding multiple molecular fingerprints, a multispectral descriptor based on the 7 Molecular Fingeprints that aims to preserve the chemical information of the different descriptors, greatly reducing the computational cost required for the training of the neural network.

4.4 EMBER - embedding multiple molecular fingerprints

In this last phase of the study is exposed the **EMBER** descriptor (embedding multiple molecular fingerprints), a representation of 7 Molecular Fingeprints (RDKit, Morgan, AtomPair, Torsion, Layered, FeatMorgan, ECFP4) stacked as a spectrum or rather as a "molecular image". This embedding, aims to provide all the characteristic information of each fingerprints without the need to build neural network ensembles such as Tuned-MLP-Out. The proposed molecular embedding is designed to exploit the ability of convolution operations to extract features as if they were real images.

EMBER was tested on multi-class multi-target classification tasks on Dataset 3 (see section 3.2.2), the largest dataset used during these 3 years of research. The dataset consists of 83373 molecules with known bioactivity on 20 different target proteins, obtained through similarity studies on CDK1. The dataset was divided into traning, validation and test set with a ratio of $1:100$ between active and inactive with reference to the least abundant protein, namely CDK6. In addition to classification performance evaluations, explainability analyses were conducted to identify features relevant to classification, and the results of this analysis confirm some very recent in vitro studies that outline the relevance of pharmacofore-like fingerprints when addressing bioactivity classification for kinase inhibitors.

The classifier architecture is a deep CNN with 9 Parametric ReLU (PReLU) layers for feature extraction and 3 MLP layers for classification. The layout of the architecture is shown in Figure 4.5(a).

$$\begin{aligned}\mathrm{PReLU}_i(x) &= \begin{cases} x & \text{if } x > 0 \\ \alpha_i x & \text{if } x \leq 0 \end{cases} \\ &= \max(0, x) + \alpha_i \min(0, x)\end{aligned} \tag{4.1}$$

If $\alpha_i = 0$, then PReLU degenerates to ReLU; if α_i is a small fixed value (such as $\alpha_i = 0.01$), then PReLU degenerates to Leaky ReLU (LReLU). In our work α_i has been set constant at 0.25.

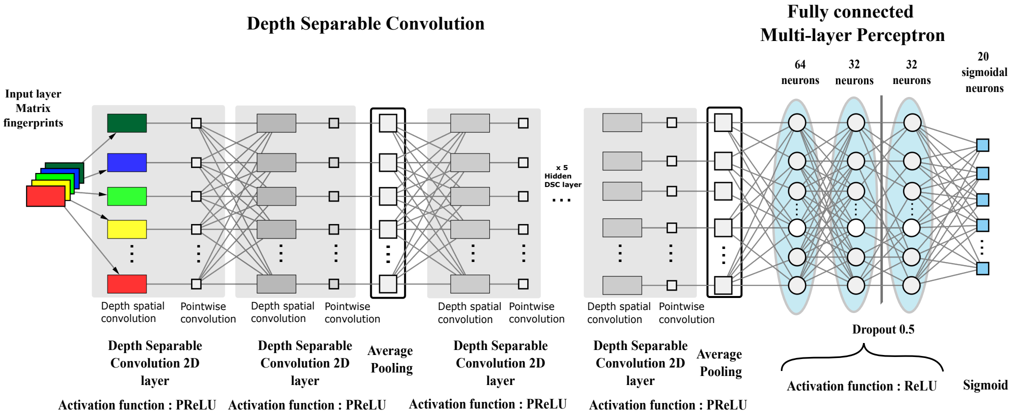
The network is trained on $7 \times 1024 \times 1$ input tensors that represent the seven 1024 long fingerprints stacked as the channels of a 1024×1 image. Multi-target bioactivity prediction is a *multi-class, multi-label* classification, that is our classifier has to assess also if a ligand is active at the same time on different targets. As a consequence, the output is a *vector label* that is a binary vector where the 1s indicate bioactivity with respect to a particular target.
In line with the most recent CNNs, we implemented the convolutional layers using *Depthwise Separable Convolution* (DSC) [124] to reduce the network parameters, and lower the computational load. The classical convolution operator computes an element of the output tensor Y by applying a kernel K with spatial extent $s \times s$ and depth d to the input tensor X:

$$\mathsf{Y}_{i,j,k} = \sum_{l=1}^{s} \sum_{m=1}^{s} \sum_{n=1}^{d} \mathsf{X}_{i-l,j-m,k-n} \mathsf{K}_{l,m,n} \tag{4.2}$$

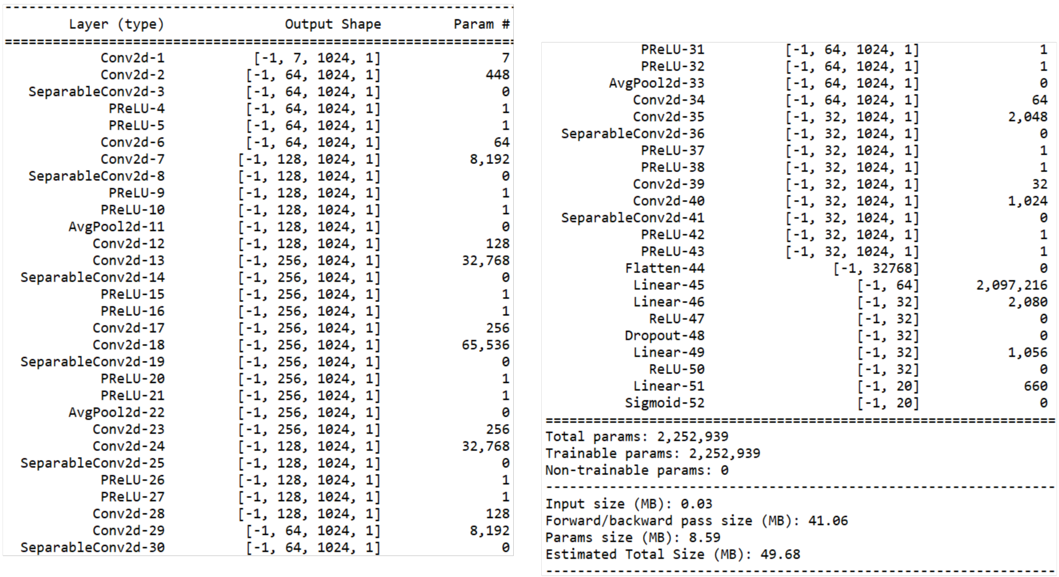
Here we are using the proper index notation for convolution without kernel flipping. In DSC, d *spatial* kernels $\mathsf{K}^{S}_{(h)}$ with $s \times s$ size compute 1-depth convolutions, and a $1 \times 1 \times d$ *depth* kernel K^{D} gives the final convolution output.

$$\begin{aligned}
\mathsf{Y}^{(h)}_{i,j} &= \sum_{l=1}^{s}\sum_{m=1}^{s} \mathsf{X}_{i-l,j-m,h}\mathsf{K}^{S}_{(h)\ l,m}, \quad h = 1\dots d \\
\mathsf{Y}_{i,j,k} &= \sum_{n=1}^{d} \mathsf{Y}^{(h-n)}_{i,j}\mathsf{K}^{D}_{n}
\end{aligned} \tag{4.3}$$

It can be shown that DSC can reduce the number of parameters by a factor $1/s^2$ for each layer: our network was built using just $2,252,959$ parameters, that is about a $1:25$ ratio with the size of the CDK1 only classifier proposed in previous section. Figure 4.5(b) reports the detail of the implemented model.


Depth Separable Convolution
Fully connected Multi-layer Perceptron
Input layer Matrix fingerprints
64 neurons
32 neurons
32 neurons
20 sigmoidal neurons
x 5 Hidden DSC layer
Depth spatial convolution
Pointwise convolution
Depth spatial convolution
Pointwise convolution
Depth spatial convolution
Pointwise convolution
Depth spatial convolution
Pointwise convolution
Dropout 0.5
Depth Separable Convolution 2D layer
Depth Separable Convolution 2D layer
Average Pooling
Depth Separable Convolution 2D layer
Depth Separable Convolution 2D layer
Average Pooling
Activation function : ReLU
Sigmoid
Activation function : PReLU
Activation function : PReLU
Activation function : PReLU
Activation function : PReLU

(a) Network layout.



```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1            [-1, 7, 1024, 1]               7
            Conv2d-2           [-1, 64, 1024, 1]             448
   SeparableConv2d-3           [-1, 64, 1024, 1]               0
             PReLU-4           [-1, 64, 1024, 1]               1
             PReLU-5           [-1, 64, 1024, 1]               1
            Conv2d-6           [-1, 64, 1024, 1]              64
            Conv2d-7          [-1, 128, 1024, 1]           8,192
   SeparableConv2d-8          [-1, 128, 1024, 1]               0
             PReLU-9          [-1, 128, 1024, 1]               1
            PReLU-10          [-1, 128, 1024, 1]               1
        AvgPool2d-11          [-1, 128, 1024, 1]               0
           Conv2d-12          [-1, 128, 1024, 1]             128
           Conv2d-13          [-1, 256, 1024, 1]          32,768
  SeparableConv2d-14          [-1, 256, 1024, 1]               0
            PReLU-15          [-1, 256, 1024, 1]               1
            PReLU-16          [-1, 256, 1024, 1]               1
           Conv2d-17          [-1, 256, 1024, 1]             256
           Conv2d-18          [-1, 256, 1024, 1]          65,536
  SeparableConv2d-19          [-1, 256, 1024, 1]               0
            PReLU-20          [-1, 256, 1024, 1]               1
            PReLU-21          [-1, 256, 1024, 1]               1
        AvgPool2d-22          [-1, 256, 1024, 1]               0
           Conv2d-23          [-1, 256, 1024, 1]             256
           Conv2d-24          [-1, 128, 1024, 1]          32,768
  SeparableConv2d-25          [-1, 128, 1024, 1]               0
            PReLU-26          [-1, 128, 1024, 1]               1
            PReLU-27          [-1, 128, 1024, 1]               1
           Conv2d-28          [-1, 128, 1024, 1]             128
           Conv2d-29           [-1, 64, 1024, 1]           8,192
  SeparableConv2d-30           [-1, 64, 1024, 1]               0
            PReLU-31           [-1, 64, 1024, 1]               1
            PReLU-32           [-1, 64, 1024, 1]               1
        AvgPool2d-33           [-1, 64, 1024, 1]               0
           Conv2d-34           [-1, 64, 1024, 1]              64
           Conv2d-35           [-1, 32, 1024, 1]           2,048
  SeparableConv2d-36           [-1, 32, 1024, 1]               0
            PReLU-37           [-1, 32, 1024, 1]               1
            PReLU-38           [-1, 32, 1024, 1]               1
           Conv2d-39           [-1, 32, 1024, 1]              32
           Conv2d-40           [-1, 32, 1024, 1]           1,024
  SeparableConv2d-41           [-1, 32, 1024, 1]               0
            PReLU-42           [-1, 32, 1024, 1]               1
            PReLU-43           [-1, 32, 1024, 1]               1
          Flatten-44                 [-1, 32768]               0
           Linear-45                    [-1, 64]       2,097,216
           Linear-46                    [-1, 32]           2,080
             ReLU-47                    [-1, 32]               0
          Dropout-48                    [-1, 32]               0
           Linear-49                    [-1, 32]           1,056
             ReLU-50                    [-1, 32]               0
           Linear-51                    [-1, 20]             660
          Sigmoid-52                    [-1, 20]               0
================================================================
Total params: 2,252,939
Trainable params: 2,252,939
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.03
Forward/backward pass size (MB): 41.06
Params size (MB): 8.59
Estimated Total Size (MB): 49.68
----------------------------------------------------------------
```

(b) Model summary.

Figure 4.5: The proposed architecture

Classification is achieved through a MLP with 64/32/32 ReLU units per layer respectively, while the output consists of 20 sigmoidal units because the probabilities of each class is independent from the other class probabilities. For this reason a *binary crossentropy* loss function has been used instead of the usual *categorical crossentropy*. This choice is reasonable because the network performs a "multi-label" "multi-class" classification task.

The results obtained are shown in Table 4.9 and report the best results with respect to Accuracy, Loss, Sensitivity, MCC, AUC, and F1-Score.

Table 4.9: Accuracy metrics for all the targets. Best/worst values for each column are in bold/italic

Target	**Acc.**	**Loss**	**Sensitivity**	**MCC**	**AUC**	**F1-score**
ACK	0.9957	0.0226	0.5000	0.6742	0.9834	0.6463
ALK	0.9930	0.0402	0.6575	0.7913	0.9904	0.7804
CDK1	0.9910	0.0314	0.4537	0.6397	0.9850	0.6059
CDK2	0.9859	0.0431	0.5281	0.6338	0.9845	0.6287
CDK6	0.9966	0.0210	0.5865	0.7523	0.9895	0.7305
INSR	0.9893	0.0329	0.3779	0.5830	0.9858	0.5342
ITK	0.9945	0.0232	0.5886	0.7302	0.9905	0.7154
JAK2	0.9898	0.0472	**0.8474**	**0.9090**	**0.9950**	**0.9114**
JNK3	**0.9967**	**0.0154**	0.5905	0.7610	0.9901	0.7381
MELK	0.9957	0.0229	0.7081	0.8270	0.9897	0.8188
CHK1	0.9895	0.0512	0.6385	0.7650	0.9846	0.7565
CK2A1	0.9942	0.0253	0.5166	0.6944	0.9857	0.6667
CLK2	0.9936	0.0259	*0.2255*	*0.4137*	0.9771	*0.3485*
DYRK1A	0.9916	0.0321	0.4080	0.5987	0.9776	0.5591
EGFR	0.9845	*0.0604*	0.7536	0.8331	0.9874	0.8357
ERK2	0.9881	0.0563	0.7295	0.8292	0.9886	0.8272
GSK3	*0.9843*	0.0554	0.5827	0.6892	*0.9762*	0.6856
IRAK4	0.9936	0.0287	0.7611	0.8611	0.9938	0.8571
MAP2K1	0.9931	0.0319	0.5497	0.7184	0.9795	0.6954
PDK1	0.9945	0.0271	0.6310	0.7757	0.9875	0.7613

As can be seen, the overall performance of the classifier on a single target is very good, in terms of overall Accuracy, with particular reference to JNK3 which has the highest Accuracy and Loss value. In the other metrics, the target on which the classifier performs best is JAK2, the protein with the highest number of molecules.

The screening capabilities of the model are highlighted by the results in Table 4.10 which contains True Positive/Positive (TP/P) and Enrichment factor (EF) values at different percentages of the test set for each target. EF after

$x\%$ of the focused library were calculated according to the following formula

$$EF = \frac{N_{\text{experimental}}{}^{x\%}}{N_{\text{expected}}{}^{x\%}} = \frac{N_{\text{experimental}}{}^{x\%}}{N_{\text{active}} \cdot x\%} \tag{4.4}$$

where $N_{experimental}$ is the number of experimentally found active structures in the top $x\%$ of the sorted database, $N_{expected}$ is the number of expected active structures, and N_{active} is total number of active structures in database[125] . The EF computes the number of predicted true actives, in decreasing probability order, in a fixed percentage of the test set. Typical percentages are 5% and 10% but in this study we tested also the performance at 1%. Such a measure is intended to provide the number of times a particular screening procedure performs better than a pure random process.

Table 4.10: True Positives versus Positives ratio and Enrichment Factors computed on the entire test set.

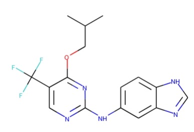
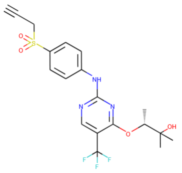
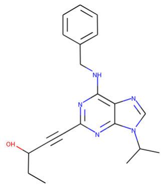
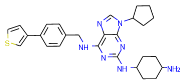
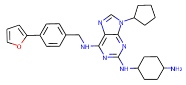
Protein	TP/P 1%*	TP/P 2%*	TP/P 5%*	TP/P 10%*	EF 1%	EF 2%	EF 5%	EF 10%
ACK	72/106	84/106	95/106	101/106	68	40	18	10
ALK	131/254	202/254	229/254	247/254	52	40	18	10
CDK1	111/205	150/205	189/205	196/205	54	37	18	10
CDK2	118/303	194/303	264/303	289/303	39	32	17	10
CDK6	79/104	90/104	98/104	101/104	76	43	19	10
INSR	110/217	145/217	195/217	206/217	51	33	18	9
ITK	107/158	125/158	148/158	155/158	68	40	19	10
JAK2	134/832	268/832	669/832	804/832	16	16	16	10
JNK3	81/105	88/105	95/105	102/105	77	42	18	10
MELK	130/185	157/185	178/185	181/185	70	42	19	10
CHK1	134/343	233/343	300/343	324/343	39	34	17	9
CK2A1	100/151	117/151	141/151	146/151	66	39	19	10
CLK2	59/102	73/102	87/102	96/102	58	36	17	9
DYRK1A	97/174	126/174	152/174	162/174	56	36	17	9
EGFR	134/702	268/702	586/702	664/702	19	19	17	9
ERK2	133/525	267/525	471/525	505/525	25	25	18	10
GSK3	132/393	226/393	327/393	353/393	34	29	17	9
IRAK4	134/339	263/339	320/339	333/339	40	39	19	10
MAP2K1	118/191	142/191	167/191	178/191	62	37	17	9
PDK1	123/187	149/187	170/187	181/187	66	40	18	10

* percentage relative to the evaluated test set evaluated (13400 compounds), i.e 1% = 134 molecules

EF results are considerably high considering the size of the test set and drop to 9 only at 10%. The prioritization capabilities of the model using EMBER are highlighted by the value of TP / P that also manages to validate the EF values of some proteins such as JAK2 and EGFR that despite having an EF below 20 even at 1% are correctly prioritized at all percentages.

In order to prove the practical effectiveness of our approach we conducted a simple experiment on the ligands prioritized by our classifier for the CDK1 target. We explicitly extracted the ChEMBLIDs of the top five molecules prioritized by our system in the test set, and inspected both their chemical structure and their activity parameters. Table 4.11 reports the results, and it can be seen that all of them are strongly active against the target.

Table 4.11: The top five test set molecules prioritized by our classifier as the most active on the CDK1 target.

Molecule ChEMBLID	**Chemical structure**	IC_{50}
CHEMBL192216		2 nM
CHEMBL3644025		82 nM
CHEMBL445125		500 nM
CHEMBL2403087		183 nM
CHEMBL2403084		148 nM

EMBER presents itself as an excellent embedding for classification, greatly improving the impact that individual fingerprints have on virtual screening. Given the great efficiency, the final phase of the study of Molecular Fingeprints I have performed studies of explainability with the well-known framework SHAP, in order to identify the features that contribute most to the classification. SHAP stands for *SHapley Additive exPlanations* [126] , and it is a game theoretic approach that was proposed first by Lipovetsky and Conklin [127] . In this work, the relevance of each predictor in a linear regression model is measured using the *Shapley Value (SV) imputation* that is a method to rank the importance of each player in a multi-player game over all the possible combinations of players. The authors use the *SHAP values* as the unique measure for feature relevance in an additive feature attribution explainability model, that is defined by a linear combination of the features to be explained z_i weighted by some importance factors ϕ_i. The SHAP value for a feature z_i is estimated as the SV ϕ_i of a conditional expectation function $E[f(z)|z_i]$ describing the expected prediction over the entire feature set z conditioned to z_i. Both model agnostic linear explanation and model specific computation of SHAP values is proposed. In my case, I adopted the so called *Deep SHAP* explanation model that is suited for CNN because it combines SHAP values with the recursive relevance scores computation proposed in *DeepLIFT* [128] . The DeepLIFT explainability model assumes that a difference $\Delta t = t - t_0$ in an output neuron between the actual activation t and a reference one t_0 is related to the activation difference Δx_i in whatever contributing neuron by the *summation-to-delta* property $\sum_i C_{\Delta x_i \Delta t} = \Delta t$ that is a constraint on the relevance scores $C_{\Delta x_i \Delta t}$. Deep SHAP applies the DeepLIFT approach to the expectation function $E[f(z)|z_i]$ reference value.

The results of this analysis are shown in Figures 4.6; on the left are SHAP values for each target and fingerprints averaged over the entire test set, and on the right are CDK1 values, as an example. Here, each fingerprints was grouped into bins of 64 to improve readability.

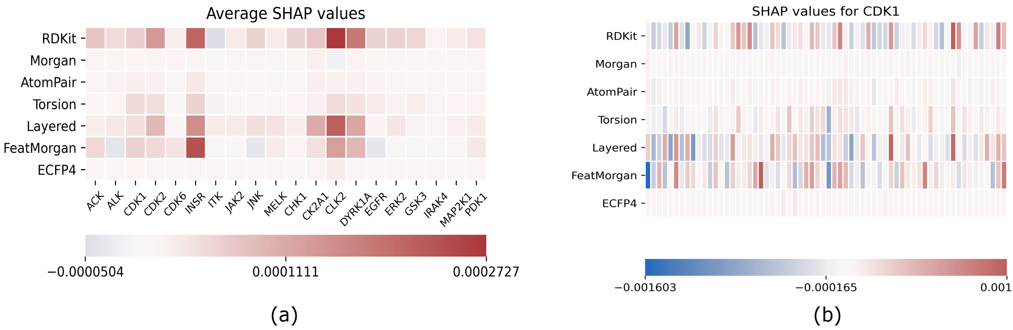


Figure 4.6: Explainability results using SHAP; (a) average SHAP values for each fingerprint computed on the entire test set separately for each target; (b) example of single target explainability analysis for CDK1: SHAP values are reported for each fingerprint, and each row has been grouped in 64 bins to enhance readability

As shown in the figure 4.6, not all fingerprints contribute equally to the classification but we can identify 3 Molecular Fingeprints that show a greater influence on all, consistent with the different information they encode. In fact,

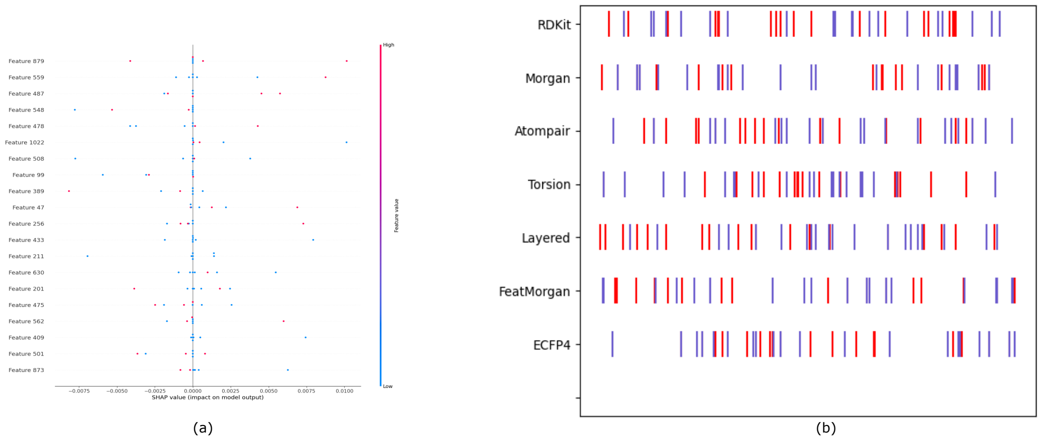


Figure 4.7: Shap values calculated on CDK1. (a) Summary Plot listing the 20 most relevant features in order. (b) Plot of the 20 most positively and negatively relevant features for the 7 Molecular Fingerprin.

RDKit and *Layered* are two Pattern fingerprints [insert label of the chapter molecular descriptor] generated with a kernel that searches for substructures but that store information in a different way, in fact Layered organizes information in layers. *FeatMorgan* is instead the third fingerprints that affects more the classification and differs from the other two for being a Circular Fingerprints of the FCFP family generated through the use of a circular kernel. Plausibly with the different encoding of structural information contained in them the neural network has extracted these features to improve the discrimination between active and inactive molecules further confirming the use of this approach.

To deepen the understanding of the impact of individual fingerprints I performed a study on the entire spectrum of SHAP Value for each fingerprints to be able to identify the most relevant bits. The analysis was performed on all the molecules of the test set on each target. The results obtained are shown in figure 4.7.

As shown in Figure 4.7(b) for each individual Molecular Fingeprints one can identify all the bits that actively participate in classification. This information could be very important for the purpose of understanding the classification mechanisms of the network, but due to the hashing function and the bit collision that can occur between bits 1 when bits are added, it cannot be translated back into chemical language.

This is the biggest limitation found in Molecular Fingerprints. Molecular Fingeprints are excellent embeddings for performing classification, but they encode information irreversibly, preventing the researcher from expanding the search domain in order to identify the characteristic chemical structures of a molecule. In light of these observations, being able to identify a numerical embedding that had the same discriminative capabilities as EMBER, allowing the researcher using it to know the chemical groups involved in the classification, was the second goal of my research activity. At the state of the art, a numerical embedding with these characteristics is not present, and for this reason I created the NMR-Like descriptor that will be described in detail in the following chapter.

Chapter 5

NMR-Like

NMR-Like is the real innovation in my research activity. It is a molecular descriptor that aims to innovate VS approaches while maintaining computationally efficient numerical embedding and preserving chemical information clearly. NMR-Like stems from the idea of using the output obtained from nuclear magnetic resonance (NMR) spectroscopy experiments of Hydrogen (H) atoms as input data for a neural network. In this chapter will be described in detail the workflow that led me to the current version of NMR-Like.

5.1 Nuclear Magnetic Resonance (NMR)

Nuclear Magnetic Resonance (NMR) is a technique of investigation of matter with a considerable number of applications, starting from synthetic chemistry for the characterization of newly synthesized compounds, up to applications in the medical field. It is based on the measurement of spin precession of protons (H) or other nuclei with magnetic moment, when they are subjected to a magnetic field. Atoms with an odd number of protons and neutrons possess a non-zero spin magnetic moment which is defined by the spin quantum number for the values $\frac{1}{2}$ and $-\frac{1}{2}$. One of these two states corresponds to the orientation that the nuclei assume parallel when subjected to a magnetic field, while the other will describe the antiparallel orientation. These two spin states have different energy. Typically nuclei subjected to a static magnetic field tend to assume the state with the lower energy.

This orientation of the nuclear magnetic moments in the static magnetic field B_0 gives rise to a macroscopic magnetization $\mathbf{M}$ obtained from the vector sum of the individual nuclear magnetic moments. At equilibrium, the magnetization M_0 is aligned with the direction of the static magnetic field. The magnetization is a vector quantity, which obeys the rules of classical electrodynamics [129]. Thus, the interaction of magnetic fields with magnetization can be described by the equation of classical physics:

$$\mathrm{d}\boldsymbol{M}/\mathrm{dt} = -\gamma \boldsymbol{M} \times \boldsymbol{B} = -\boldsymbol{M} \times \omega \tag{5.1}$$

where γ is the gyromagnetic constant defined by the ratio of the nuclear magnetic moment to the angular momentum. According to equation [insert reference], if the magnetization vector is aligned with the magnetic field B_0 (typically aligned with the Z axis) the vector product **M** x B_0 is equal to 0 and M is static. However, if **M** and B_0 are not parallel, **M** undergoes a magnetic precession towards B0 with an angular frequency $\omega \mid (\omega = 2\pi v,$ where v is the frequency in Hz). The magnitude of B_0 is several Tesla (T) and the *Larmor frequency* typically hovers in the range of 50 to $900MHz$. The magnetization **M** can be detected only when it is not static and therefore with the same orientation as B_0. It has been observed that by using a second low amplitude radio frequency (RF) magnetic field, B_1, produced by an RF electric current in the coil with the axis perpendicular to B_0 it is possible to rotate the magnetization. The rotation occurs around the Z axis with the Larmor frequency ω. Therefore, by turning on B1 field for a few milliseconds, **M** is tilted towards the x'y' plane. The application of the B1 field for a short time, t, is called the RF pulse. After switching off B_1, the tilted magnetization precesses around B_0 and induces a decaying electromotive force in the coil, according to a mechanism called relaxation which is a first-order process called Free Induction Decay (FID) or NMR signal [129]. The FID, a time domain signal, is difficult to read and is translated into the frequency spectrum of the NMR signal through the use of a Fourier transform, as shown in Figure 5.1.

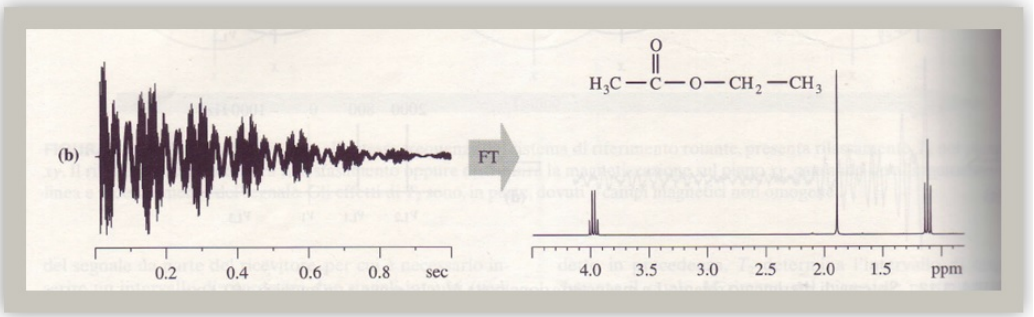


Figure 5.1: On the right we observe the spectrum obtained through the Fourier Transform (FT). The x-axis represents the scale of δ with values ranging from 0 (right) to 10 (left) expressed in ppm, while the y-axis represents the intensity of the resonance signal

It has been observed that nuclei of the same chemical species (e.g., H) within molecules can have slightly different resonant frequencies. This difference arises from the fact that the nuclei are surrounded by electrons that locally produce a magnetic field oriented opposite to the external magnetic field, reducing the current magnetic moment of the nuclei. This phenomenon is called electron screen effect that decreases the resonance frequency of nuclei that is influenced by chemical enviroment and it is characteristic for each chemical group. As the electron shield effect is proportional to the intensity of the external magnetic field B_0, it has been defined a scale expressed in ppm (parts per million) called *chemical shift* that is independent of the intensity of B_0 and allows a standardization of the values obtained with magnetic resonance equipment with different intensity. The chemical shift is defined as follows:

$$\delta = (v - v_{\text{ref}}) / v_{\text{ref}} \tag{5.2}$$

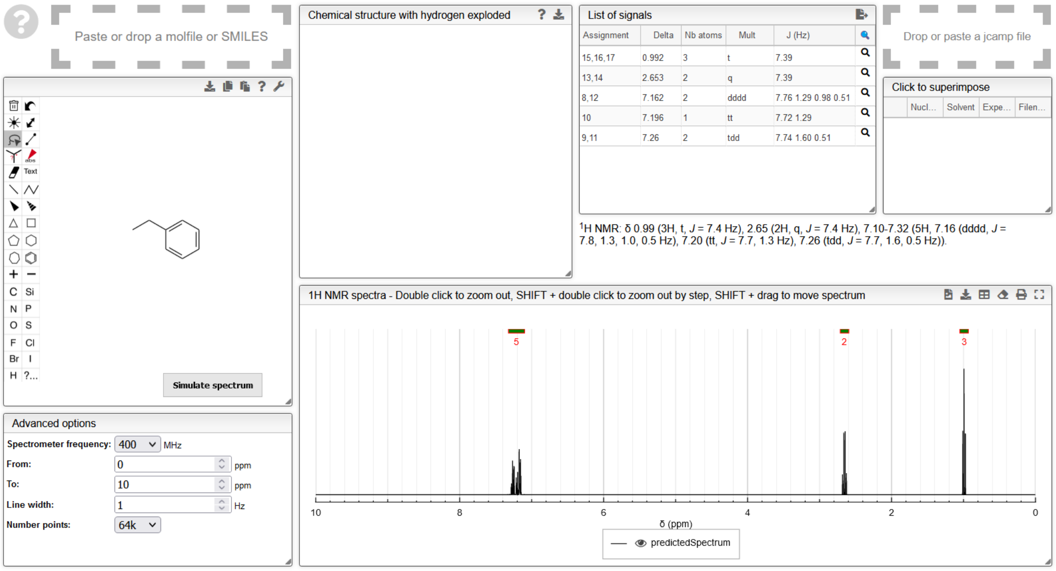
Where v is the resonance frequency in Hz of the spectroscope and v_{ref} is the resonant frequency of a reference compound (typically tetramethylsilane is used for H-NMR). Another phenomenon that characterizes NMR spectra is spin-spin coupling or *J-coupling*. This effect is responsible for the splitting of signal lines into multiplets. The coupling between nuclei or pairs of nuclei is mediated by the polarization of electrons or chemical bonds connecting these nuclei. This effect is mutual, meaning that it is the same for pairs of nuclei or groups of nuclei. The magnitude of these interactions is called the *coupling constant J* and is expressed in Hz. As can be seen in figure 5.1 the signal on the right is divided into many smaller signals due to the coupling constant J between groups of nuclei close to each other.

The most important spectroscopic parameter is the intensity of the integral of the spectroscopy line. The intensity of the line is proportional to the molar concentration of the nuclei represented by this line. This parameter is very important to measure the amount of a given compound within a tissue, to determine metabolic profiles in the medical field. The NMR spectrum is therefore a very complex representation of the molecule, because it takes into account the interaction between the nuclei and electrons within the molecule, describing the arrangement they have in three-dimensional space.

5.2 Spectra generation

The goal of VS approaches, as abundantly described in previous sections, is to be able to discriminate active compounds among a large library of compounds. Consequently, in order to be able to test the capabilities of the NMR-Like descriptor in this domain it was necessary to obtain the highest number of target-specific representations. Unfortunately, at the state of the art, there are no public databases of H-NMR spectra and it was necessary to identify reliable simulators to accumulate a sufficient number of molecules to train a DNN.

The first simulator tested was the online tool "predict 1H NMR" [130, 131, 132] available at nmrdb.org and presenting the interface shown in Figure 5.2.


Paste or drop a molfile or SMILES
Chemical structure with hydrogen exploded
Drop or paste a jcamp file
Click to superimpose
Nucl...
Solvent
Expe...
Filen...
Text
C Si
N P
O S
F Cl
Br I
H ?...
Simulate spectrum
1H NMR spectra - Double click to zoom out, SHIFT + double click to zoom out by step, SHIFT + drag to move spectrum
5
2
3
10
8
6
4
2
0
δ (ppm)
predictedSpectrum

List of signals

Assignment	Delta	Nb atoms	Mult	J (Hz)
15,16,17	0.992	3	t	7.39
13,14	2.653	2	q	7.39
8,12	7.162	2	dddd	7.76 1.29 0.98 0.51
10	7.196	1	tt	7.72 1.29
9,11	7.26	2	tdd	7.74 1.60 0.51

^{1}H NMR: δ 0.99 (3H, t, *J* = 7.4 Hz), 2.65 (2H, q, *J* = 7.4 Hz), 7.10-7.32 (5H, 7.16 (dddd, *J* = 7.8, 1.3, 1.0, 0.5 Hz), 7.20 (tt, *J* = 7.7, 1.3 Hz), 7.26 (tdd, *J* = 7.7, 1.6, 0.5 Hz)).

Advanced options

Spectrometer frequency: 400 MHz
From: 0 ppm
To: 10 ppm
Line width: 1 Hz
Number points: 64k

Figure 5.2: Web user interface of the nmrdb tool.

The tool also presented web services that made it easier to produce H-NMR spectra. Unfortunately, the simulated spectrum showed irregularities, in particular chemical groups containing very electronegative atoms (e.g., $-OH$, $-NH_2$), were not predicted thus losing fundamental chemical information.

The second choice fell on the ChemAxon tool [133], an advanced chemical editor, with API for all operating systems and which can also be used from the command line. The latter feature greatly speeded up the prediction of spectra allowing to iterate the predictive procedure, managing to obtain ∼1000 spectra/h on Single-Core operations. The predicted spectra have been carefully checked and validated by experts of the chemistry computation group of "Fondazione Ri.MED". The simulator is very accurate so as to consider the diasterotopic protons differentiated and the pairs $H - H$, $H - F$ and $C - F$ are represented with an approximation of the first order.

The tool uses canonical SMILES, from which it reconstructs the two-dimensional graph that will be used for H-NMR prediction and has been set with the following parameters:

- Spin-Spin Coupling: On
- Implicit Hydrogen Mode: On
- NMR Prediction Frequency: 500 MHz
- Spectrum Display: Realistic Spectrum
- Spectrum Labels: Chemical Shifts
- Measurement Unit: ppm
- Integral Curve: On

The chemical shifts are estimated by a mixed HOSE- and linear model based on a topological description scheme and are in relation to the chemical shift of tetramethylsilane. The aforementioned spectrum is saved in the JDX-CAMP format, a standard for NMR data analysis, recommended by the IUPAC [134]. The JDX-CAMP file consists of two sections, the CORE and NOTES, which will respectively contain Global information (e.g. Power in MHz of the machinery, compound used for the standardization in the chemical shift δ, etc.) and the information NMR Datatype specific, that is the spectrum, which will be represented as a vector of 8192 bit [135].

An example of a predicted H-NMR spectrum is shown in Figure 5.3.

5.3 Dimensional Exploration of NMR-Like

Having defined and validated the predictor to generate in silico H-NMR spectra, I set about studying the structure of each one. The spectra are presented as 8192bit vectors with a sparsity that varies from molecule to molecule in a range of 40-80%. This feature of the data can give problems during the training phases of neural networks, for example, increasing the bias of convolutional operations. At this stage of the study I was concerned with finding a version of the spectrum that would retain the chemical information while reducing the overall sparsity of the representation. To be able to do this, knowledge of the

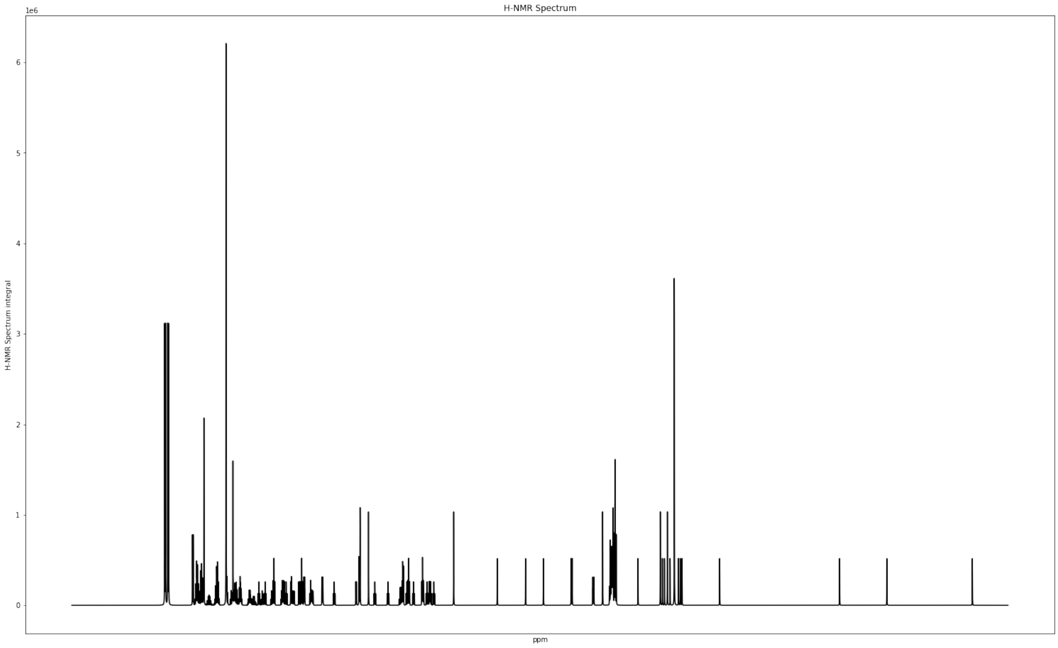


Figure 5.3: H-NMR predicted by ChemAxon's MarvinSketch simulator.

data was critical. The peaks of the H-NMR spectrum represent the various chemical groups that make up the molecule, especially the shape and intensity of the peak also give us information on the three-dimensional arrangement that the nuclei have in space. We can therefore assume that each peak in the scale is influenced by the peak that precedes it in the scale and the whole spectrum can be idealized as a time series. Based on this assumption, I furthered my study of data reduction algorithms applied to data mining [136].

The first that has been tested is the Piecewise Aggregate Approximation (PAA). This method divides the time-series into intervals of size k (bin) denoted by $[t_1, t_k], [t_{k+1}, t_{2k}], etc.$ The bins will be of the same size and each of them will contain the same number of points. Then the new binned value will be y'_{i+1} where (5.3) [136].

$$y'_{i+1} = \frac{\sum_{r=1}^{k} y_{i \cdot k + r}}{k} \tag{5.3}$$

The use of this technique will provide a compressed representation of the H-NMR spectrum that as can be seen in Figure 2 does not lose any information despite the reduction to 1024 bin. Further reductions deform excessively the shape of the spectrum, making lose important structural information, as can be in figure 5.4.

It is also possible to use the median rather than the mean of the values for each bin. Typically, the median provides a more robust estimate because the

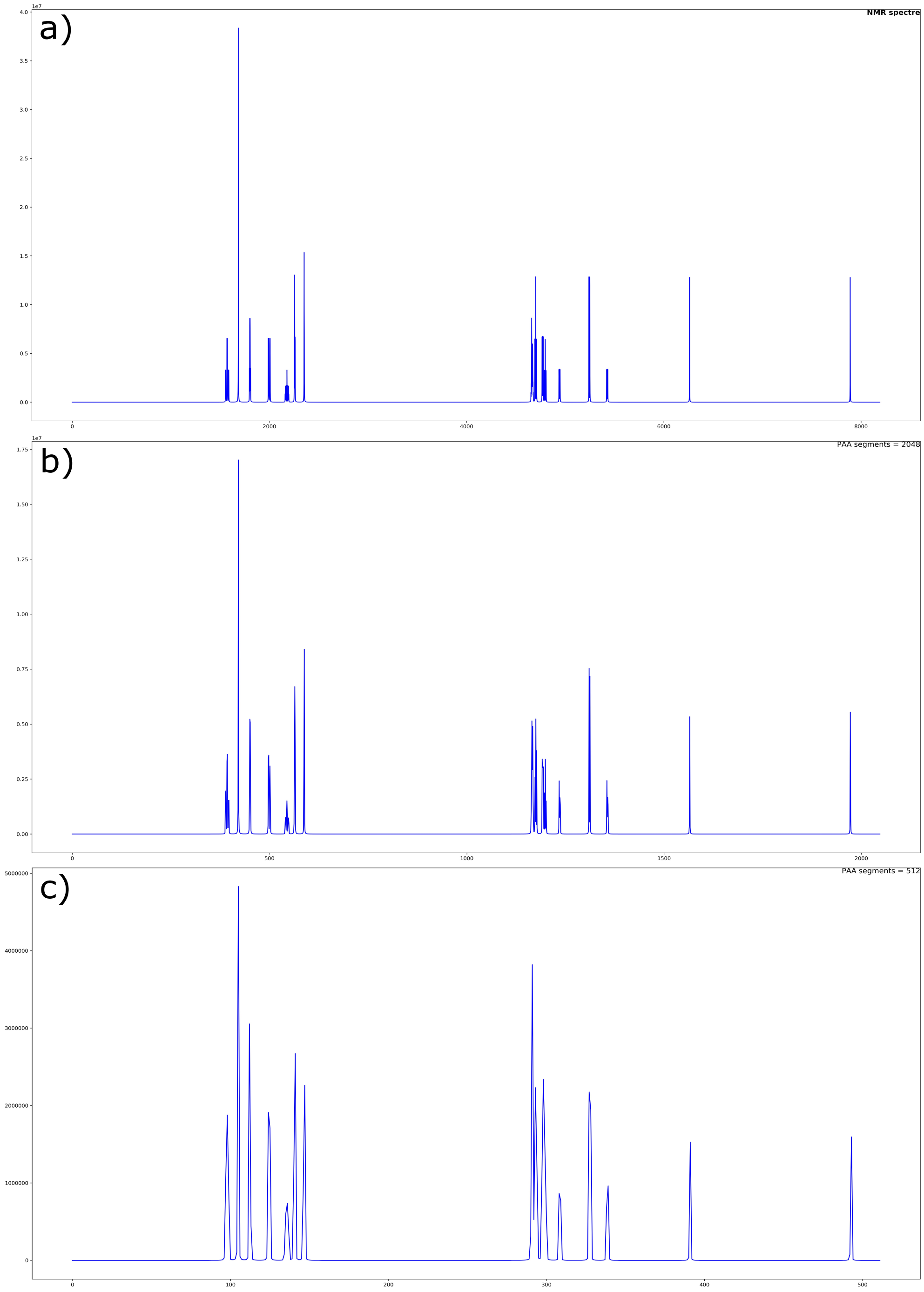


Figure 5.4: Plot of H-NMR spectra after manipulation with PAA. a) Original H-NMR spectrum (8192bit), b) 2048bit H-NMR spectrum, c) 512bit H-NMR spectrumt.

outlier points do not disproportionately affect the median. For this reason the spectra were also tested after compression using PAA based on the median.

A further data compression algorithm tested was the Wavelet Transform, a preprocessing method that converts a signal or a time series into a multidimensional data set in which temporal continuity is ignored. In the wavelet, the coefficients describe the contiguous time regions of the series, providing a coefficient that is equal to half the difference in mean value between a pair of contiguous, carefully chosen segments of the series. Also, as the number of coefficients retained is much smaller than the length of the time series itself, a reduction in dimensionality is still achieved [136]. The wavelet transforms used are part of the Discrete Wavelet Transform (DWT) family which use a discrete subset of all possible values. Specifically, the wavelets used are Daubechies wavelets, based on the work of Ingrid Daubechies, a family of orthogonal wavelets that define a discrete wavelet transform and are characterised by a maximum number of escape moments for a given medium. With each type of wavelet in this class, there is a scaling function (called the parent wavelet) that generates an orthogonal multiresolution analysis.

Today, wavelet transforms are used in the analysis and encoding of a large number of different types of data, such as images, heartbeat and ECG analysis, DNA analysis and protein analysis. Given the nature of the data obtained with the ChemAxon simulator [133] the transform appears to be ideal for testing an additional compressed structure.

The experiments conducted in this phase were performed using dataset 2 (545 active molecules, 4907 inactive on CDK1, see 3.2.1) in order to identify the ideal embedding size. The assays were conducted using different types of deep neural networks. Architectures with Depth Separable Convolution layer, with Convolutional layers, with GRU recurrent layers specifically have been tested. The common factor of all architectures was the low number of parameters needed for training and the small number of layers needed to extract features. A schematic representation of two of the neural architectures used is shown in Figure 5.5.

Once the most suitable architectures were identified, all combinations of PAA (length, binning type) and DWT were tested using all DNNs. This step was important to be able to isolate the most performing embedding in the classification of molecules active on CDK1. The results obtained from this research are reported in Table 5.1. The table reports information regarding the type of architecture used, the type of data compression applied and the metrics of Accuracy, Loss, bACC, Sensitivity, F1-Score and AUC.

Sensitivity was the main evaluation criterion, because it is the metric that shows the ability to correctly discriminate active molecules from inactive ones, assisted by the value of bACC. NMR-Like obtained with the Piecewise Aggre-

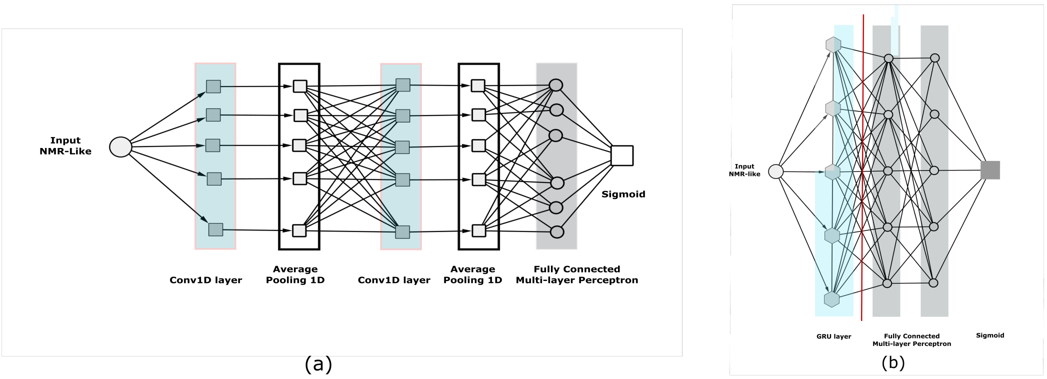


Figure 5.5: Schematic representation of the CNN (a) and GRU (b) architectures used.

Table 5.1: Results of exploratory NMR-Like dimension analysis. Results of exploratory NMR-Like dimension analysis. The best result is shown in bold.

NN*	Data Compression	Accuracy	Loss	Bal. accuracy	Sensitivity	F1-score	AUC
DSC	Wavelet DB1	0.7018	1.0885	0.7527	0.8193	0.3942	0.8323
CNN	Wavelet DB1	0.9087	0.2562	0.7761	0.6024	0.6098	0.8861
RNN	PAA median 2048bin	0.8630	0.5460	0.7868	0.6867	0.5428	0.8548
DSC	**PAA median 1024bin**	**0.7590**	**0.9113**	**0.8215**	**0.9036**	**0.4702**	**0.9018**
CNN	PAA media 1024bin	0.9400	0.1943	0.8043	0.6265	0.7246	0.9245

*Neural Network

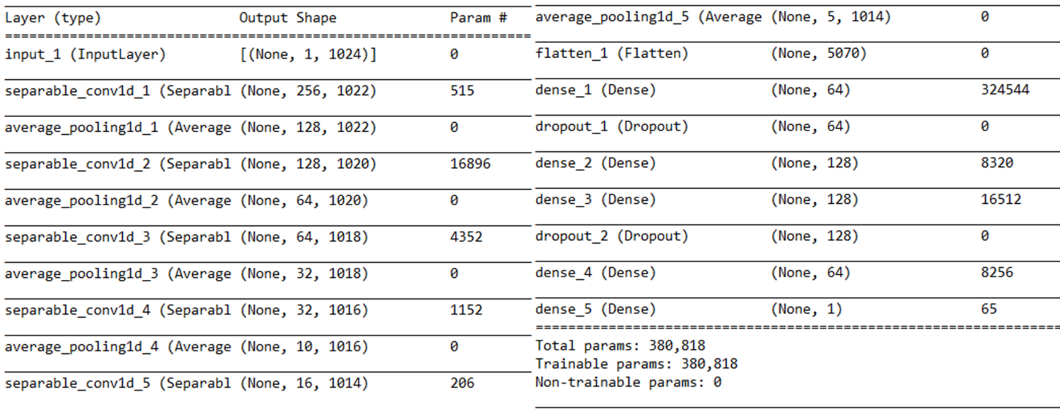
gate Approximation with 1024bin calculated using the median, presents itself as the best representation among those tested, in combo with the DSC architecture and was used as embedding of NMR-Like for the experiments reported in the next sections.

5.4 Classification on multiple targets

The next step aims to analyze the performance of NMR-Like. This step aims to validate the new descriptor in a real application domain. The sets of compounds used in this step of the study are 4: the dataset 2 [insert ref dataset 2] and three subdatasets extrapolated from dataset 3 [insert ref dataset 3]. A summary of the active molecules is shown in Table 5.2.

Table 5.2: Summary of actives and inactives used in the four datasets tested

Protein	Dataset	Tanimoto similarity	Active	Inactive
CDK1	Dataset 2	1	869	6278
JAK2	Dataset 3	0.87	5526	55260
INSR	Dataset 3	0.86	1423	14230
CLK2	Dataset 3	0.82	671	7040



```
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 1, 1024)]         0

separable_conv1d_1 (Separabl (None, 256, 1022)         515

average_pooling1d_1 (Average (None, 128, 1022)         0

separable_conv1d_2 (Separabl (None, 128, 1020)         16896

average_pooling1d_2 (Average (None, 64, 1020)          0

separable_conv1d_3 (Separabl (None, 64, 1018)          4352

average_pooling1d_3 (Average (None, 32, 1018)          0

separable_conv1d_4 (Separabl (None, 32, 1016)          1152

average_pooling1d_4 (Average (None, 10, 1016)          0

separable_conv1d_5 (Separabl (None, 16, 1014)          206

average_pooling1d_5 (Average (None, 5, 1014)           0

flatten_1 (Flatten)          (None, 5070)              0

dense_1 (Dense)              (None, 64)                324544

dropout_1 (Dropout)          (None, 64)                0

dense_2 (Dense)              (None, 128)               8320

dense_3 (Dense)              (None, 128)               16512

dropout_2 (Dropout)          (None, 128)               0

dense_4 (Dense)              (None, 64)                8256

dense_5 (Dense)              (None, 1)                 65
=================================================================
Total params: 380,818
Trainable params: 380,818
Non-trainable params: 0
```

Figure 5.6: Model summary

A DSC architecture with 5-layer Depth Separable Convolution PReLU with 256, 128, 64, 32, 16 filters for feature extraction with two Average Pooling (One after the second DSC layer and one before the classification block) and a classification block consisting of 4 MLP layers with 64, 128, 64 ReLU units. The architecture is not very deep and has only 380.818 parameters, as shown in Figure 5.6.

All datasets used were divided into Training, Validation, and Test sets with an 80:10:10 ratio, and an active to inactive ratio of 1:10 for each. The Validation set was used to perform hyperparameter tuning for each of the DNNs. The results for each of the targets are shown in Table 5.3. The metrics used are global accuracy, Loss, Sensitivity, MCC, AUC ROC-curve, F1-score.

Table 5.3: Results for the active/inactive discrimination task, and Training scheme 1. Best/worst values for each column are in bold/italic.

Protein	Accuracy	Loss	Sensitivity	MCC	AUC	F1-score
CDK1	*0.7846*	0.5831	0.8795	0.4570	*0.9058*	0.4916
CLK2	0.8031	*0.6842*	**0.9254**	*0.4542*	0.9223	*0.4493*
INSR	0.9284	0.2068	*0.7324*	0.6155	0.9268	0.6500
JAK2	**0.9447**	**0.1833**	0.7848	**0.6933**	**0.9473**	**0.7209**

The results shown in Table 5.3 are very encouraging in terms of *Sensitivity* and demonstrate how the overall performance increases proportionally to the number of samples assayed. In fact, JAK2, the protein with the most abundant dataset, is the target on which the best values are recorded, except for Sensitivity. This result comes from the high number of active molecules that are tested. It is enough to look at the TP/P value shown in Table 5.4 to observe how despite the Sensitivity dataset is low, the JAK2 classifier manages to prioritize the actives well. Table 5.4 shows the results of the ratio of True Positive over Positive (TP/P) and Enrichment Factor (EF, see equation 4.4) for all 4 targets.

Table 5.4: True Positives versus Positives ratio and Enrichment Factors computed on the entire test set.

Protein	TP/P 1%	TP/P 2%	TP/P 5%	TP/P 10%	EF 1%	EF 2%	EF 5%	EF 10%
CDK1[1]	7/83	13/83	27/83	47/83	8	8	7	6
CLK2[2]	6/67	10/67	23/67	41/67	9	7	7	6
INSR[3]	16/142	30/142	61/142	98/142	11	11	9	7
JAK2[4]	61/553	120/553	277/553	417/553	11	11	10	8

Percentage relative to the evaluated test set evaluated (13400 compounds), [1] 1% = 7 molecules, [2] 1% = 8 molecules, [3] 1% = 16 molecules, [4] 1% = 61 molecules.

The EF values, although not as high as found with the classifier that used EMBER, are still satisfactory. Typically, an algorithm used in the VS is called good when its EF value > 5, which is the case for all 4 models [125].

In conclusion, NMR-Like proves to be a high-performance numerical embedding for bioactivity classification, capable of highly accurate prioritization of highly active molecules, while keeping the chemical information it carries accessible. This is the aspect that I went to study in the final stages of NMR Like analysis through *Explainable AI* approaches.

5.5 Explainable AI

After determining the efficiency of descriptor classification, to increase knowledge about the features that determine classification I conducted explainability experiments on each of the best trained models for each of the targets in order to identify the canonical feature pattern for the active molecules. The DeepLIFT-based approach of the well-known SHAP framework was used to perform the analysis. In order to perform a useful analysis from the application point of view, the SHAP values were not calculated on the whole test set but only on the molecules that were prioritized as most active by the neural network. In this way, I hypothesized to be able to isolate all the characteristic features of these active molecules, describing the canonical pattern of active molecules. Below, the summary plot of the shap value obtained from the molecules prioritized on CDK1 is shown (Figure 5.7).

In Figure 5.7, each value associated with the "Feature" represents the position in the 1024bin NMR-Like vector, allowing us to isolate the most relevant regions of the spectrum by identifying the chemical groups characterizing the

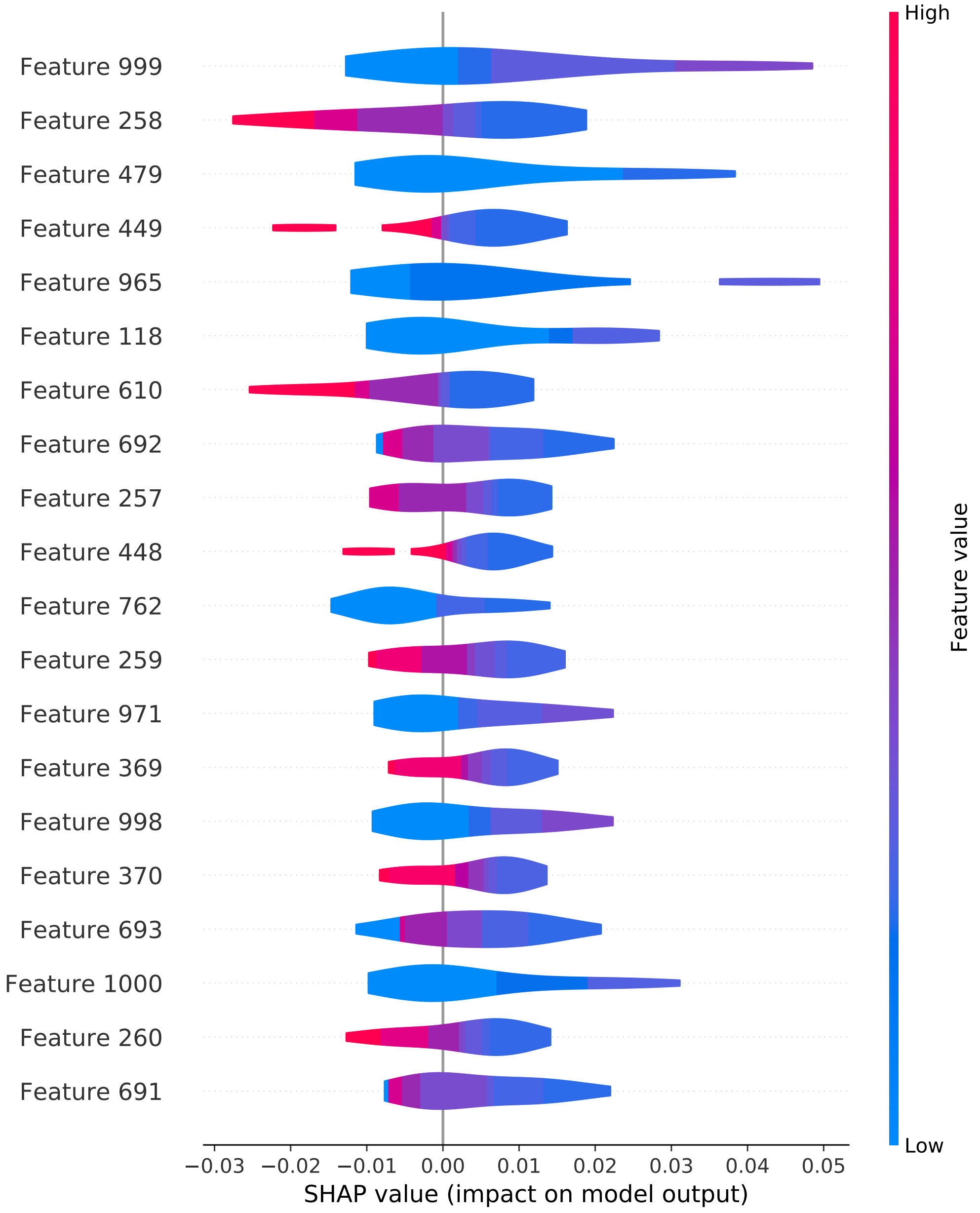


Figure 5.7: Representative image of the most impactful features on classification, calculated on 1% of the test set for CDK1 protein.

active molecule. This global representation provides us with a view of the pattern common to all the prioritized molecules but the analysis can be deepened on each of the molecules, isolating the SHAP values most relevant for the classification (see Figure 5.8) in order to map the peaks of the H-NMR spectrum as shown in Figure 5.9.

The opportunity to analyze also from a purely chemical point of view the results obtained from the classification, makes the NMR-Like embedding an innovative tool for the screening phases. The scope of the descriptor could

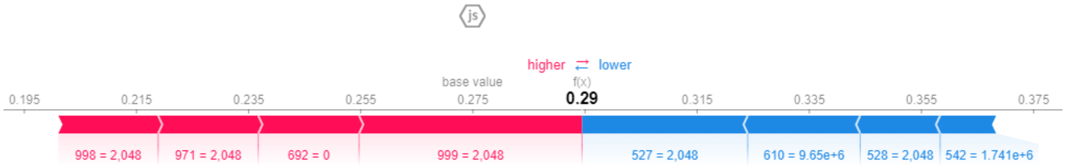


Figure 5.8: Example of how the most relevant SHAP values were calculated for each individual molecule.

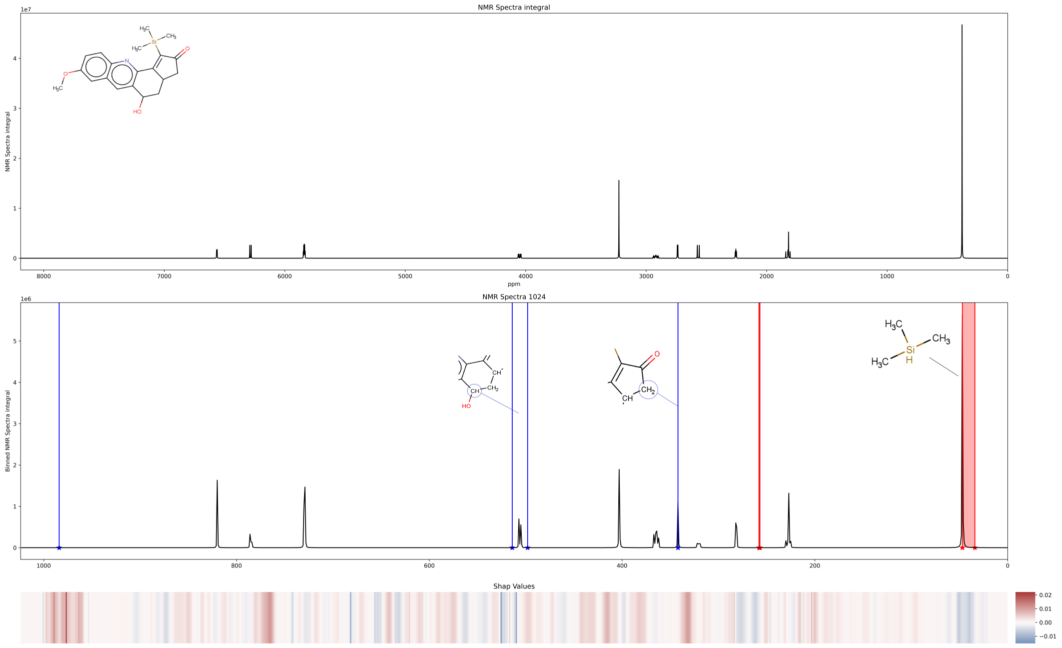


Figure 5.9: Assignment of substituent groups identified through the most relevant SHAP values. In red we find the values that promote the classification, while in blue those that worsen the performance. a) the original H-NMR spectrum, b) the H-NMR spectrum in its compressed form at 1024 bin, c) heatmap of SHAP values.

also be extended to Structure-Based VS applications, as a supporting element to Molecular Docking approaches both in the HTVS phase and in the high priority screening phase.

5.6 Drug Repurposing application

The experiments that will be described in this section were conducted within the CLAIRE (Confederation of Laboratories for Artificial Intelligence in Europe) Task Force against COVID-19, in the context of Drug Repurposing [insert reference 7 Year II report](i.e. the search for existing drugs for new therapeutic purposes). The use of Artificial Intelligence approaches in Drug Repurposing, could contribute in the identification phases of new drugs for the treatment of SARS-CoV-2 infection. The speed with which the pandemic swept the world and the virulence that the strain presented, necessitated a prompt response that unfortunately could not be achieved with traditional approaches. Computational chemistry approaches and techniques such as Deep Learning were key tools to try to counteract the rate at which the virus was killing victims around the world. The CLAIRE Task Force has tried to make its contribution in this battle against the virus. With the Human-Computer Interaction Lab, we have been working on the use of chemical descriptors such as Molecular Fingerprints and NMR-Like descriptors, to be able to build a classifier able to identify active drugs against the infection.

The experiments were conducted using Dataset 4 (see 3.2.3), consisting of 213 active and 940 inactive drugs against SARS-CoV-2 infection. In Drug Repositioning studies, the number of compounds available to the researcher is not comparable to those in Virtual Screening studies, unfortunately hindering Deep Learning and Machine Learning algorithms from expressing their full potential.

NMR-Like and Molecular Fingeprints, were used as inputs to the deep neural networks used. Specifically CNN and DSC architectures, both one and two dimensional were used. The Molecular Fingeprints, used are RDKit, Morgan, AtomPair, Torsion, Layered, FeatMorgan, ECFP4 and were tested both individually and as EMBER. During the time we were addressing this issue, EMBER was still in the early stages of its testing, and was tested both in its final configuration (7 x 1024 x 1), and in several variants formed by a changing number of Fingerprints. Finally, the 7-channel version was tested, using a dual-input Depth Separable Convolution architecture, in combination with NMR-Like with the intention of integrating the information of the multispectral representation with that of the H-NMR spectrum.

The results obtained from the study are shown in Table 5.5.

As can be seen, the best results were provided by EMBER even if they are not satisfactory to define the study of Drug Repurposing as complete.

Table 5.5: Results obtained by DrugBank's approved drug classifier.

Molecular Descriptor	Accuracy	Loss	Bal. Accuracy	Sentivity	Roc-Curve
DSC 1D(E)	0.7241	0.4670	0.5825	0.3478	0.7295
EMBER	0.7192	0.5303	0.7105	0.6956	0.7345
NMR-Like	0.6930	0.6480	0.6452	0.5652	0.6490
EMBER-NMR-Like	0.7105	0.4771	0.6562	0.5652	0.7618
DSC 2D (T-E)	0.7672	0.4796	0.6912	0.5652	0.7480
DSC 2D (R-T-F)	0.7672	0.4591	0.7076	0.6087	0.7492
DSC 2D (A-T-F-E)	0.7576	0.4974	0.7021	0.6087	0.7359
DSC 2D (R-M-A-T-F)	0.7500	0.4950	0.6968	0.6067	0.7802

Fingerprint types: *R*DKit, *M*organ, *A*tomPair,*(T)orsion*, *(L)ayered*, *(F)eatMorgan*, *(E)CFP4*

Unfortunately, the low number of positive samples (approved drugs), does not allow the neural network to identify the feature pattern to classify efficiently the known drugs. This degree of uncertainty did not allow us to test the classifier obtained on the entire set of approved drugs downloaded from DrugBank. During the writing phases of this dissertation, further experiments are carried out, in order to increase the number of drugs available in the training phases and to obtain a high performance classifier to be tested on the complete test set obtained from DrugBank.

Chapter 6

Conclusion

The creation of a new drug is a very complex process that requires a considerable amount of time and money. The Development phases include in vivo experiments and clinical trials that must follow a rigorous process and cannot be rushed. The research community is investing time and resources to revolutionize the Discovery approach of the lead compound, thanks to the technologies provided by computer science. The phase that is mainly benefiting from all this is Virtual Screening, the domain within which my activity took place. The research activity conducted during this PhD course aims to understand and test the impact of molecular descriptors in the domain of interest. The thesis is presented as a theoretical and applicative description of the most used numerical descriptors in the state of the art, the Molecular Fingerprints. The latter are vectors of fixed length that describe the molecular structure in a different way as the kernel with which they are generated varies. In the proposed study we examine 7 of them (RDKit, Morgan, AtomPair, Torsion, Layered, FeatMorgan and ECFP4), selected for the different encoding of chemical information within them.

In accordance with the initial proposal, a first experimental evaluation allowed us to identify the most suitable fingerprint size for the classification task, showing the 1024bit embedding as the most efficient one. The complementarity of the chemical information present in each of the fingerprints correlated to the computational efficiency that they presented has allowed us to create the first embedding proposed as a final result of this work, EMBER (embedding multiple molecular fingerprints) which integrates the information expressed by the individual fingerprints in a multispectral representation of the molecule. As has been shown, it allows multi-class multi-target classification to be performed on a large number of small compounds using a deep neural network with a relatively small number of parameters. Explainability experiments have shown that embedding is not fixed and that not all Molecular Fingeprints are needed to transfer chemical information. As shown in the figure 4.6, FeatMorgan, Layered and RDKit show more influence when compared to the others. I tried to rationalize this observation based on their composition.

FeatMorgan is an FCFP circular fingerprint where the molecule is characterized by the functional description of the atoms directly involved in the interaction with the binding site (e.g., acceptor and hydrogen donor groups, polarity, aromaticity and so on). Probably, such a type of classification, not based simply on the chemical pathway, but on the ability of the ligand to bind specific protein residues, performs better than the simple ECFP circular fingerprint, only with respect to atomic type pathways. RDKIT and Layered fingerprint are both based on substructure decomposition (e.g., aromatic rings). In a recently published work by Zhu et al. [137] the authors performed a chemoinformatics analysis of 2139 protein kinase inhibitors and found most of these molecules to be "flat" with a very low fraction of sp^3 carbons and a high number of aromatic rings. From the study, it was also shown that the weighted average number of hydrogen bonds was inversely proportional to the number of aromatic rings. In detail, it appears that in the binding affinity to protein kinases, there is a correlated offset between H-bond interactions and aromatic and non-bond interactions. This inverse relationship strongly suggests the importance of the balanced presence of donors and acceptors of hydrogen bonds and aromatic moieties within the ligand for the molecular recognition of protein kinase inhibitors.

From this study it emerges that FCFP, RDKIT and Layered that fingeprints containing information on the pharmacophoric role of individual chemical groups, perform better than the others in this specific task. The use of Explainability allows then to perform a tuning of the descriptor, reducing the size of the Tensor, allowing to use a combination of Fingerprints that best suits the type of classification that you want to pursue.

Despite its excellent performance in classification, EMBER, as well as the structural units of which it is composed, showed its greatest limitation during the Explainability phases, in which the impossibility of interpreting the chemical information that influences the bioactivity classification was highlighted. This is due to the hashing algorithm that is used for their generation and that in communion with the collision bit events prevents the technician from tracing the chemical group that each bit 1 represents. Based on these observations, my research activity has evolved towards the creation of a new molecular descriptor. NMR-Like aims to integrate the VS Structure-based steps with a Ligand-based approach, allowing the user to have complete control of the most influential features in the classification. This feature overcomes the limit of Molecular Fingeprints allowing the cheminformatic operator to instantly recognize the chemical groups common to all active molecules, providing a tool of immediate application for the synthetic chemist, who during the characterization phases of the new molecule, performed with nuclear magnetic resonance spectroscopy, could know in addition to the structure, also the target on which the new molecule is active. The ability to classify bioactivity, with the use of very small DNN is a further strength of the proposed descriptor. It succeeds in classifying with satisfactory Enrichment Factor and TP/P values.

The lack of a centralized database of target-specific H-NMR spectra made the development of this new embedding and approach complex, which at the state of the art remains unique for classification tasks in the VS. The overall performance and reliability of the descriptor could be increased by using real H-NMR spectra, an issue mainly encountered during the training phases of the neural network. With the computational chemistry and medicinal chemistry group of Ri.MED foundation, work is continuing to succeed in definitively validating the simulated spectra.

Acknowledgements

The first person I would like to thank is my Tutor Prof. Roberto Pirrone, for everything he has done for me. For his patience, wisdom and lightness that allowed me to overcome even the most complex challenges. Since the first day he has encouraged me to give my best, leaving me free to make mistakes, showing me every time a better way to grow. I would like to thank him for the trust he continues to give me today, I could not hope for a better guide. I hope we can work together for many more years. **Thanks**. The second person I would like to thank is my co-tutor Ugo Perricone, for providing support, expertise by endorsing the ideas that a young PhD student sets out to pursue. **Thanks**. A thought goes to my companion in adventure, Ing. Isabella Mendolia with whom I have shared since day one the experiences that this journey has reserved for us. We have worked closely together, ever since the "for cycles" were my nightmare, comparing, clashing and learning, in order and in disorder, how to approach this doctoral path. Probably, these 3 years would have been much more complex if she had not been there to share this journey with me. **Thanks**. I would like to thank my parents and sister for their constant presence, psychological and moral support. You are the main source of my energy, the reservoir of my love and inexhaustible source of stimulation. All my efforts in completing this path stem from the example you have given me. **Thanks**. Finally, a thought goes out to my Grandpa Nino, with whom I would have loved to comment on every single moment of this trip but who unfortunately is no more with us today. I'm sure he will be proud of me and I hope to make him happy wherever he is. **Thanks**.

Publications

Parts of the work in this thesis have been published in

INTERNATIONAL CONFERENCES AND WORKSHOPS

- Mendolia, I., **Contino, S.**, Perricone, U., Pirrone, R., Ardizzone, E., 2019. A Convolutional Neural Network for Virtual Screening of Molecular Fingerprints, in: Ricci, E., Rota Bulò, S., Snoek, C., Lanz, O., Messelodi, S., Sebe, N. (Eds.), Image Analysis and Processing – ICIAP 2019. Springer International Publishing, Cham, pp. 399–409. https://doi.org/10.1007/978-3-030-30642-736.

- Mendolia, I., **Contino, S.**, Perricone, U., Ardizzone, E., Pirrone, R. Deep Neural Networks for Virtual Screening. 1st International Workshop on Artificial Intelligence for Health (AIxHealth 2019). Rende (CS), 19-22 novembre 2019. **Oral communication**.

NATIONAL CONFERENCES AND WORKSHOPS

- Mendolia, I., **Contino, S.**, Perricone, U. Pirrone, R. Ardizzone, E. Tecniche di Deep Learning per i Drug Design. Convegno Nazionale CINI sull'intelligenza artificiale, Ital-IA. Roma 18-19 marzo 2019. **Oral communication**.

- Mendolia, I., **Contino, S.**, Perricone, U. Pirrone, R. Ardizzone, E.. Virtual Screening of Molecular Fingerprint through Convolutional Neural Network. Bioinformatics Italian Society Annual Meeting 2019. Palermo 26-28 giugno. **Oral communication**.

- Mendolia, I., Contino, S., Perricone, U. Pirrone, R. Ardizzone, E.. Virtual Screening of Molecular Fingerprint through Convolutional Neural Network. Bioinformatics Italian Society Annual Meeting 2019. Palermo 26-28 giugno. **Oral communication**.

INTERNATIONAL JOURNAL

- Mendolia, I., **Contino, S.**, Perricone, U., Ardizzone, E., Pirrone, R., 2020. Convolutional architectures for virtual screening. BMC Bioinformatics 21, 310. https://doi.org/10.1186/s12859-020-03645-9

- Mendolia, I., **Contino, S.**, De Simone, G., Perricone, U., Pirrone, R., 2022. EMBER—Embedding Multiple Molecular Fingerprints for Virtual Screening. International Journal of Molecular Sciences, 23(4), 2156. https://doi.org/10.3390/ijms23042156

Bibliography

[1] A. Good, *4.19 - Virtual Screening*, p. 459–494. Elsevier, Jan 2007.

[2] S. Pushpakom, F. Iorio, P. A. Eyers, K. J. Escott, S. Hopper, A. Wells, A. Doig, T. Guilliams, J. Latimer, C. McNamee, A. Norris, P. Sanseau, D. Cavalla, and M. Pirmohamed, "Drug repurposing: progress, challenges and recommendations," *Nature Reviews Drug Discovery*, vol. 18, p. 41–58, Jan 2019.

[3] L. David, J. Arús-Pous, J. Karlsson, O. Engkvist, E. J. Bjerrum, T. Kogej, J. M. Kriegl, B. Beck, and H. Chen, "Applications of deep-learning in exploiting large-scale and heterogeneous compound data in industrial pharmaceutical research," *Frontiers in Pharmacology*, vol. 10, p. 1303, Nov 2019.

[4] L. David, A. Thakkar, R. Mercado, and O. Engkvist, "Molecular representations in ai-driven drug discovery: a review and practical guide," *Journal of Cheminformatics*, vol. 12, p. 56, Sep 2020.

[5] M. Malumbres, E. Harlow, T. Hunt, T. Hunter, J. M. Lahti, G. Manning, D. O. Morgan, L.-H. Tsai, and D. J. Wolgemuth, "Cyclin-dependent kinases: a family portrait," *Nature Cell Biology*, vol. 11, p. 1275–1276, Nov 2009.

[6] "My Cancer Genome." `https://www.mycancergenome.org/content/pathways/cell-cycle-control/`.

[7] J. Jiménez-Luna, F. Grisoni, and G. Schneider, "Drug discovery with explainable artificial intelligence," *Nature Machine Intelligence*, vol. 2, pp. 573–584, Oct. 2020. Number: 10 Publisher: Nature Publishing Group.

[8] J. DiMasi, R. Hansen, and H. Grabowski, "The price of innovation: New estimates of drug development costs," vol. 22, pp. 151–85.

[9] J. R. Turner, *New Drug Development*, p. 1–10. Springer New York, 2010.

[10] J. R. Turner, *New drug development: an introduction to clinical trials.* Statistics in practice, Springer, 2nd ed ed., 2010.

[11] D. K. Agrafiotis, M. K. Holloway, S. A. Johnson, C. H. Reynolds, T. R. Stouch, A. Tropsha, and C. L. Waller, "Chemistry, information and frank: a tribute to frank brown," *Journal of Computer-Aided Molecular Design*, vol. 32, p. 723–729, Jul 2018.

[12] K. Martinez-Mayorga, A. Madariaga-Mazon, J. L. Medina-Franco, and G. Maggiora, "The impact of chemoinformatics on drug discovery in the pharmaceutical industry," *Expert Opinion on Drug Discovery*, vol. 15, p. 293–306, Mar 2020.

[13] R. Duelen, M. Corvelyn, I. Tortorella, L. Leonardi, Y. C. Chai, and M. Sampaolesi, *Medicinal Biotechnology for Disease Modeling, Clinical Therapy, and Drug Discovery and Development*, p. 89–128. Springer International Publishing, 2019.

[14] J. G. Moffat, F. Vincent, J. A. Lee, J. Eder, and M. Prunotto, "Opportunities and challenges in phenotypic drug discovery: an industry perspective," vol. 16, p. 531–543, Aug 2017.

[15] S. Saeidnia, A. R. Gohari, and A. Manayi, "Reverse pharmacognosy and reverse pharmacology; two closely related approaches for drug discovery development," *Current Pharmaceutical Biotechnology*, vol. 17, no. 11, p. 1016–1022, 2016.

[16] C. Morrison, "Boom: 2018's biotech ipos," *Nature Reviews Drug Discovery*, vol. 18, p. 3–6, Jan 2019.

[17] O. o. t. Commissioner, "U.s. food and drug administration," May 2021.

[18] I. Bighelli and C. Barbui, "What is the european medicines agency?," *Epidemiology and Psychiatric Sciences*, vol. 21, p. 245–247, Sep 2012.

[19] A. Hamza, N.-N. Wei, and C.-G. Zhan, "Ligand-based virtual screening approach using a new scoring function," *Journal of chemical information and modeling*, vol. 52, p. 963–974, Apr 2012.

[20] M. Lill, "Virtual screening in drug design," *Methods in Molecular Biology (Clifton, N.J.)*, vol. 993, p. 1–12, 2013.

[21] Y. Baoyu, M. Jing, G. Bing, and L. Xiuli, “Computer-assisted drug virtual screening based on the natural product databases,” *Current Pharmaceutical Biotechnology*, vol. 20, p. 293–301, Feb 2019.

[22] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne, “The Protein Data Bank,” *Nucleic Acids Research*, vol. 28, pp. 235–242, Jan. 2000.

[23] A. T. Brint and P. Willett, “Algorithms for the identification of three-dimensional maximal common substructures,” *Journal of Chemical Information and Computer Sciences*, vol. 27, p. 152–158, Nov 1987.

[24] C. Bron and J. Kerbosch, “Algorithm 457: finding all cliques of an undirected graph,” *Communications of the ACM*, vol. 16, p. 575–577, Sep 1973.

[25] T. J. A. Ewing and I. D. Kuntz, “Critical evaluation of search algorithms for automated molecular docking and database screening,” *Journal of Computational Chemistry*, vol. 18, no. 9, p. 1175–1189, 1997.

[26] I. D. Kuntz, J. M. Blaney, S. J. Oatley, R. Langridge, and T. E. Ferrin, “A geometric approach to macromolecule-ligand interactions,” *Journal of Molecular Biology*, vol. 161, p. 269–288, Oct 1982.

[27] E. F. Pettersen, T. D. Goddard, C. C. Huang, G. S. Couch, D. M. Greenblatt, E. C. Meng, and T. E. Ferrin, “Ucsf chimera?a visualization system for exploratory research and analysis,” *Journal of Computational Chemistry*, vol. 25, p. 1605–1612, Oct 2004.

[28] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, “Equation of state calculations by fast computing machines,” *The Journal of Chemical Physics*, vol. 21, p. 1087–1092, Jun 1953.

[29] R. Abagyan, M. Totrov, and D. Kuznetsov, “Icm—a new method for protein modeling and design: Applications to docking and structure prediction from the distorted native conformation,” *Journal of Computational Chemistry*, vol. 15, no. 5, p. 488–506, 1994.

[30] C. McMartin and R. S. Bohacek, “Qxp: powerful, rapid computer algorithms for structure-based drug design,” *Journal of Computer-Aided Molecular Design*, vol. 11, p. 333–344, Jul 1997.

[31] R. Taylor, P. Jewsbury, and J. Essex, "A review of protein-small molecule docking methods," *Journal of Computer-Aided Molecular Design*, vol. 16, p. 151–166, Mar 2002.

[32] D. S. Goodsell and A. J. Olson, "Automated docking of substrates to proteins by simulated annealing," *Proteins: Structure, Function, and Bioinformatics*, vol. 8, no. 3, p. 195–202, 1990.

[33] G. M. Morris, R. Huey, W. Lindstrom, M. F. Sanner, R. K. Belew, D. S. Goodsell, and A. J. Olson, "Autodock4 and autodocktools4: Automated docking with selective receptor flexibility," *Journal of computational chemistry*, vol. 30, no. 16, pp. 2785–2791, 2009.

[34] G. Jones, P. Willett, R. C. Glen, A. R. Leach, and R. Taylor, "Development and validation of a genetic algorithm for flexible docking11edited by f. e. cohen," *Journal of Molecular Biology*, vol. 267, p. 727–748, Apr 1997.

[35] D. M. Lorber and B. K. Shoichet, "Hierarchical docking of databases of multiple ligand conformations," *Current Topics in Medicinal Chemistry*, vol. 5, p. 739–749, Aug 2005.

[36] D. Joseph-McCarthy, B. E. Thomas, M. Belmarsh, D. Moustakas, and J. C. Alvarez, "Pharmacophore-based molecular docking to account for ligand flexibility," *Proteins*, vol. 51, p. 172–188, May 2003.

[37] C. N. Cavasotto and R. A. Abagyan, "Protein flexibility in ligand docking and virtual screening to protein kinases," *Journal of Molecular Biology*, vol. 337, p. 209–225, Mar 2004.

[38] L. C. Ray and R. A. Kirsch, "Finding chemical records by digital computers," *Science*, vol. 126, no. 3278, p. 814–819, 1957.

[39] P. Willett, J. M. Barnard, and G. M. Downs, "Chemical similarity searching," *Journal of Chemical Information and Computer Sciences*, vol. 38, p. 983–996, Nov 1998.

[40] J. R. Ullmann, "An algorithm for subgraph isomorphism," *Journal of the ACM*, vol. 23, p. 31–42, Jan 1976.

[41] G. M. Downs, M. F. Lynch, P. Willett, G. A. Manson, and G. A. Wilson, “Transputer implementations of chemical substructure searching algorithms,” *Tetrahedron Computer Methodology*, vol. 1, no. 3, p. 207–217, 1988.

[42] *Reviews in Computational Chemistry*. Reviews in Computational Chemistry, John Wiley Sons, Inc., Apr 2017.

[43] S.-K. Lin, “Pharmacophore perception, development and use in drug design. edited by osman f. güner,” *Molecules*, vol. 5, p. 987–989, Jul 2000.

[44] A. C. Good, J. S. Mason, and S. D. Pickett, *Pharmacophore Pattern Application in Virtual Screening. Library Design and QSAR*, p. 131–159. John Wiley Sons, Ltd, 2000.

[45] J. H. Van Drie, D. Weininger, and Y. C. Martin, “Aladdin: an integrated tool for computer-assisted molecular design and pharmacophore recognition from geometric, steric, and substructure searching of three-dimensional molecular structures,” *Journal of Computer-Aided Molecular Design*, vol. 3, p. 225–251, Sep 1989.

[46] Y. C. Martin, M. G. Bures, E. A. Danaher, J. DeLazzer, I. Lico, and P. A. Pavlik, “A fast new approach to pharmacophore mapping and its application to dopaminergic and benzodiazepine agonists,” *Journal of Computer-Aided Molecular Design*, vol. 7, p. 83–102, Feb 1993.

[47] Y. Patel, V. J. Gillet, G. Bravi, and A. R. Leach, “A comparison of the pharmacophore identification programs: Catalyst, disco and gasp,” *Journal of Computer-Aided Molecular Design*, vol. 16, p. 653–681, Aug 2002.

[48] N. Nosengo, “Can you teach old drugs new tricks?,” *Nature*, vol. 534, p. 314–316, Jun 2016.

[49] J. Lamb, E. D. Crawford, D. Peck, J. W. Modell, I. C. Blat, M. J. Wrobel, J. Lerner, J.-P. Brunet, A. Subramanian, K. N. Ross, M. Reich, H. Hieronymus, G. Wei, S. A. Armstrong, S. J. Haggarty, P. A. Clemons, R. Wei, S. A. Carr, E. S. Lander, and T. R. Golub, “The connectivity map: Using gene-expression signatures to connect small molecules, genes, and disease,” *Science*, Sep 2006.

[50] J. C. Venter, M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell, C. A. Evans, R. A. Holt, J. D. Gocayne, P. Amanatides, R. M. Ballew, D. H. Huson, J. R. Wortman, Q. Zhang, C. D. Kodira, X. H. Zheng, L. Chen, M. Skupski, G. Subramanian, P. D. Thomas, J. Zhang, G. L. Gabor Miklos, C. Nelson, S. Broder, A. G. Clark, J. Nadeau, V. A. McKusick, N. Zinder, A. J. Levine, R. J. Roberts, M. Simon, C. Slayman, M. Hunkapiller, R. Bolanos, A. Delcher, I. Dew, D. Fasulo, M. Flanigan, L. Florea, A. Halpern, S. Hannenhalli, S. Kravitz, S. Levy, C. Mobarry, K. Reinert, K. Remington, J. Abu-Threideh, E. Beasley, K. Biddick, V. Bonazzi, R. Brandon, M. Cargill, I. Chandramouliswaran, R. Charlab, K. Chaturvedi, Z. Deng, V. D. Francesco, P. Dunn, K. Eilbeck, C. Evangelista, A. E. Gabrielian, W. Gan, W. Ge, F. Gong, Z. Gu, P. Guan, T. J. Heiman, M. E. Higgins, R.-R. Ji, Z. Ke, K. A. Ketchum, Z. Lai, Y. Lei, Z. Li, J. Li, Y. Liang, X. Lin, F. Lu, G. V. Merkulov, N. Milshina, H. M. Moore, A. K. Naik, V. A. Narayan, B. Neelam, D. Nusskern, D. B. Rusch, S. Salzberg, W. Shao, B. Shue, J. Sun, Z. Y. Wang, A. Wang, X. Wang, J. Wang, M.-H. Wei, R. Wides, C. Xiao, C. Yan, A. Yao, J. Ye, M. Zhan, W. Zhang, H. Zhang, Q. Zhao, L. Zheng, F. Zhong, W. Zhong, S. C. Zhu, S. Zhao, D. Gilbert, S. Baumhueter, G. Spier, C. Carter, A. Cravchik, T. Woodage, F. Ali, H. An, A. Awe, D. Baldwin, H. Baden, M. Barnstead, I. Barrow, K. Beeson, D. Busam, A. Carver, A. Center, M. L. Cheng, L. Curry, S. Danaher, L. Davenport, R. Desilets, S. Dietz, K. Dodson, L. Doup, S. Ferriera, N. Garg, A. Gluecksmann, B. Hart, J. Haynes, C. Haynes, C. Heiner, S. Hladun, D. Hostin, J. Houck, T. Howland, C. Ibegwam, J. Johnson, F. Kalush, L. Kline, S. Koduru, A. Love, F. Mann, D. May, S. McCawley, T. McIntosh, I. McMullen, M. Moy, L. Moy, B. Murphy, K. Nelson, C. Pfannkoch, E. Pratts, V. Puri, H. Qureshi, M. Reardon, R. Rodriguez, Y.-H. Rogers, D. Romblad, B. Ruhfel, R. Scott, C. Sitter, M. Smallwood, E. Stewart, R. Strong, E. Suh, R. Thomas, N. N. Tint, S. Tse, C. Vech, G. Wang, J. Wetter, S. Williams, M. Williams, S. Windsor, E. Winn-Deen, K. Wolfe, J. Zaveri, K. Zaveri, J. F. Abril, R. Guigó, M. J. Campbell, K. V. Sjolander, B. Karlak, A. Kejariwal, H. Mi, B. Lazareva, T. Hatton, A. Narechania, K. Diemer, A. Muruganujan, N. Guo, S. Sato, V. Bafna, S. Istrail, R. Lippert, R. Schwartz, B. Walenz, S. Yooseph, D. Allen, A. Basu, J. Baxendale, L. Blick, M. Caminha, J. Carnes-Stine, P. Caulk, Y.-H. Chiang,

M. Coyne, C. Dahlke, A. D. Mays, M. Dombroski, M. Donnelly, D. Ely, S. Esparham, C. Fosler, H. Gire, S. Glanowski, K. Glasser, A. Glodek, M. Gorokhov, K. Graham, B. Gropman, M. Harris, J. Heil, S. Henderson, J. Hoover, D. Jennings, C. Jordan, J. Jordan, J. Kasha, L. Kagan, C. Kraft, A. Levitsky, M. Lewis, X. Liu, J. Lopez, D. Ma, W. Majoros, J. McDaniel, S. Murphy, M. Newman, T. Nguyen, N. Nguyen, M. Nodell, S. Pan, J. Peck, M. Peterson, W. Rowe, R. Sanders, J. Scott, M. Simpson, T. Smith, A. Sprague, T. Stockwell, R. Turner, E. Venter, M. Wang, M. Wen, D. Wu, M. Wu, A. Xia, A. Zandieh, and X. Zhu, "The sequence of the human genome," *Science*, vol. 291, p. 1304–1351, Feb 2001.

[51] E. S. Lander, L. M. Linton, B. Birren, C. Nusbaum, M. C. Zody, J. Baldwin, K. Devon, K. Dewar, M. Doyle, W. FitzHugh, R. Funke, D. Gage, K. Harris, A. Heaford, J. Howland, L. Kann, J. Lehoczky, R. LeVine, P. McEwan, K. McKernan, J. Meldrim, J. P. Mesirov, C. Miranda, W. Morris, J. Naylor, C. Raymond, M. Rosetti, R. Santos, A. Sheridan, C. Sougnez, N. Stange-Thomann, N. Stojanovic, A. Subramanian, D. Wyman, J. Rogers, J. Sulston, R. Ainscough, S. Beck, D. Bentley, J. Burton, C. Clee, N. Carter, A. Coulson, R. Deadman, P. Deloukas, A. Dunham, I. Dunham, R. Durbin, L. French, D. Grafham, S. Gregory, T. Hubbard, S. Humphray, A. Hunt, M. Jones, C. Lloyd, A. McMurray, L. Matthews, S. Mercer, S. Milne, J. C. Mullikin, A. Mungall, R. Plumb, M. Ross, R. Shownkeen, S. Sims, R. H. Waterston, R. K. Wilson, L. W. Hillier, J. D. McPherson, M. A. Marra, E. R. Mardis, L. A. Fulton, A. T. Chinwalla, K. H. Pepin, W. R. Gish, S. L. Chissoe, M. C. Wendl, K. D. Delehaunty, T. L. Miner, A. Delehaunty, J. B. Kramer, L. L. Cook, R. S. Fulton, D. L. Johnson, P. J. Minx, S. W. Clifton, T. Hawkins, E. Branscomb, P. Predki, P. Richardson, S. Wenning, T. Slezak, N. Doggett, J.-F. Cheng, A. Olsen, S. Lucas, C. Elkin, E. Uberbacher, M. Frazier, R. A. Gibbs, D. M. Muzny, S. E. Scherer, J. B. Bouck, E. J. Sodergren, K. C. Worley, C. M. Rives, J. H. Gorrell, M. L. Metzker, S. L. Naylor, R. S. Kucherlapati, D. L. Nelson, G. M. Weinstock, Y. Sakaki, A. Fujiyama, M. Hattori, T. Yada, A. Toyoda, T. Itoh, C. Kawagoe, H. Watanabe, Y. Totoki, T. Taylor, J. Weissenbach, R. Heilig, W. Saurin, F. Artiguenave, P. Brottier, T. Bruls, E. Pelletier, C. Robert, P. Wincker, A. Rosenthal, M. Platzer, G. Nyakatura, S. Taudien, A. Rump, D. R. Smith, L. Doucette-Stamm, M. Rubenfield, K. Weinstock, H. M. Lee, J. Dubois, H. Yang, J. Yu, J. Wang, G. Huang, J. Gu,

L. Hood, L. Rowen, A. Madan, S. Qin, R. W. Davis, N. A. Federspiel, A. P. Abola, M. J. Proctor, B. A. Roe, F. Chen, H. Pan, J. Ramser, H. Lehrach, R. Reinhardt, W. R. McCombie, M. de la Bastide, N. Dedhia, H. Blöcker, K. Hornischer, G. Nordsiek, R. Agarwala, L. Aravind, J. A. Bailey, A. Bateman, S. Batzoglou, E. Birney, P. Bork, D. G. Brown, C. B. Burge, L. Cerutti, H.-C. Chen, D. Church, M. Clamp, R. R. Copley, T. Doerks, S. R. Eddy, E. E. Eichler, T. S. Furey, J. Galagan, J. G. R. Gilbert, C. Harmon, Y. Hayashizaki, D. Haussler, H. Hermjakob, K. Hokamp, W. Jang, L. S. Johnson, T. A. Jones, S. Kasif, A. Kaspryzk, S. Kennedy, W. J. Kent, P. Kitts, E. V. Koonin, I. Korf, D. Kulp, D. Lancet, T. M. Lowe, A. McLysaght, T. Mikkelsen, J. V. Moran, N. Mulder, V. J. Pollara, C. P. Ponting, G. Schuler, J. Schultz, G. Slater, A. F. A. Smit, E. Stupka, J. Szustakowki, D. Thierry-Mieg, J. Thierry-Mieg, L. Wagner, J. Wallis, R. Wheeler, A. Williams, Y. I. Wolf, K. H. Wolfe, S.-P. Yang, R.-F. Yeh, F. Collins, M. S. Guyer, J. Peterson, A. Felsenfeld, K. A. Wetterstrand, R. M. Myers, J. Schmutz, M. Dickson, J. Grimwood, D. R. Cox, M. V. Olson, R. Kaul, C. Raymond, N. Shimizu, K. Kawasaki, S. Minoshima, G. A. Evans, M. Athanasiou, R. Schultz, A. Patrinos, M. J. Morgan, I. H. G. S. Consortium, C. f. G. R. Whitehead Institute for Biomedical Research, T. S. Centre:, W. U. G. S. Center, U. D. J. G. Institute:, B. C. of Medicine Human Genome Sequencing Center:, R. G. S. Center:, Genoscope, C. UMR-8030:, I. o. M. B. Department of Genome Analysis, G. S. Center:, B. G. I. G. Center:, T. I. f. S. B. Multimegabase Sequencing Center, S. G. T. Center:, U. of Oklahoma's Advanced Center for Genome Technology:, M. P. I. for Molecular Genetics:, L. A. H. G. C. Cold Spring Harbor Laboratory, G. R. C. for Biotechnology:, a. i. i. l. u. o. h. *Genome Analysis Group (listed in alphabetical order, U. N. I. o. H. Scientific management: National Human Genome Research Institute, S. H. G. Center:, U. of Washington Genome Center:, K. U. S. o. M. Department of Molecular Biology, U. of Texas Southwestern Medical Center at Dallas:, U. D. o. E. Office of Science, and T. W. Trust:, "Initial sequencing and analysis of the human genome," *Nature*, vol. 409, p. 860–921, Feb 2001. Bandiera$_{a}btest : aCg_{t}ype : NatureResearchJournalsnumber : 6822Primary_{a}type : Researchpublisher : NaturePublishingGroup.$

[52] J. Vamathevan, D. Clark, P. Czodrowski, I. Dunham, E. Ferran, G. Lee, B. Li, A. Madabhushi, P. Shah, M. Spitzer, and S. Zhao, "Applications of machine

learning in drug discovery and development," *Nature reviews. Drug discovery*, vol. 18, p. 463–477, Jun 2019.

[53] P. R. Costa, M. L. Acencio, and N. Lemke, "A machine learning approach for genome-wide prediction of morbid and druggable human genes based on systems-level data," *BMC genomics*, vol. 11 Suppl 5, p. S9, Dec 2010.

[54] S. A. Ament, J. R. Pearl, J. P. Cantle, R. M. Bragg, P. J. Skene, S. R. Coffey, D. E. Bergey, V. C. Wheeler, M. E. MacDonald, N. S. Baliga, J. Rosinski, L. E. Hood, J. B. Carroll, and N. D. Price, "Transcriptional regulatory networks underlying gene expression changes in huntington's disease," *Molecular Systems Biology*, vol. 14, p. e7435, Mar 2018.

[55] K. A. Carpenter and X. Huang, "Machine learning-based virtual screening and its applications to alzheimer's drug discovery: A review," *Current Pharmaceutical Design*, vol. 24, p. 3347–3358, Dec 2018.

[56] W. Cheng and C. A. Ng, "Using machine learning to classify bioactivity for 3486 per- and polyfluoroalkyl substances (pfass) from the oecd list," *Environmental Science and Technology*, vol. 53, p. 13970–13980, Dec 2019.

[57] D. Plewczynski, M. Grotthuss, L. Rychlewski, and K. Ginalski, "Virtual high throughput screening using combined random forest and flexible docking," *Combinatorial Chemistry High Throughput Screening*, vol. 12, p. 484–489, Jun 2009.

[58] A. Lavecchia, "Machine-learning approaches in drug discovery: methods and applications," *Drug discovery today*, vol. 20, no. 3, pp. 318–331, 2015.

[59] F. Cheng, Y. Yu, J. Shen, L. Yang, W. Li, G. Liu, P. W. Lee, and Y. Tang, "Classification of cytochrome p450 inhibitors and noninhibitors using combined classifiers," *Journal of Chemical Information and Modeling*, vol. 51, p. 996–1011, May 2011.

[60] J. Meslamani, R. Bhajun, F. Martz, and D. Rognan, "Computational profiling of bioactive compounds using a target-dependent composite workflow," *Journal of Chemical Information and Modeling*, vol. 53, p. 2322–2333, Sep 2013.

[61] W. Hussain, N. Rasool, and Y. D. Khan, "Insights into machine learning-based approaches for virtual screening in drug discovery: Existing strategies and streamlining through fp-cadd," *Current Drug Discovery Technologies*, vol. 18, p. 463–472, Jul 2021.

[62] C. Angermueller, T. Pärnamaa, L. Parts, and O. Stegle, "Deep learning for computational biology," vol. 12, no. 7, p. 878. Publisher: John Wiley & Sons, Ltd.

[63] S. M. Anwar, M. Majid, A. Qayyum, M. Awais, M. Alnowami, and M. K. Khan, "Medical Image Analysis using Convolutional Neural Networks: A Review," *Journal of Medical Systems*, vol. 42, p. 226, Oct. 2018.

[64] Y. Jing, Y. Bian, Z. Hu, L. Wang, and X.-Q. S. Xie, "Deep learning for drug design: an artificial intelligence paradigm for drug discovery in the big data era," *The AAPS Journal*, 2018.

[65] G. Schneider, "Mind and machine in drug design," *Nature Machine Intelligence*, vol. 1, pp. 128–130, Mar. 2019. Number: 3 Publisher: Nature Publishing Group.

[66] I. Wallach, M. Dzamba, and A. Heifets, "Atomnet: A deep convolutional neural network for bioactivity prediction in structure-based drug discovery," *CoRR*, vol. abs/1510.02855, 2015.

[67] D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams, "Convolutional networks on graphs for learning molecular fingerprints," in *Advances in neural information processing systems*, pp. 2224–2232, 2015.

[68] I. Muegge and P. Mukherjee, "An overview of molecular fingerprint similarity search in virtual screening," *Expert Opinion on Drug Discovery*, vol. 11, pp. 137–148, Feb. 2016. Publisher: Taylor & Francis.

[69] J. C. Pereira, E. R. Caffarena, and C. N. dos Santos, "Boosting Docking-Based Virtual Screening with Deep Learning," *Journal of Chemical Information and Modeling*, vol. 56, pp. 2495–2506, Dec. 2016.

[70] M. Hirohara, Y. Saito, Y. Koda, K. Sato, and Y. Sakakibara, "Convolutional neural network based on smiles representation of compounds for detecting chemical motif," *BMC Bioinformatics*, vol. 19, p. 526, Dec 2018.

[71] D. Weininger, "SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules," *Journal of Chemical Information and Computer Sciences*, vol. 28, pp. 31–36, Feb. 1988. Publisher: American Chemical Society.

[72] D. Jimenez-Carretero, V. Abrishami, L. Fernández-de Manuel, I. Palacios, A. Quílez-Álvarez, A. Díez-Sánchez, M. A. d. Pozo, and M. C. Montoya, "Tox(r)cnn: Deep learning-based nuclei profiling tool for drug toxicity screening," *PLOS Computational Biology*, vol. 14, p. e1006238, Nov 2018.

[73] G. B. Goh, C. Siegel, A. Vishnu, N. O. Hodas, and N. Baker, "Chemception: A deep neural network with minimal chemistry knowledge matches the performance of expert-developed qsar/qspr models," *arXiv:1706.06689 [cs, stat]*, Jun 2017. arXiv: 1706.06689.

[74] A. Varnek and I. Baskin, "Machine Learning Methods for Property Prediction in Chemoinformatics: *Quo Vadis* ?," *Journal of Chemical Information and Modeling*, vol. 52, pp. 1413–1437, June 2012.

[75] F. Ghasemi, A. Mehridehnavi, A. Pérez-Garrido, and H. Pérez-Sánchez, "Neural network and deep-learning algorithms used in QSAR studies: merits and drawbacks," *Drug Discovery Today*, vol. 23, pp. 1784–1790, Oct. 2018.

[76] K. Yao and J. Parkhill, "Kinetic energy of hydrocarbons as a function of electron density and convolutional neural networks.," *Journal of chemical theory and computation*, 2016.

[77] T. B. Kimber, Y. Chen, and A. Volkamer, "Deep learning in virtual screening: Recent applications and developments," *International Journal of Molecular Sciences*, vol. 22, no. 9, 2021.

[78] D. Sydow, L. Burggraaff, A. Szengel, H. W. T. van Vlijmen, A. P. IJzerman, G. J. P. van Westen, and A. Volkamer, "Advances and challenges in computational target prediction," *Journal of Chemical Information and Modeling*, vol. 59, no. 5, pp. 1728–1742, 2019.

[79] R. Gupta, D. Srivastava, M. Sahu, S. Tiwari, R. K. Ambasta, and P. Kumar, "Artificial intelligence to deep learning: machine intelligence approach for drug discovery," *Molecular Diversity*, vol. 25, p. 1315–1360, Aug 2021.

[80] "CHeMBL Database." `https://www.ebi.ac.uk/chembl/`. Accessed: 24/09/2018.

[81] A. Gaulton, L. J. Bellis, A. P. Bento, J. Chambers, M. Davies, A. Hersey, Y. Light, S. McGlinchey, D. Michalovich, B. Al-Lazikani, and J. P. Overington, "Chembl: a large-scale bioactivity database for drug discovery," *Nucleic Acids Research*, vol. 40, pp. D1100–1107, Jan 2012.

[82] *rdkit/rdkit:* $2021_09_2(Q32021)$ *Release.* Oct 2021.

[83] A. Dalby, J. G. Nourse, W. D. Hounshell, A. K. I. Gushurst, D. L. Grier, B. A. Leland, and J. Laufer, “Description of several chemical structure file formats used by computer programs developed at molecular design limited,” *Journal of Chemical Information and Computer Sciences*, vol. 32, p. 244–255, May 1992.

[84] “Daylight Chemical Information Systems.” `https://www.daylight.com/`. Accessed: 24/01/2019.

[85] A. Dietz, “Yet another representation of molecular structure,” *Journal of Chemical Information and Computer Sciences*, vol. 35, p. 787–802, Sep 1995.

[86] E. J. Bjerrum, “SMILES Enumeration as Data Augmentation for Neural Network Modeling of Molecules,” *arXiv:1703.07076 [cs]*, May 2017. arXiv: 1703.07076.

[87] M. Yadav and S. E. Jujjavarapu, “Neural network methodology for the identification and classification of lipopeptides based on smiles annotation,” *Computers*, vol. 10, p. 74, Jun 2021.

[88] P. T. Habib, A. M. Alsamman, S. E. Hassanein, and A. Hamwieh, “Tardict: A randomforestclassifier based software predicts drug-target interaction using smiles,” *Highlights in Bioinformatics*, vol. 1, Mar 2021.

[89] L. A. Currie, “Nomenclature in evaluation of analytical methods including detection and quantification capabilities (iupac recommendations 1995),” *Pure and Applied Chemistry*, vol. 67, p. 1699–1723, Jan 1995.

[90] N. I. of Standards and Technology, “Security requirements for cryptographic modules,” Tech. Rep. Federal Information Processing Standards Publications (FIPS PUBS) 140-2, Change Notice 2 December 03, 2002, U.S. Department of Commerce, Washington, D.C., 2001.

[91] S. R. Heller, A. McNaught, I. Pletnev, S. Stein, and D. Tchekhovskoi, “Inchi, the iupac international chemical identifier,” *Journal of Cheminformatics*, vol. 7, p. 23, May 2015.

[92] A. Cereto-Massagué, M. J. Ojeda, C. Valls, M. Mulero, S. Garcia-Vallvé, and G. Pujadas, “Molecular fingerprint similarity search in virtual screening,” *Methods*, vol. 71, p. 58–63, Jan 2015.

[93] "Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations?," vol. 7.

[94] G. Maggiora, M. Vogt, D. Stumpfe, and J. Bajorath, "Molecular similarity in medicinal chemistry: Miniperspective," *Journal of Medicinal Chemistry*, vol. 57, p. 3186–3204, Apr 2014.

[95] R. E. Carhart, D. H. Smith, and R. Venkataraghavan, "Atom pairs as molecular features in structure-activity studies: definition and applications," *Journal of Chemical Information and Computer Sciences*, vol. 25, p. 64–73, May 1985.

[96] P. koda and D. Hoksza, "Exploration of topological torsion fingerprints," in *2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, p. 822–828, Nov 2015.

[97] D. Rogers and M. Hahn, "Extended-Connectivity Fingerprints," Apr 2010.

[98] J. L. Durant, B. A. Leland, D. R. Henry, and J. G. Nourse, "Reoptimization of mdl keys for use in drug discovery," *Journal of Chemical Information and Computer Sciences*, vol. 42, p. 1273–1280, Dec 2002.

[99] J. M. Barnard, G. M. Downs, A. von Scholley-Pfab, and R. D. Brown, "Use of markush structure analysis techniques for descriptor generation and clustering of large combinatorial libraries," *Journal of Molecular Graphics and Modelling*, vol. 18, p. 452–463, Oct 2000.

[100] n. Schneider, n. Neidhart, n. Giller, and n. Schmid, ""scaffold-hopping" by topological pharmacophore search: A contribution to virtual screening," *Angewandte Chemie (International Ed. in English)*, vol. 38, p. 2894–2896, Oct 1999.

[101] M. Yang, B. Tao, C. Chen, W. Jia, S. Sun, T. Zhang, and X. Wang, "Machine learning models based on molecular fingerprints and an extreme gradient boosting method lead to the discovery of jak2 inhibitors," *Journal of Chemical Information and Modeling*, vol. 59, p. 5002–5012, Dec 2019.

[102] S. Zhong, J. Hu, X. Fan, X. Yu, and H. Zhang, "A deep neural network combined with molecular fingerprints (dnn-mf) to develop predictive models for hydroxyl radical rate constants of water contaminants," *Journal of Hazardous Materials*, vol. 383, p. 121141, Feb 2020.

[103] B. Zagidullin, Z. Wang, Y. Guan, E. Pitkänen, and J. Tang, "Comparative analysis of molecular fingerprints in prediction of drug combination effects," *Briefings in Bioinformatics*, vol. 22, p. bbab291, Nov 2021.

[104] K. Abbasi, P. Razzaghi, A. Poso, S. Ghanbari-Ara, and A. Masoudi-Nejad, “Deep learning in drug target interaction prediction: Current and future perspectives,” *Current Medicinal Chemistry*, vol. 28, p. 2100–2113, Apr 2021.

[105] P. S. Roy and B. J. Saikia, “Cancer and cure: A critical analysis,” vol. 53, p. 441–442, Sep 2016.

[106] D. M. Hausman, “What is cancer?,” *Perspectives in Biology and Medicine*, vol. 62, no. 4, p. 778–784, 2019.

[107] M. Varjosalo, S. Keskitalo, A. Van Drogen, H. Nurkkala, A. Vichalkovski, R. Aebersold, and M. Gstaiger, “The protein interaction landscape of the human cmgc kinase group,” *Cell Reports*, vol. 3, p. 1306–1320, Apr 2013.

[108] M. K. Diril, C. K. Ratnacaram, V. Padmakumar, T. Du, M. Wasser, V. Coppola, L. Tessarollo, and P. Kaldis, “Cyclin-dependent kinase 1 (cdk1) is essential for cell division and suppression of dna re-replication but not for liver regeneration,” *Proceedings of the National Academy of Sciences*, vol. 109, no. 10, pp. 3826–3831, 2012.

[109] S. Izadi, A. Nikkhoo, M. Hojjat-Farsangi, A. Namdar, G. Azizi, H. Mohammadi, M. Yousefi, and F. Jadidi-Niaragh, “Cdk1 in breast cancer: Implications for theranostic potential,” *Anti-Cancer Agents in Medicinal Chemistry*, vol. 20, no. 7, p. 758–767, 2020.

[110] J. Li, Y. Wang, X. Wang, and Q. Yang, “Cdk1 and cdc20 overexpression in patients with colorectal cancer are associated with poor prognosis: evidence from integrated bioinformatics analysis,” *World Journal of Surgical Oncology*, vol. 18, p. 50, Mar 2020.

[111] M. Li, F. He, Z. Zhang, Z. Xiang, and D. Hu, “Cdk1 serves as a potential prognostic biomarker and target for lung cancer,” vol. 48, p. 0300060519897508, Feb 2020.

[112] X. Ying, X. Che, J. Wang, G. Zou, Q. Yu, and X. Zhang, “Cdk1 serves as a novel therapeutic target for endometrioid endometrial cancer,” *J Cancer*, vol. 12, pp. 2206–2215, 2021.

[113] S. Sunada, H. Saito, D. Zhang, Z. Xu, and Y. Miki, “Cdk1 inhibitor controls g2/m phase transition and reverses dna damage sensitivity,” *Biochemical and Biophysical Research Communications*, vol. 550, p. 56–61, Apr 2021.

[114] B. S. Hendriks, "Functional pathway pharmacology: chemical tools, pathway knowledge and mechanistic model-based interpretation of experimental data," *Current Opinion in Chemical Biology*, vol. 14, no. 4, p. 489–497, 2010.

[115] J. MacQueen, "Some methods for classification and analysis of multivariate observations," *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, vol. 5.1, p. 281–298, Jan 1967.

[116] D. D. Wang, M.-T. Chan, and H. Yan, "Structure-based protein–ligand interaction fingerprints for binding affinity prediction," vol. 19, p. 6291–6300, Jan 2021.

[117] M. R. Berthold, N. Cebron, F. Dill, T. R. Gabriel, T. Kötter, T. Meinl, P. Ohl, K. Thiel, and B. Wiswedel, "Knime - the konstanz information miner: Version 2.0 and beyond," *SIGKDD Explor. Newsl.*, vol. 11, pp. 26–31, Nov. 2009.

[118] A. J. Kooistra, G. K. Kanev, O. P. van Linden, R. Leurs, I. J. de Esch, and C. de Graaf, "Klifs: a structural kinase-ligand interaction database," *Nucleic Acids Research*, vol. 44, no. D1, p. D365–D371, 2015.

[119] J. J. Irwin, "Community benchmarks for virtual screening," *Journal of Computer-Aided Molecular Design*, vol. 22, p. 193–199, Apr 2008.

[120] M. Réau, F. Langenfeld, J.-F. Zagury, N. Lagarde, and M. Montes, "Decoys selection in benchmarking datasets: Overview and perspectives," *Frontiers in Pharmacology*, vol. 9, 2018.

[121] T. Madden, *The BLAST Sequence Analysis Tool.* National Center for Biotechnology Information (US), Aug 2003.

[122] T. J. VanderWeele and P. Ding, "Sensitivity analysis in observational research: Introducing the e-value," *Annals of Internal Medicine*, vol. 167, p. 268, Aug 2017.

[123] I. Mendolia, S. Contino, U. Perricone, R. Pirrone, and E. Ardizzone, *A Convolutional Neural Network for Virtual Screening of Molecular Fingerprints*, vol. 11751 of *Lecture Notes in Computer Science*, p. 399–409. Cham: Springer International Publishing, 2019.

[124] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," *arXiv:1610.02357 [cs]*, Apr. 2017. arXiv: 1610.02357.

[125] A. Bender and R. C. Glen, “A Discussion of Measures of Enrichment in Virtual Screening: Comparing the Information Content of Descriptors with Increasing Levels of Sophistication,” *Journal of Chemical Information and Modeling*, vol. 45, pp. 1369–1375, Sept. 2005.

[126] S. M. Lundberg and S.-I. Lee, “A Unified Approach to Interpreting Model Predictions,” *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[127] S. Lipovetsky and M. Conklin, “Analysis of regression in game theory approach,” *Applied Stochastic Models in Business and Industry*, vol. 17, no. 4, pp. 319–330, 2001. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/asmb.446.

[128] A. Shrikumar, P. Greenside, and A. Kundaje, “Learning Important Features Through Propagating Activation Differences,” *arXiv:1704.02685 [cs]*, Oct. 2019. arXiv: 1704.02685.

[129] V. Mlynárik, “Introduction to nuclear magnetic resonance,” vol. 529, p. 4–9, Jul 2017.

[130] D. Banfi and L. Patiny, “www.nmrdb.org: Resurrecting and processing nmr spectra on-line,” *CHIMIA*, vol. 62, p. 280–280, Apr 2008.

[131] A. M. Castillo, L. Patiny, and J. Wist, “Fast and accurate algorithm for the simulation of nmr spectra of large spin systems,” *Journal of Magnetic Resonance*, vol. 209, p. 123–130, Apr 2011.

[132] J. Aires-de Sousa, M. C. Hemmer, and J. Gasteiger, “Prediction of 1h nmr chemical shifts using neural networks,” *Analytical Chemistry*, vol. 74, p. 80–90, Jan 2002.

[133] “Marvin | ChemAxon.”

[134] J. I. Baumbach, A. N. Davies, P. Lampen, and H. Schmidt, “Jcamp-dx. a standard format for the exchange of ion mobility spectrometry data (iupac recommendations 2001),” *Pure and Applied Chemistry*, vol. 73, p. 1765–1782, Nov 2001.

[135] A. N. Davies and P. Lampen, “Jcamp-dx for nmr,” *Applied Spectroscopy*, vol. 47, p. 1093–1099, Aug 1993.

[136] C. C. Aggarwal, *Data Mining*. Springer International Publishing, 2015.

[137] Y. Zhu, S. Alqahtani, and X. Hu, “Aromatic Rings as Molecular Determinants for the Molecular Recognition of Protein Kinase Inhibitors,” *Molecules*, vol. 26, p. 1776, Jan. 2021. Number: 6 Publisher: Multidisciplinary Digital Publishing Institute.

«Amat Victoria Curam»
Gaius Valerius Catullus