universität wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis

„Large-scale, structure-based virtual screening for the identification of novel Staphylococcal exfoliative toxin inhibitors“

verfasst von / submitted by

Iris Rami, BSc

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien, 2023 / Vienna 2023

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	UA 066 606
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Masterstudium Drug Discovery and Development
Betreut von / Supervisor:	Assoc.-Prof. Dr. Johannes Kirchmair

Acknowledgements

I would like to thank my supervisor, Assoc.-Prof. Dr. Johannes Kirchmair, for dedicating his time, for giving me precious advice and for the opportunity to work on an exciting project, which highly motivated but also challenged me.

I would like to thank the COMP3D group for the warm welcome and support. Especially Dr. Jorge E. Hernández González for introducing me in this exciting field, patiently helping me out when needed and for proofreading my thesis. Thank you Hosein Fooladi and Thi Ngoc Lan Vu, for always answering my many questions and helping me improve my coding skills. Many thanks to all the group members for your support, your time and inspiring conversations.

I would like to thank my family and friends for being there for me throughout the years. Especially my parents for unconditionally supporting me in every way. Many thanks to my dear friend Jessica Trenkwalder for her moral support and for motivating and inspiring me. Last but not least, I would like to thank my partner, Andrea Gremes, for believing in me and motivating me to always give my best.

Abstract

Structure-based virtual screening is the method of choice for identifying new virtual hits when no ligands for a target protein are known. Additionally, screening chemical libraries, which are as vast as possible, has many advantages as the chances of identifying novel scaffolds and true positives are higher. However, as the sizes of chemical libraries increase with each new release of molecular libraries, it becomes more and more computationally and time expensive to perform ultra-large virtual screening campaigns, which leverage the entirety of these libraries. Machine learning implementations have been very effective in mitigating these issues. Here, we explore the most relevant protocols for machine learning accelerated docking-based virtual screening and personalize two of the protocols (namely Deep Docking and MolPal) for identifying novel virtual inhibitors of exfoliative toxin A.

Exfoliative toxin A is a glutamic acid specific serine protease, which cleaves desmoglein-1 an important protein of the skin, responsible for keeping skin cells of the upper layer connected. This target protein is therefore responsible for a dangerous skin disease, which is associated with loss of the stratum granulosum and dehydration. So far, no inhibitors are known, therefore a structure based virtual screening is the best way for identifying new virtual hits.

By adopting the Deep Docking framework for virtually screening almost 900 million compounds the needed computational cost was reduced 12 times in comparison to conventional docking. In the end, 10 novel, prospective inhibitors, which show unique scaffolds and promising ligand-protein interactions, were retrieved.

In addition, a smaller comparative study was performed using the MolPal framework, which also reduced the computational cost substantially when screening a 2.1 million sized library. However, this framework only retrieved 10.2% of the actual 1,000 best-scoring compounds. This indicated room for improvement in future studies: New surrogate models, structure-based methods and scoring functions could be investigated in order to retrieve more personalizable protocols for handling target specific issues and different aims.

Kurzfassung

Strukturbasiertes virtuelles Screening ist die Methode der Wahl zur Identifizierung neuer, virtueller Hits, wenn keine Liganden für ein Target bekannt sind. Darüber hinaus bietet das Screening möglichst umfangreicher chemischer Bibliotheken deutliche Vorteile, da die Chancen, neue Grundgerüste und bioaktive Substanzen zu identifizieren erhöht werden. Da jedoch die Größe chemischer Bibliotheken mit jeder neuen Veröffentlichung zunimmt, wird die Durchführung extrem großer virtueller Screenings, die die Gesamtheit dieser Bibliotheken nutzen, immer rechen- und zeitaufwendiger. Die Implementierung von maschinellem Lernen stellt dabei eine effektive Lösung dar. In dieser Arbeit werden die wichtigsten Protokolle für das durch maschinelles Lernen beschleunigte Docking-basierte virtuelle Screening besprochen. Die zwei vielversprechendsten Protokolle (namentlich Deep Docking und MolPal) wurden ausgewählt, personalisiert, und für die Identifizierung neuartiger virtueller Hits von Exfoliatin A angewandt.

Exfoliatin A ist eine Glutaminsäure-spezifische Serinprotease, die Desmoglein-1 spaltet, ein wichtiges Protein der Haut, welches für das Zusammenheften der Hautzellen in der obersten Hautschicht zuständig ist. Dieses Zielprotein ist somit für eine gefährliche Hauterkrankung verantwortlich, die mit dem Verlust der oberflächlichen Hautschicht und mit Haut-Austrocknung assoziiert wird. Bisher gibt es keine Medikamente gegen Exfoliatin A, weshalb ein strukturbasiertes, virtuelles Screening eine gute Möglichkeit im Rahmen von computerbasierten Methode darstellt, um neue potentielle Inhibitoren zu identifizieren.

Durch den Einsatz des Deep Docking Frameworks für das virtuelle Screening von fast 900 Millionen Verbindungen konnten die erforderlichen Rechenkosten im Vergleich zum herkömmlichen Docking um das Zwölffache gesenkt werden. Es konnten zehn neue potenzielle Inhibitoren gefunden werden, die einzigartige Grundgerüste und vielversprechende Wechselwirkungen mit dem Protein aufweisen. Darüber hinaus wurde eine kleinere Vergleichsstudie mit dem MolPal-Framework durchgeführt, das beim Screening einer chemischen Bibliothek (2,2 Millionen Substanzen) ebenfalls zu einer erheblichen Reduzierung des Rechenaufwands führte. Allerdings wurden mit diesem Protokoll nur 10,2 % der 1.000 besten Verbindungen gefunden. Dies deutet darauf hin, dass in künftigen Studien noch Raum für Verbesserung besteht: Neue “surrogate-models", strukturbasierte Methoden und Scoring-Funktionen könnten untersucht werden, um eine Anpassung des Protokolls je nach Forschungsfrage und Zielprotein zu ermöglichen.

Contents

List of Figures

List of Tables

Acronyms

3D	three-dimensional
AUROC	receiver operating characteristics
D-MPN	direct message-passing neural network
DL	deep learning
DNN	deep neural network
Dsg-1	desmoglein 1
Dsg-3	desmoglein 3
DUD-E	Directory of Useful Decoys, Extended
ECFP	extended connectivity fingerprint
ET	exfoliative toxin
ETA	exfoliative toxin A
ETB	exfoliative toxin B
ETE	exfoliative toxin E
FFNN	feed forward neural network
FPDE	full predicted database enrichment
GCNN	graph convolutional neural network
GOLD	genetic optimisation for ligand docking
GPR	gaussian process regression
HASTEN	machine learning boosted docking
HTS	high throughput screening
LBVS	ligand-based virtual screening
LR	logistic regression
MEMES	machine learning framework for enhanced molecular screening
ML	machine learning
MLP	multi-layer perceptron

MolPal	molecular pool-based active learning
MPNN	message passing neural network
QSAR	quantitative structure−activity relation-ships
RF	random forest
RMSD	root-mean-square deviation
SBVS	structure-based virtual screening
SSSS	staphylococcal scalded skin syndrome
SVM	support vector machine
VS	virtual screening

1. Introduction

In recent years the size of most chemical libraries has substantially increased and this development is likely to continue.[1,2] Indeed, the number of drug-like compounds is roughly estimated to be far larger than the size of the current available libraries, with ~10^{60} possible compounds, when considering molecules with up to 30 atoms containing C, N, O and S.[3]

There are multiple chemical libraries which can be used for virtual screening (VS). Some of the most popular and publicly available ones are: (1) ZINC22 with 37 billion commercially accessible compounds,[2] Enamine, which offers (2) the Real[4] database with 6 billion compounds and (3) the Real Space[5] library with 36 billion accessible compounds. Therefore, it is of high interest to find new techniques of screening these ultra large libraries in order to explore the vast diversity of the chemical space. In fact, the more compounds are screened, the larger are the possibilities to identify compounds with new chemotypes and the higher the rate of true positives.[6,7]

In the past, the main method of discovering new hit compounds has been through experimental high throughput screening (HTS). This approach is associated with various issues, such as time- and cost-effectiveness.[8] However, as often the same set of molecules is screened for multiple targets, the explored chemical space during HTS campaigns is limited, thus restricting the scaffold novelty.[8] Due to the limitations regarding the amount of compounds that can be screened, it is also challenging to find hits with good activity profiles. Moreover, it is difficult to address challenging protein sites, such as protein-protein interactions and allosteric sites.[8]

1.1 Virtual Screening

In VS, ligand libraries are computationally screened in order to assess their activity for a given target. Through VS, most of the above-mentioned issues can be mitigated. Below, details of the most widely-used VS approaches are provided.

1.1.1 Ligand-based virtual screening

Ligand-based virtual screening (LBVS) relies on the similarity-property principle, which generally assumes that structures with similar chemical properties show similar biological activity.[9,10] Therefore, known ligands are used for finding new prospective hit compounds.[9] This method is therefore especially useful when no protein structure is

available. Widely used ligand-based methods include quantitative structure – activity relationships (QSAR), pharmacophore matching and similarity searching.[9]

1.1.2 Structure-based virtual screening

Structure-based virtual screening (SBVS) relies on the three-dimensional (3D) structure of the target protein, which can be derived experimentally or computationally.[9] SBVS, especially docking, is often chosen over LBVS as SBVS is more likely to find new scaffolds and it can be used when no ligands are known for a specific target.[8] Docking is a popular structure-based method and its potential for screening ultra large libraries has been demonstrated in multiple studies.[6,7,11,12]

1.1.3 Limitations of structure-based virtual screening

One major obstacle when screening ultra-large libraries through docking, especially in academia, is the computational cost.[13] A further limitation is the availability of high resolution structural information about the target protein.[13,14] Deep learning (DL) is trying to solve some of these problems for example via de novo structure prediction using AlphaFold.[15] Furthermore, DL-based docking programs are faster than traditional docking programs, as for instance GNINA[16] and AutoDock Vina[17], which are full docking programs adopting DL both for pose prediction as well as scoring.[16,17]

Another approach to reduce the computational cost of ultra-large virtual screening is the implementation of DL, not on the docking level, but rather on the screening level. Instead of docking the entire ultra-large library, models are trained iteratively on just a subset of the library. The resulting model is then applied to predict the activity of the compounds of the entire chemical library.[8] Consequently, this method can be highly effective in reducing the computational cost of docking ultra-large libraries. Protocols based on this principle will be discussed, evaluated and employed in this thesis.

1.2 Exfoliative toxin A

Exfoliative toxins (ETs) are highly selective glutamyl endopeptidases, which are members of the chymotrypsin-like serine protease family. ETs are secreted by different *Staphylococcus spp.*, including *S. aureus*, a dangerous pathogen responsible for various diseases in humans.[18,19] The main substrate of ETs is the membrane protein desmoglein 1 (Dsg-1). More specifically, Dsg-1 is a desmosomal cadherin responsible for the integrity of desmosomes, which are cell-to-cell adhesive structures. Early on, it was observed that ETs cause skin exfoliation only on the top layer of the skin, the stratum

granulosum, leaving deeper skin layers intact.[19] This is explained by the selectivity in hydrolyzing Dsg-1, which is the only isoform present in the stratum granulosum. Whereas in the other strata, Dsg-3 is also expressed and able to compensate for Dsg-1 cleavage.[18] This mechanism of action is the direct cause of staphylococcal scalded skin syndrome (SSSS), a disease which occurs mainly in infants and shows symptoms such as loss of the superficial skin layer and dehydration as well as leading to secondary infections.[18]

There are at least four isoforms of ETs, which have been characterized. The main isoforms known to cause SSSS in humans are ETA and ETB.[19] The latest isolated isoform is ETE, which is also capable of cleaving Dsg-1 in the human epidermis.[20] However, ETA is far more prevalent than the other ETs and, therefore, also the target protein for this work.[18]

As for the structure of ETs, all of its isoforms have the catalytic triad (Asp-His-Ser) that is characteristic to all the chymotrypsin-like serine proteases. Moreover, ETs also have some structural elements leading to the selectivity for cleaving the glutamic acid-glycine bond in Dsg-1, which are also typical for glutamyl endopeptidases.[19] However, ETs possess two very specific structural features: firstly, an N-terminal α-helix containing charged amino acids can be observed in the available crystal structures. Secondly, the residues forming the oxyanion hole of the enzyme have a distinct conformation. These structural variations may be the reasons why ETs show no activity against broad spectrum substrates of chymotrypsin-like serine proteases.[19] The structure of ETA is depicted in Figure 1, the catalytic triad (D102-H57-S195) is highlighted as well as the important amino acids of the binding pocket: T190, H213 and K216. Furthermore, P192, a key residue of the oxyanion hole is depicted.[19]

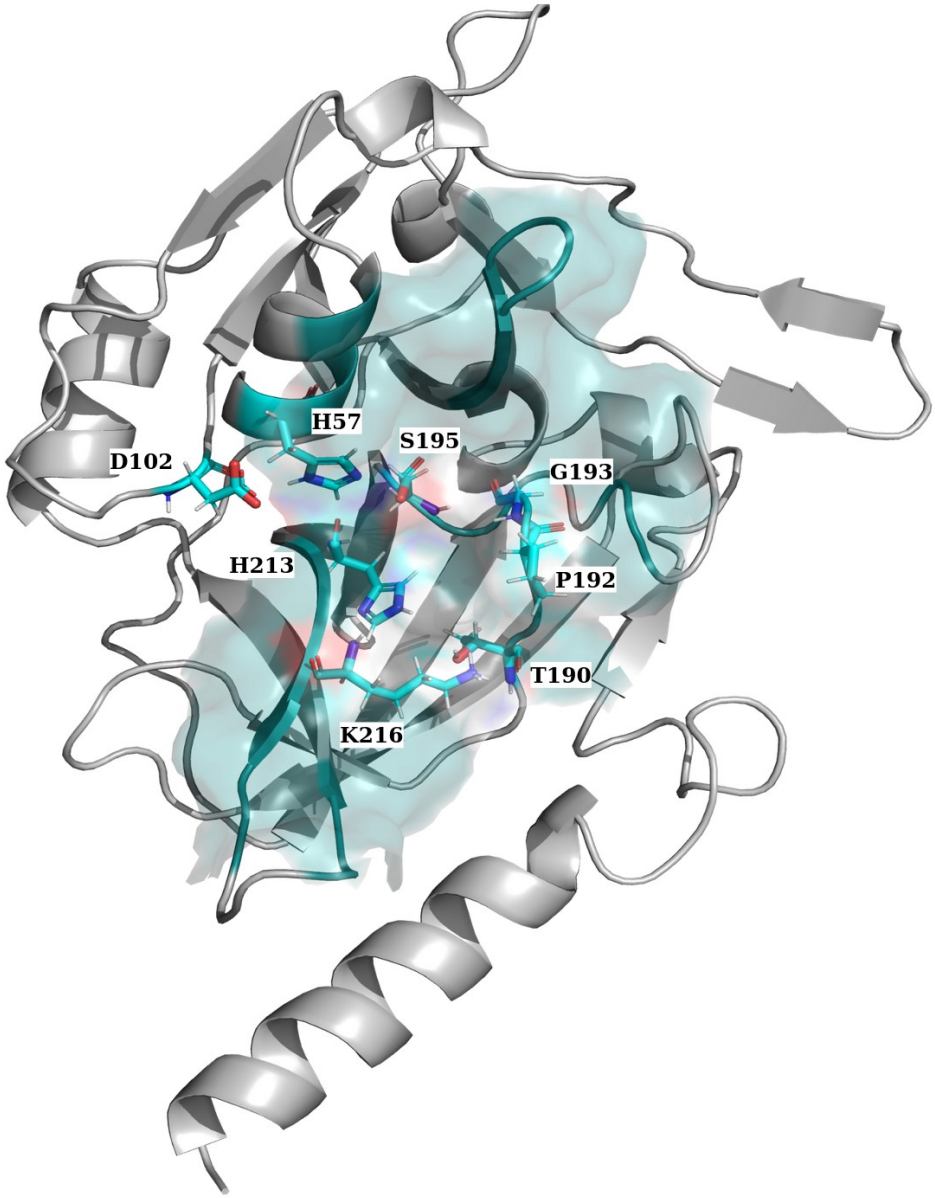

Figure 1: ETA structure retrieved and modified from PDB: 1AGJ21 in order to retrieve the active conformation for docking purposes. The conformation of P192 was manually changed as described in the "Protein preparation" section. The protein is depicted in cartoon representation. Important amino acids as well as the catalytic triad are highlighted in cyan, the binding site is highlighted as a transparent surface.

As of now there is no structural data available showing the interaction of ETA with small molecule inhibitors, thus making a LBVS approach unfeasible. Therefore, it is extremely important to find innovative techniques for discovering novel potential inhibitors of ETA. In this work, a machine learning (ML) framework for accelerated docking-based VS will be adopted. This allows the screening of billion-size libraries, thus increasing the chances of finding potential ligands of the target protein. The right framework is chosen based on the available protocols which are discussed in the following chapters.

1.3 Related work

A brief summary of the most important work in the field of ML accelerated docking-based VS is provided in Table 1. The protocols are referred to either by their name or, if no name is provided, by the surname of the first author and the publication year. In the following sub-chapters the studies from Table 1 are presented in more detail. Every work mentioned presents different variations of the above mentioned method for more efficient SBVS. Two of the protocols will be chosen and adapted in order to establish a customized ML accelerated SBVS protocol.

Table 1: List of work related to ML accelerated VS in the last 5 years. [1]

Protocol name	Input	ML model	Advantages of the method	Disadvantages of the method
Svensson - 2017[22]	Structural and physical descriptors	RF	+ More than 10 fold reduction of the screening dataset	- Experimental testing required
Ahmed - 2018[23]	Signature molecular descriptors	SVM	+ 62.61% reduction of docking library	- Still time and resources consuming for ultra-large libraries
Deep Docking - 2020[24]	Morgan fingerprints	FFNN	+ 50 to 100-fold reduction of library	- Computationally still demanding - Easy to implement other docking programs
MEMES - 2021[25]	ECFP, Mol2Vec, CDDD	GPR	+ Only 6% of library was docked	- Use of k-means to cluster the compounds
MolPAL - 2021[26]	Atom-pair fingerprints or molecular graph	RF, FFNN, MPNN	+ 40-fold reduction of library + Use of Beysian optimization	- Only supports docking using pyscreener
Yang - 2021[27]	Molecular graph or Morgan fingerprints	GCNN, RF, DNN	+ 14-fold reduction of computing cost	- Use of not publicly available tools

HASTEN - 2021[28]	RDKit descriptors	MPNN	+ 10- to 100-fold reduction of library + easy to change ML or docking methodology	- New test and validation set after each iteration
Martin - 2021[29]	Morgan fingerprints	LR	+ Uses a simple ML model able to achieve promising results	- Not peer reviewed work - LR model is prone to overfitting

[1] Abbreviations: RF, random forest; SVM, support vector machine; FFNN, feed forward neural network; ECFP, extended connectivity fingerprint; GPR, gaussian process regression; MPNN, message passing neural network; GCNN, graph convolutional neural network; DNN, deep neural network; LR, logistic regression

1.3.1 Svensson et al. - 2017

In the study of Svensson et al.[20], a subset of a library is docked. The predicted hits are experimentally validated and the results are used to train a conformal predictor, which is based on a RF classification model trained on RDKit[30] descriptors. The conformal predictor classifies the remaining compounds as either of the two classes (active or inactive) but also as both or none of the classes. Only compounds with one “active” label were chosen for a second round of screening. This is done until a desired number of active compounds is identified. This approach is validated by using the Directory of Useful Decoys, Extended (DUD-E)[31] database.[22] Overall 57% of the active compounds were identified, when only screening 9.4% of the original library.[22] This approach is not purely computational, as experimental results are necessary for training the model. For the objective of screening ultra-large libraries this approach would still be costly and time consuming.

1.3.2 Ahmed et al. - 2018

In the study of Ahmed et al.[23], a subset of compounds from a chemical library is docked, the compounds are classified based on the docking scores as “active” and “inactive”. An inductive conformal prediction approach with SVM as an underlining classifier is trained and used for predictions on the entire database. The compounds with one single “inactive” label are removed from the library and, therefore, never docked. The model is retrained by choosing a small subset of compounds from the reduced library, which are again docked and labeled. This is done iteratively until the desired efficiency of the model is reached.[23] For validating this approach the entire library is docked for better comparison with the Ahmed 2018 protocol, which, in fact, reduced the number of docked

compounds by more than 60% while maintaining an accuracy of 94%, when aiming for the top 30 hits.[23] However, a 60% reduction of the to-dock library, although remarkable, is still not enough for screening ultra-large libraries with limited computational resources.

1.3.3 Deep Docking - 2020

In 2020, Gentile et al. developed Deep Docking.[24] Here, a subset of compounds is chosen, docked and labeled as “active” or “inactive” depending on a threshold set based on the top scoring compounds. A FFNN is trained based on the Morgan fingerprints of the labeled compounds and the resulting model is used to classify the entire library. In the following iterations, new and improved models are generated by adding a fraction of prospective hits to the training set, a procedure called training set augmentation. The chosen compounds are again docked, and a new model is trained. This process is repeated for a predefined number of iterations or until the number of prospective hit compounds is sufficiently small.[14] This protocol is the first of its kind to be applied on an ultra-large library (1.36 billion) and was able to reduce the docking library 50 to 100 times without any notable hit loss being reported.[14,24]

1.3.4 MEMES - 2021

In MEMES (machine learning framework for enhanced molecular screening)[25] a major difference in comparison to other protocols is introduced: After featurizing the entire chemical library, the compounds are clustered using k-means. A small subset from each cluster is chosen and subsequently docked for training the model. From this point, Bayesian optimization is applied for training the GPR surrogate model and choosing new compounds to add to the training set based on an acquisition function. This is done iteratively until the maximal number of iterations is reached.[25] This protocol was able to identify about 90% of the top-scoring 1,000 compounds when only docking 6% of a chemical library containing 100 million compounds.[25] The main drawback of this protocol lies in the k-mean clustering. In fact, if the number of clusters is not correctly set by the user it might lead to over- or under-clustering.[32] This is especially difficult with ultra-large libraries. Secondly, k-means clustering is time-consuming when applied to ultra-large libraries.[32]

1.3.5 MolPal - 2021

MolPAL[26] (molecular pool-based active learning) is similar to MEMES[25] in the sense that Bayesian optimization is used for iteratively training the surrogate model with new compounds selected based on an acquisition function. The main difference is that the

initial batch of compounds for training the surrogate model is chosen randomly instead of by clustering. This protocol retrieved almost 95% of the top 50,000 compounds with a 40-fold reduction of the docking library.[26] A key advantage of MolPal is its flexibility in choosing the surrogate model, acquisition metric and batch size depending on the dataset size and aims of the work.

The same group developed a method named “active design space pruning”[33] for excluding all low scoring compounds from the dataset, based on the observation that, when using MolPal, most compounds predicted as low-scoring ones in the first iteration remain that way during subsequent iterations. This method excludes compounds by calculating their probability of being a hit. This probability is based on the mean (predicted docking score) and the uncertainty of a compound. This does not involve high additional computational cost as the mean and uncertainty are already required by Bayesian optimization used in MolPal.[33] Again, the user must evaluate whether the lower computational cost is worth potentially losing hit compounds.

1.3.6 Yang et al. - 2021

Yang et al. published a protocol[27] with a similar concept to Deep Docking. One main difference is that the AutoQSAR/DeepChem package[34] of Schrodinger is used for selecting the best model between GCNN and RF. Furthermore, the hyperparameters are defined in an automated manner. The user needs to specify a time period within which the hyperparameter and model search is performed.[27] The main disadvantage of this approach is that it adopts tools which are currently not publicly available. Though, the automated ML approach could be an advantage for non-experienced programmers.

1.3.7 HASTEN - 2021

HASTEN[28] (machine learning boosted docking) shares many similarities with the Deep Docking protocol: In HASTEN a subset of compounds from a library is randomly chosen, docked and a ML model is trained based on the docking scores and applied to the chemical library. For the subsequent iterations compounds for training the ML model are selected based on the pool of predicted hits. This is done for a set number of iterations. The authors report some advantages over previously discussed protocols. Firstly, HASTEN allows a simple implementation of any docking program and ML methodology. Secondly, with each iteration step, new compounds with docking scores are added to the training, validation and test set, and not only to the training set as described with Deep Docking.[28] This last point might be a disadvantage as the models generated after each iteration cannot be compared to each other as different test sets will be used.

1.3.8 Martin et al. - 2021

In Martin et al.[29] a simple LR model is used for ML, which enables ultra large VS. In this study, Morgan fingerprints[35] with a pharmacophoric invariance of size 8,192 were used. Their results are also compared with those of MolPal and it is concluded that the more simple logistic regression reaches better results than the MPNN model, which was reported to be the best performing model in MolPal.[29] However, the linear regression model might have a tendency to overfit when handling a large number of compounds.[36]

1.4 Objective of this work

The objective of this work is to establish a customized workframe for accelerating docking-based VS. For this aim, the protocols Deep Docking and MolPal were chosen and GOLD was incorporated into both protocols for docking. Regardless of the chosen ML framework, the accuracy of the protocol is dependent on the reliability of the docking algorithm employed, as the ML models are trained on the docking scores. Although GOLD was first developed in 1995 it has been continuously improved over the years and remains one of the most used docking algorithms.[37,38] Having GOLD as an additional option to perform docking within the Deep Docking and MolPal protocols is, therefore, an enormous advantage.

Deep Docking was chosen as it is the first protocol of its kind to be applied on ultra-large libraries and showing good results in retrieving top-scoring compounds while reducing computational cost. Furthermore, Deep Docking has been adopted by other groups and proven its efficiency in retrieving actual hits, when testing the chosen prospective hits in *in vitro* studies.[39,40] MolPal, on the other hand, is the only protocol using active learning and incorporating pruning of the inactive labeled compounds. These methods are very efficient in both retrieving prospective active compounds and reducing computational cost.

1.5 Computational Methods

In the following the main computational methods adopted in the selected VS protocols are explained.

1.5.1 Deep learning in drug discovery

DL approaches have been widely implemented in every part of the drug discovery pipeline. In the previous chapters, some implementations in VS were discussed. There are different types of DL algorithms, which are able to recognize patterns and extract features in different ways based on how the training set is structured.[41] In the first protocol, a multi-layer perceptron (MLP) will be employed for the purpose of generating models, which will predict the potential ligands from a whole chemical library based on calculated docking scores of a subset of the library. In the second protocol a direct message passing neural network (D-MPN) will be employed for the same aim.

1.5.1.1 Multi-layer perceptron

In Deep Docking, the MLP is the DL model of choice. MLP is a type of FFNN and also the most common type to be used in drug discovery and, therefore, has been extensively studied. The MLP is trained by detecting the optimal parameters for minimizing the error between the predicted and the actual label of the input data set.[42] As can already be deduced from the name, the MLP consist of three or more layers: an input layer, an output layer and one or more hidden layers. Various representations of the molecules can be used for training, in this case, 1,024-bit Morgan fingerprints[35] with a radius of 2 were chosen. The exact architecture of the MLP model is described in the respective method section in detail.

1.5.1.2 Message-passing neural network

When using MolPal, the D-MPN was adopted, as it was the best performing model reported by the authors.[26] The main difference to the MLP is that it operates on a graph rather than on calculated features. The D-MPN functions in two phases, a message passing phase and a readout phase. During the first phase “messages” or information are passed between atoms and/or bonds and their hidden state is updated depending on the incoming information. This procedure is repeated for a specified number of iterations. Finally, the hidden states are aggregated in order to produce a molecule-level feature vector.[26]

1.5.2 Bayesian optimization

MolPal[26] adopts Bayesian optimization to build the D-MPN model. Bayesian optimization is a form of active learning that iteratively selects the experiments to perform depending on the predictions made by a surrogate model. In this protocol the Bayesian optimization was performed on a chemical library and the experiments, here molecules, were selected in batches rather than one by one, therefore, “batched Bayesian optimization” was performed. Batched Bayesian optimization generally begins by calculating the objective function for a random set within the pool. Subsequently, a surrogate model is trained on the data and passed on to an acquisition function together with the value of the current maximum objective function. The acquisition function calculates the utility of evaluating the objective function of a new point. In this case, the utility was measured in a greedy way, meaning that batches are selected based on the highest objective function value.[26,43] The points with the largest utility are acquired, the objective value is calculated and the model is updated with the new data. This process is repeated iteratively.[26]

1.5.3 Docking

Molecular docking refers to a method that evaluates the predicted poses of molecules in the binding site of a target protein. Search algorithms generate poses, which are then ranked by scoring functions.[44] In the case of GOLD (Genetic Optimisation for Ligand Docking), a genetic algorithm is employed to explore the possible ligand conformations. In this work, the default ChemPLP score was used as a scoring function.[37] ChemPLP models van der Waals and repulsive terms in order to evaluate the generated poses and it is generally very efficient for VS campaigns.[45]

2. Methods

2.1 Ligand preparation

On the one hand the ZINC20[1] "Wait OK" library, containing over 880 million compounds, was used for the experiments run using the Deep Docking protocol. The library was downloaded using the tranches browser on the ZINC website[46]. This resulted in the download of 1,900 tranches. On the other hand, the Enamine HTS library (2.1 million compounds) was used for evaluating the MolPal workframe. Both libraries were prepared using tools from OpenEye as described in the Deep Docking protocol[14]: Flipper, which is part of OMEGA 4.1.0.0[47], was used to calculate possible stereoisomers of each molecule. The tautomers tool from QUACPAC 2.1.1.0[48] was used to determine the dominant tautomer (at pH 7.4) of each isomer. Default settings were used for both procedures. The only change was limiting the maximum number of chirality centers to be enumerated to 8 when using flipper, as the ZINC library would get too large to be handled on one single workstation when using the default value of 12. To ensure an efficient execution of these steps, a shell script was written that prepares all the 1,900 tranches of the ZINC library. After the preparation the ZINC library consisted in 890 million compounds and the Enamine library in 3.3 million compounds.
Before starting with the docking procedure, 3D structures of each compound to be docked were generated using the OPLS4 force field. This was done with LigPrep from Schrodinger 20221-1.

2.2 Protein preparation

Firstly, the position of P192 was manually changed in order to retrieve the active conformation of ETA. In PDB 1AGJ[21], the carbonyl oxygen of the mentioned proline residue is flipped 180° in comparison to all other chymotrypsin-like serine proteases, thus hindering the entrance of the ligand into the oxyanion hole. This conformation is also reported in other isoforms and assumed to be the inactive state of the protein.[19]
The protein was further prepared in Maestro 12.7.161 using the protein preparation wizard from Schrodinger's Epik 5.5161 module[49] and keeping mostly default parameters: hydrogens were added, water molecules were removed as they might hinder prospective hits from forming interactions within the binding site (this is a variation from default), bond orders were assigned based on the Cambridge crystallographic database (CCD),

the states of the heteroatoms at pH 7.0+/-2.0 were calculated using Epik. Furthermore, the protein structure was subjected to energy minimization by applying the OPLS4 force field[50] and converging heavy atoms to a root-mean-square deviation (RMSD) of 0.30 Å. The prepared protein was loaded into Hermes 1.10.5[51]. Here, the binding pocket grid was defined based on a previously docked ligand. Only atoms, which were within 6 Å from the ligand and were solvent accessible, were chosen for building the binding cavity grid. Lastly the protein structure was saved in mol2 format. The ligand chosen for the purpose of defining the binding pocket is a protein-like substrate, which was docked to ETA based on the homology model created by Gismene et al.[19] showing the homolog protein ETE bound to a peptide mimicking the human Dsg1 loop containing the cleavage site.

2.3 Docking and selection of final prospective hits

GOLD was used to dock the compounds during each iteration and in both protocols. All parameters were set to their default values, except for autoscaling, which was set to 0.5, thus allowing an acceptable balance between docking speed and accuracy. The number of generated poses was set to 3 and ChemPLP was used as the scoring function.
These parameter values were kept the same when docking the final library of predicted hits retrieved after running the last iteration of Deep Docking. The 10,000 best scoring compounds from this last library were docked again with parameters allowing more accurate predictions: the autoscale parameter was set to 1, which allows for maximum accuracy and the number of generated poses for each compound was set to 10. After this final docking step, the best scoring 1,000 compounds were selected based on the ligand efficiency (eq. 1) and manually evaluated.

$$\text{ligand efficiency} = \frac{ChemPLP\ score}{number\ of\ heavy\ atoms} \qquad \text{eq. 1}$$

2.4 Deep Docking

To ensure simple and user-friendly execution of the Deep Docking protocol, a Jupyter Notebook[52] was generated. This enabled the step-by-step execution of the right scripts provided by the developers of the protocol, as well as the ligand and protein preparation and docking. The non-automated version of the protocol was chosen as it is more suitable for limited computational resources.[14] A general overview of the workflow is depicted in Figure 2.

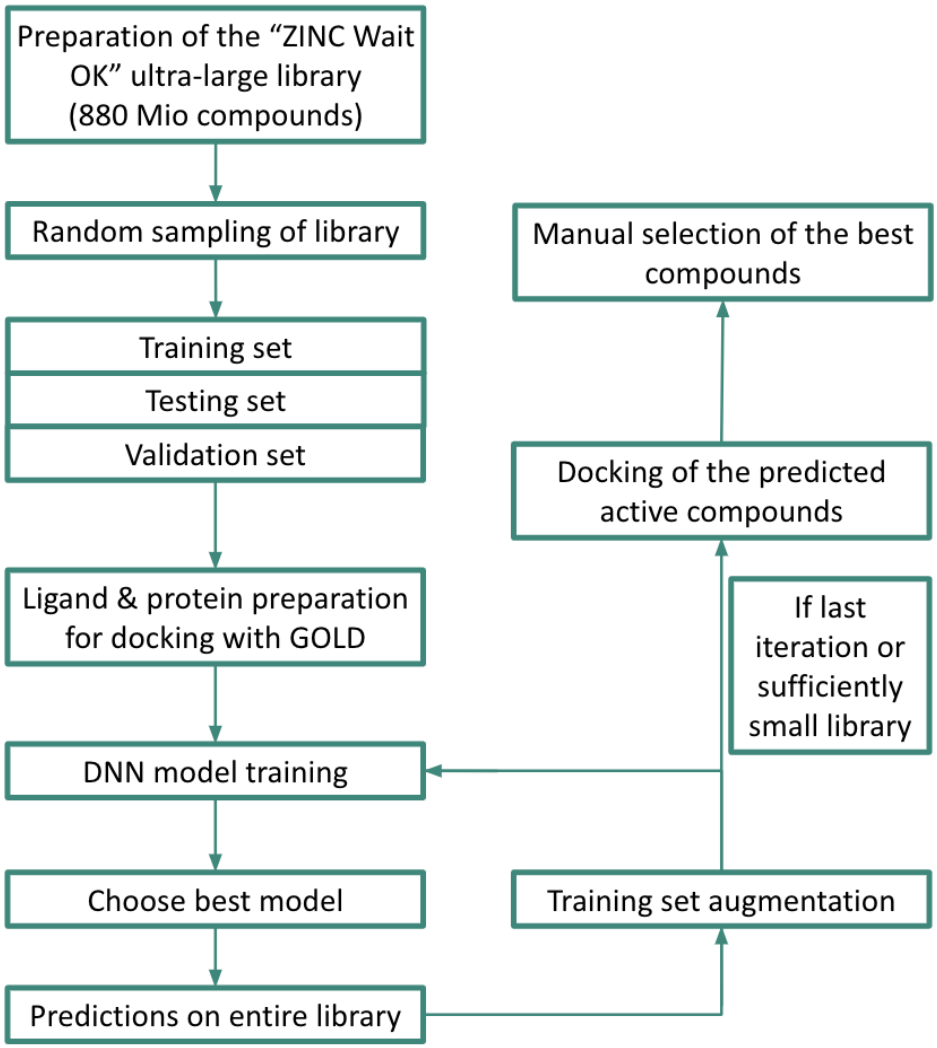

Figure 2: General workflow of the Deep Docking, the first protocol to be adopted. After sampling a random subset of compounds from a prepared library, the compounds are docked and ML models are trained on the docking scores. The best model is chosen and used to classify the entire library. Subsequently, new compounds to be added to the training set in the next iteration are chosen from the pool of prospective hits. Finally, the last pool of prospective hits is actually docked and manually evaluated.

2.4.1 Descriptors and sample size

After the first preparation of the ligands, 1,024-bit circular Morgan fingerprints[35] with radius 2 were calculated for all compounds. Subsequently, compounds for training, testing and validation of the MLP model were selected. For each set 900,000 compounds were randomly chosen. Moreover, overlaps between the sets were eliminated by removing the common compounds. Then, the compounds were docked and, for better compatibility with the original protocol, the GOLD scores were inverted by multiplying them by -1 and the compounds with lower values of the negative scores were considered better.[53] In fact, the protocol was originally designed to use Glide[54] and FRED[55], which produce negative docking scores.[54,55] From the generated docking scores, a cutoff is calculated based on the scores of the validation dataset. As per default, cutoff is set in

order to classify 1% of the compounds as active and 99% as inactive in the first iteration. With every subsequent iteration the percentage of compounds considered active is linearly decreased until reaching 0.01% in the last iteration, meaning the cutoff gets lower with every iteration in order to retrieve more active compounds. The validation and test set are the same in each iteration, however the training set is augmented after the first iteration with 900,000 compounds retrieved from the predicted hits of the previous iteration. In every iteration, compounds overlapping with those included in the sets used in previous iterations are removed.

2.4.2 Model training and selection

The Keras Python library was used for training and building the MLP model. The recommended number of models to generate (24) was chosen. The hyperparameters for the models were chosen from the default hyperparameter grid:

- ➔ dropout = [0.2, 0.5]
- ➔ learning rate = [0.0001]
- ➔ bin-array (number of hidden layers) = [2, 3]
- ➔ classweight = [2, 3]
- ➔ number of units (neurons) = [100, 1500, 2000]

Additionally, by default, a 10-fold oversampling of the minority class was performed and the classweight for the minority class was set to 2 or 3 in order to deal with the highly imbalanced data. Finally, the threshold used to determine if a compound is active or not is chosen based on the threshold needed in order to correctly predict 90% or more of the active compounds in the validation set. This threshold was then also used during inference.

The best model among 24 generated models was chosen based on the full predicted database enrichment (FPDE) value on the test set. The FPDE is calculated as the ratio between the precision (eq. 2) and the random precision (eq. 3).

$$\text{precision} = \frac{TP}{TP * FP} \qquad \text{eq. 2}$$

$$\text{random precision} = \frac{TP_{\ database}}{total\ molecules_{\ database}} \qquad \text{eq. 3}$$

2.5 MolPal

A general overview of the MolPal protocol can is described in Figure 3.

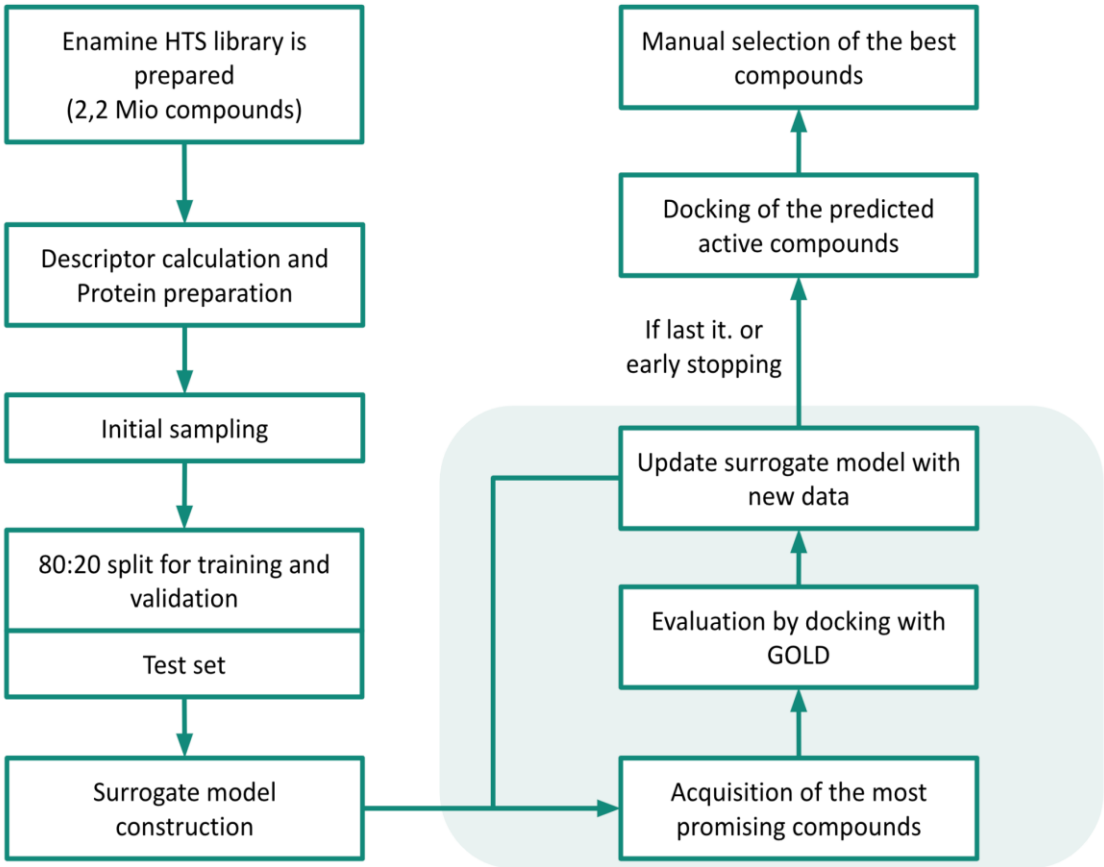

Figure 3: General workflow of the MolPal protocol: A random subset of the chemical library is chosen and docked. A surrogate model is trained on the docking scores and applied to predict the docking scores of the library. From here the Bayesian optimization proc process (highlighted in the green square) begins: an acquisition function is used to acquire new compounds, which are docked and used to update the model. This is done for a set number of iterations.

2.5.1 Ligand preparation, sampling and docking

For timing reasons, only a small study was performed using the MolPal protocol. The Enamine HTS dataset containing 2.1 million compounds was downloaded and prepared as described above using the programs flipper and tautomers. After preparation, the library size consisted of 3.3 million compounds. The D-MPN was the best performing surrogate model as described by the authors of MolPal, therefore, it was adopted in this work.[26] 0.2% of the entire library was randomly chosen for the initial model training, as it was reported to retrieve accurate results.[26] The same threshold is also set for the batch sizes in subsequent iterations.

MolPal only supports pyscreener[56] for performing the docking. Pyscreener exclusively allows docking with the four vina docking softwares, Vina[17], QVina2[57], PSOVina[58] and Smina[59] and with DOCK6[60]. To enable the use of GOLD in MolPal, GOLD was incorporated into pyscreener. So, when GOLD is chosen as the docking software, 3D conformations of the ligands to be docked are generated as described above. Furthermore, the ligands are docked and the best score and conformation is retrieved by parsing the "bestranked" file automatically generated by GOLD for each ligand.

2.5.2 Model training and subsequent iterations

The D-MPN was implemented by PyTourch[61] through PyTourchLightning[62] using the MoleculeModel class from the Chemprop library[63] and employing standard settings. Adam optimization was adopted to train the model and a Noam learning rate scheduler, with initial, maximum and final learning rates of 10^{-4}, 10^{-3} and 10^{-4}, respectively. The network was trained over 50 epochs, the mean square error was used as a loss function, with a batch size of 50. An early stopping based on the validation score was applied with a patience value of 10. The model was trained and then applied to the entire molecule pool. The model generates a predicted mean for each compound and another subset of compounds of the same size as the initial sample size is chosen using greedy as an acquisition metric. The model is updated with the new data and trained on all the observed compounds. This procedure is done for 5 iterations and a list with the top 1% compounds and their predicted scores is retrieved at the end. The top-scoring compounds were compared to the top scoring compounds retrieved when screening the same library using the same settings for GOLD.

2.6 Hardware

The in-house setup used for this work consisted of a Linux computer with CentoOS Stream 8 as operating system, 1 GPU (NVIDIA GeForce RTX 4090 with 24 GB of memory) and 32 CPUs (AMD Ryzen 9 7950X 16-Core Processor). The GPU was used for MLP and D-MPN training and inference and 26 CPUs were used for docking.

3. Results

3.1 Decrease in computational time

Overall, a drastic decrease of computational cost and time for both Deep Docking and MolPal could be seen. The Deep Docking protocol was used to virtually screen over 890 million compounds. Had the compounds been docked on the same workstation as used for this work, it would have taken over 2 years, whereas using the Deep Docking protocol it just took 4 weeks. A more detailed description of the duration for the most important and time-consuming steps can be seen in Table 2.

Table 2: Approximate compute times for the most important steps executed during Deep Docking.

Key step	Time on one workstation
Iteration 1	
Preparation of the 3D conformations	24 h
Docking	51 h
Model training	8 h
Predictions on entire library	7 h
Subsequent iterations	
Preparation of the 3D conformations	8 h
Docking	17 h
Model training	8 h
Predictions on entire library	7 h

As for MolPal, only a relatively small library was used for evaluating the performance of the protocol. Still, a substantial time reduction could be observed. In fact, if 3.3 million compounds were to be docked on the same workstation, it would have taken approximately 3 days to conclude. When using the MolPal protocol, the library could be screened in 10 hours. The duration of the most important steps is described in Table 3.

Table 3: General compute time for the most important steps during one iteration executed during MolPal.

Key step	Time on one workstation
Preparation of the 3D compounds and docking	50 min
Model training	15 min
Predictions on entire library	60 min

3.2 Performance of the protocols

3.2.1 Deep Docking

The performance of the models is evaluated by calculating the area under the receiver operating characteristics (AUROC). The AUROC is suitable for evaluating the performance of classification problems. It quantifies how well the model can distinguish between active and inactive compounds. As can be seen in Figure 4, the AUROC value is increasing after each iteration. The last and best generated model shows an AUROC value of 0.967, which means that the model has a chance of 96.7% for correctly distinguishing between active and inactive compounds. These are encouraging results. However, the highest performance boost can be seen in the second iteration as the AUROC increases by 2.2%. After the second iteration, the AUROC values are still rising but the changes are less significant.

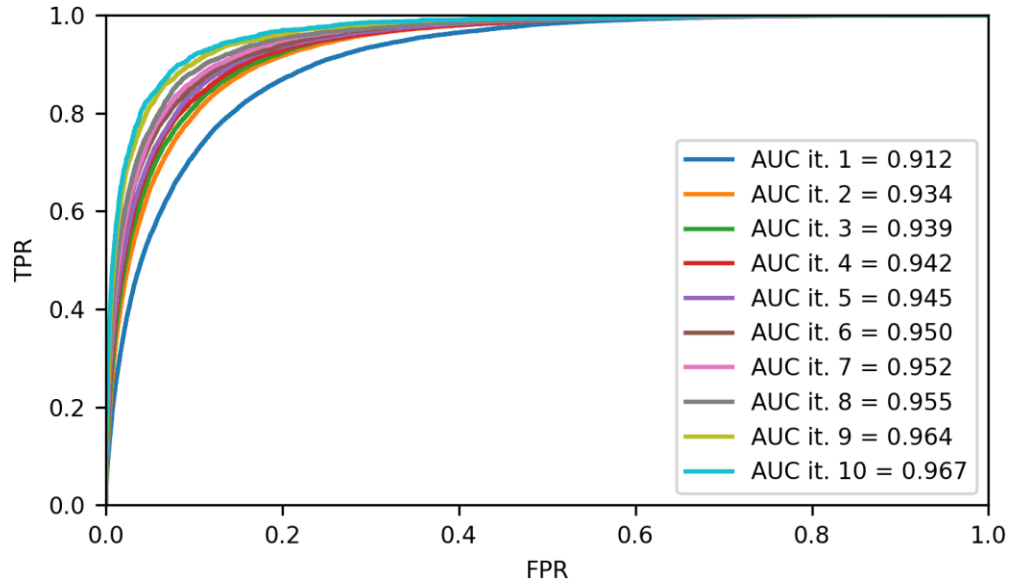

Figure 4: Increasing AUROC values of the best model after each iteration of Deep Docking.

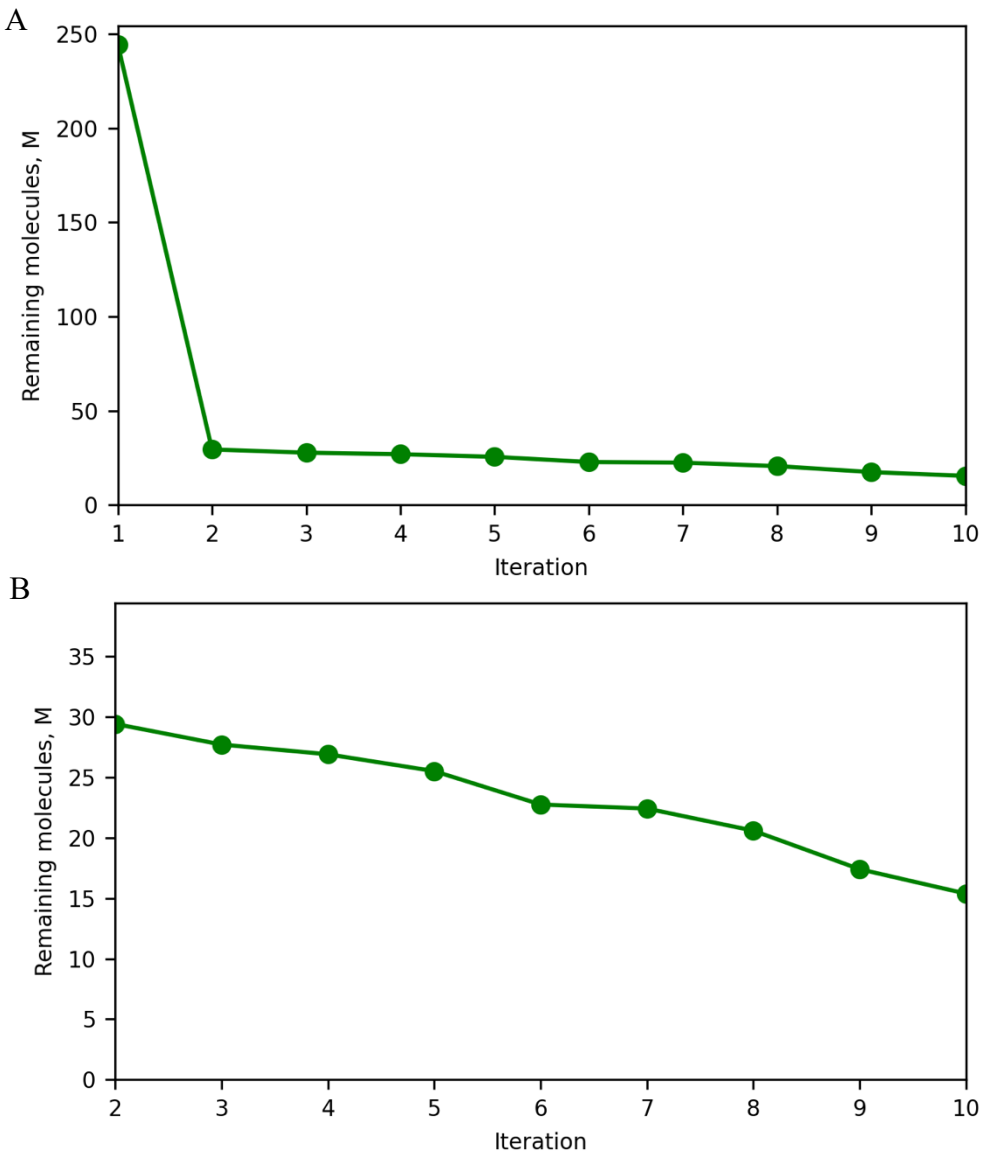

Figure 5: A, Decreasing number of prospective hits in the library after every iteration. B, Decreasing amount of prospective hits in the library from the second to the last iteration of Deep Docking.

The higher performance increase of the model during the second iteration could also be the reason for the steep decrease in the amount of predicted active compounds, as can be observed in Figure 5. Here, after the first iteration, the number of prospective hits predicted by the model amounts to about 250 million compounds. Already after the second iteration, the amount of prospective hits had decreased to about 30 million compounds. In Figure 5a the decrease in the subsequent iterations does not seem significant, however, until the tenth iteration the amount of prospective hits is cut in half when compared to the amount of prospective hits in the second iteration, this can be better observed in Figure 5b.

3.2.2 MolPal

By docking only 1.2% of the 3.3 million compounds, the MolPal protocol was able to capture, under the predicted top 1% of compounds, 102 compounds matching the top-1000 scoring compounds in the library, according to the explicit docking results. These results are not as promising as the ones the authors were able to retrieve.[26] The reason behind this discrepancy might lie in the different acquisition function used or it might be target specific. Further experiments using different surrogate models and acquisitions functions would need to be conducted in order to retrieve a better performance of the protocol.

3.3 Predicted hits

3.3.1 Deep Docking

After execution of the entire protocol, approximately 15 million compounds were retrieved, which had been predicted to be active on ETA by the last Deep Docking model. It took additional 6 days to dock these prospective hits. Through more accurate docking the best 1,000 compounds were selected based on ligand efficiency and the 10 most promising compounds are presented below.

The results are evaluated by considering the crystal structure of a glutamic acid specific serine protease bound to a peptide-like ligand (PDB: 1HPG).[64] The structure is depicted in Figure 6 and the amino acids interacting with the ligand are highlighted. On the one hand, it is clear that some amino acids are conserved in the ETA binding pocket: H57, G193, S195 and H213. On the other hand, the serine in position 190 of this protease is substituted by threonine in ETA. Also, the serine in position 216 is replaced by lysine in ETA. The co-crystallized peptide forms hydrogen bonds with all the aforementioned amino acids, indicating their importance for protein-ligand interactions. It is assumed that these interactions play a key role when targeting ETA, as it is also a glutamic acid specific serine protease.

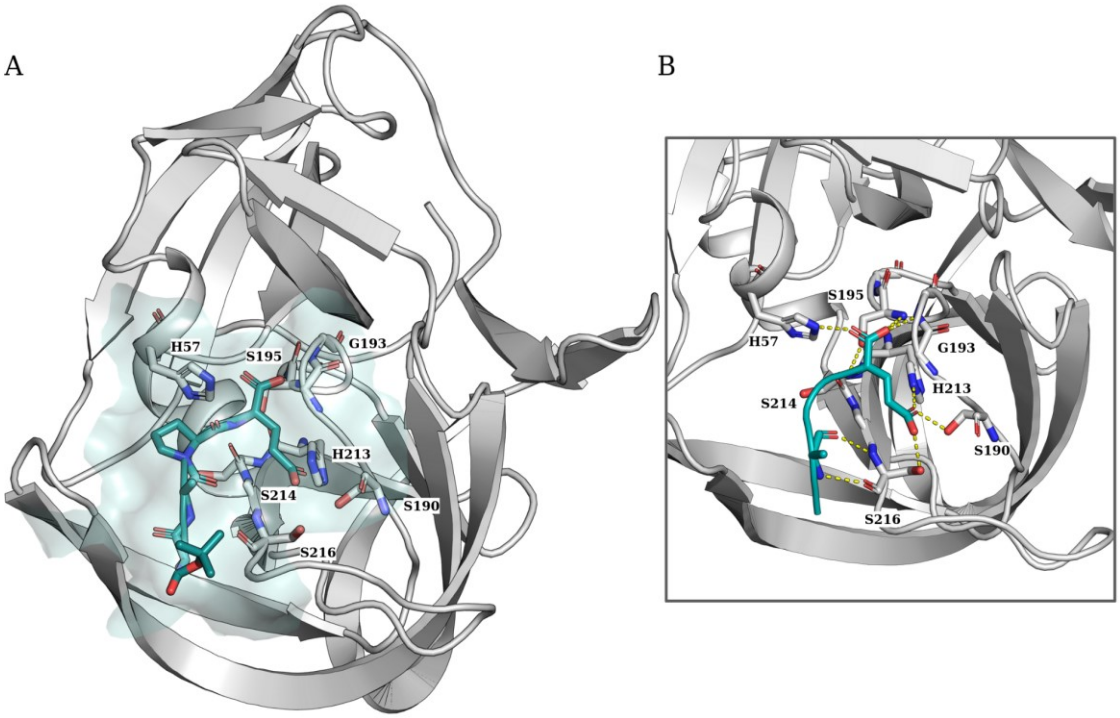

Figure 6: PDB 1HPG64 showing a glutamic acid specific serine protease bound to its ligand. A, the protein in cartoon and the peptide ligand in cyan sticks, with the amino acids forming interactions highlighted, as well as the surface of the binding site. B, zoom-in view of the binding site with H-bonds shown as yellow dashed lines and the amino acids forming the interactions highlighted. Here the residues of the ligand, which do not perform interactions are hidden for better understanding.

The first five compounds (Figures 7-16) share in general very similar patterns: 1) a carboxylic acid bound to an aromatic 5-membered ring or to an aliphatic chain on the one side of the structure, 2) on the opposite end they all have a planar aromatic residue, and 3) the two aforementioned sub-structures are connected by a more flexible and polar linker. In all instances, the carboxylic acid is predicted to establish H-bond interactions and form salt bridges with the imidazole nitrogen of H213 and the amine group of K216. The polar moieties in the linker interact with S195 and G193 by forming H-bond interactions.

In compound A, an additional H-bond can be observed with T40 as shown in Figure 7.

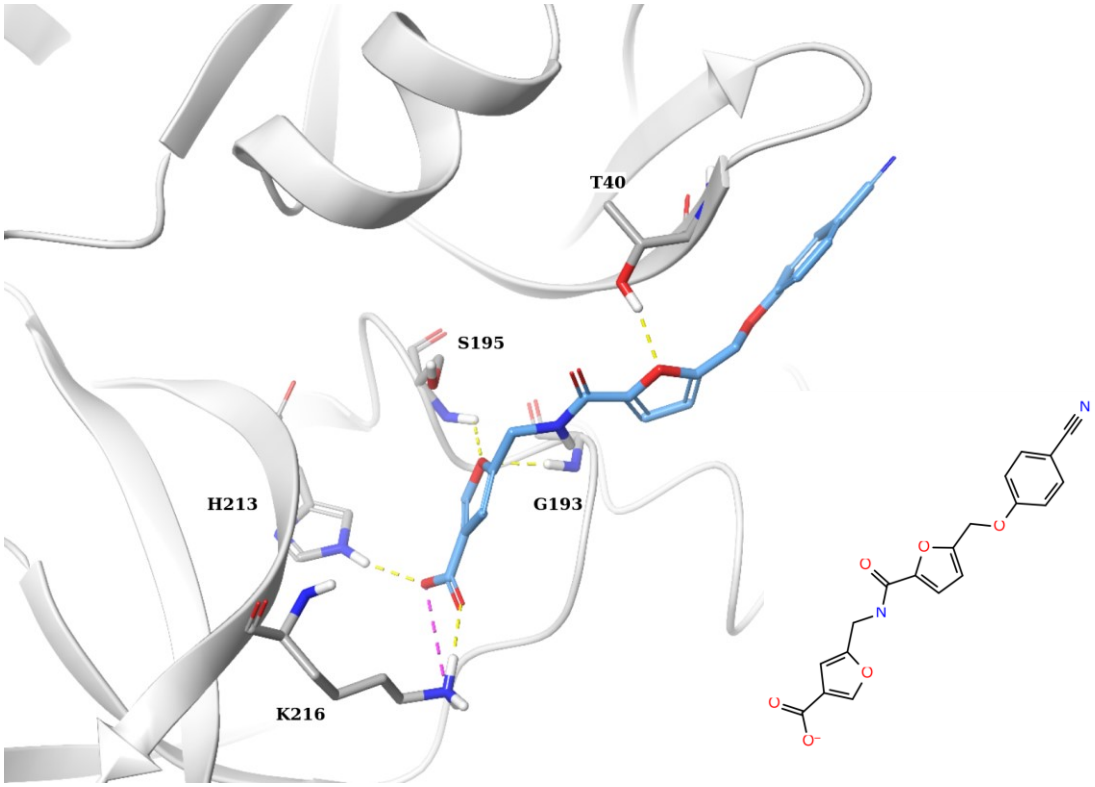

Figure 7: ETA (1AGJ) represented as a cartoon, with bound compound A, shown as blue sticks as well as the amino acids forming interactions represented as gray sticks. The predicted interactions are visible as dashed lines: H-bonds in yellow, salt bridge in pink.

Also, in compounds B (Figure 8) the same interaction with T40 is established with the nitrogen in the pyrazole ring.

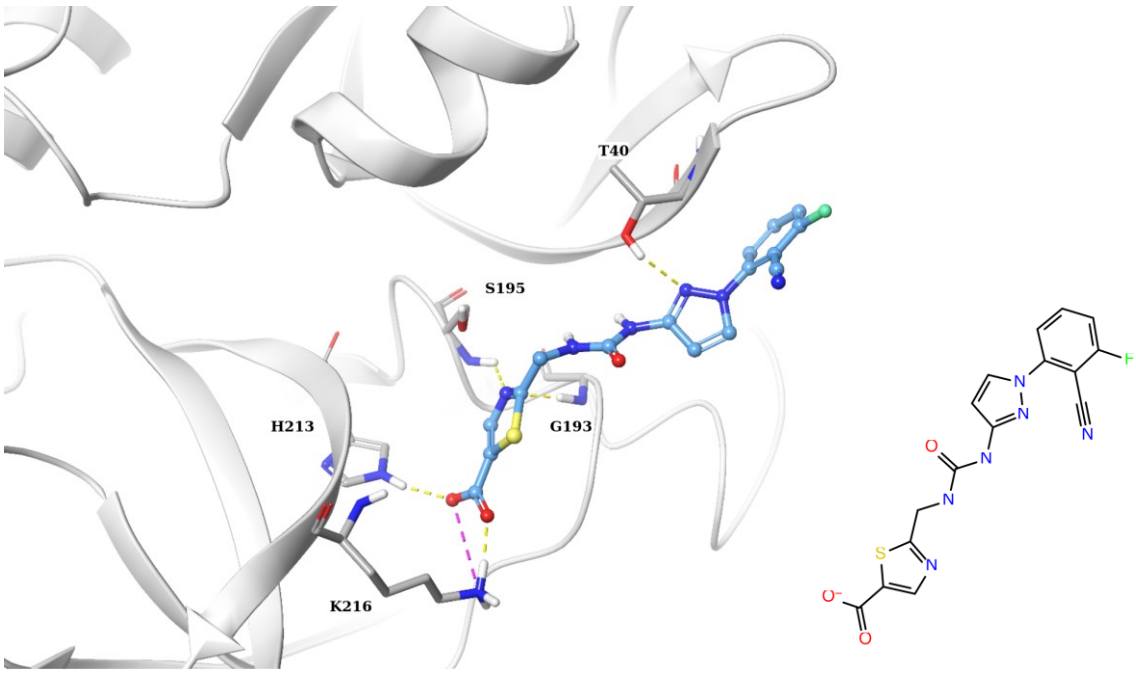

Figure 8: ETA (1AGJ) represented as a cartoon, with bound compound B, shown as blue sticks as well as the amino acids forming interactions represented as gray sticks. The predicted interactions are visible as dashed lines: H-bonds in yellow, salt bridge in pink.

As for compound C, the outer phenyl group is forming π-π stacking interactions with the phenyl group of F34 (Figure 9).

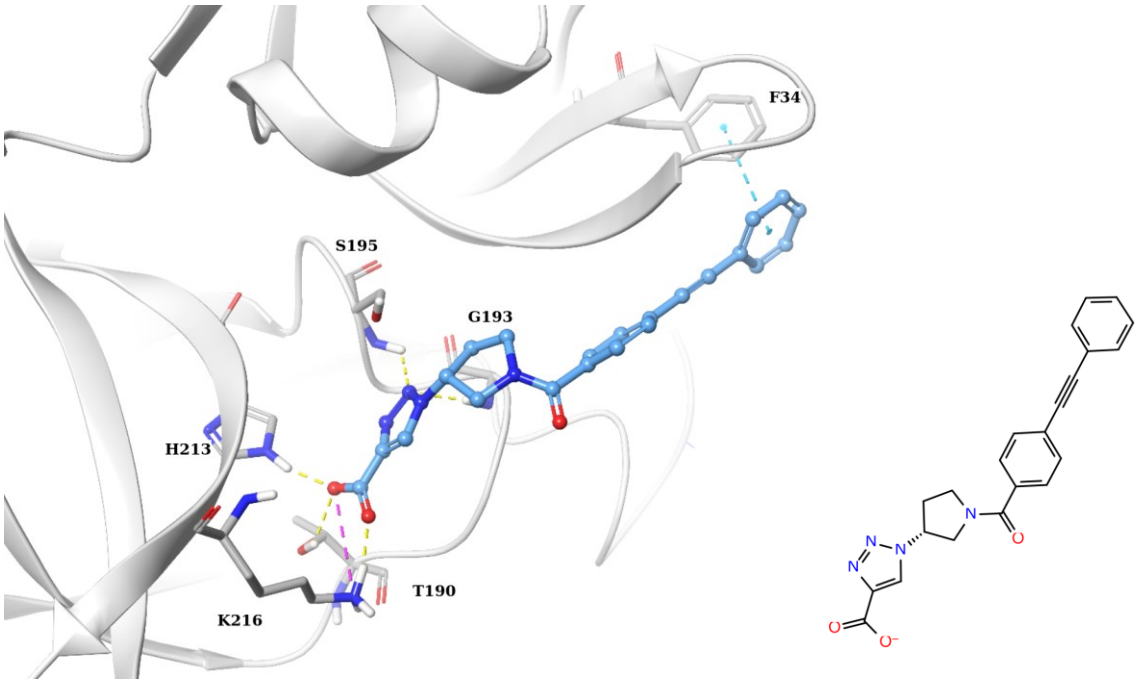

Figure 9: ETA (1AGJ) represented as a cartoon, with bound compound C, shown as blue sticks as well as the amino acids forming interactions represented as gray sticks. The predicted interactions are visible as dashed lines: H-bonds in yellow, salt bridge in pink, π-stacking in cyano.

Compound D (Figure 10) displays additional interactions with the backbone oxygen of S214 (hydrogen bond) as well as with the nitrogen in G28 (halogen bond).

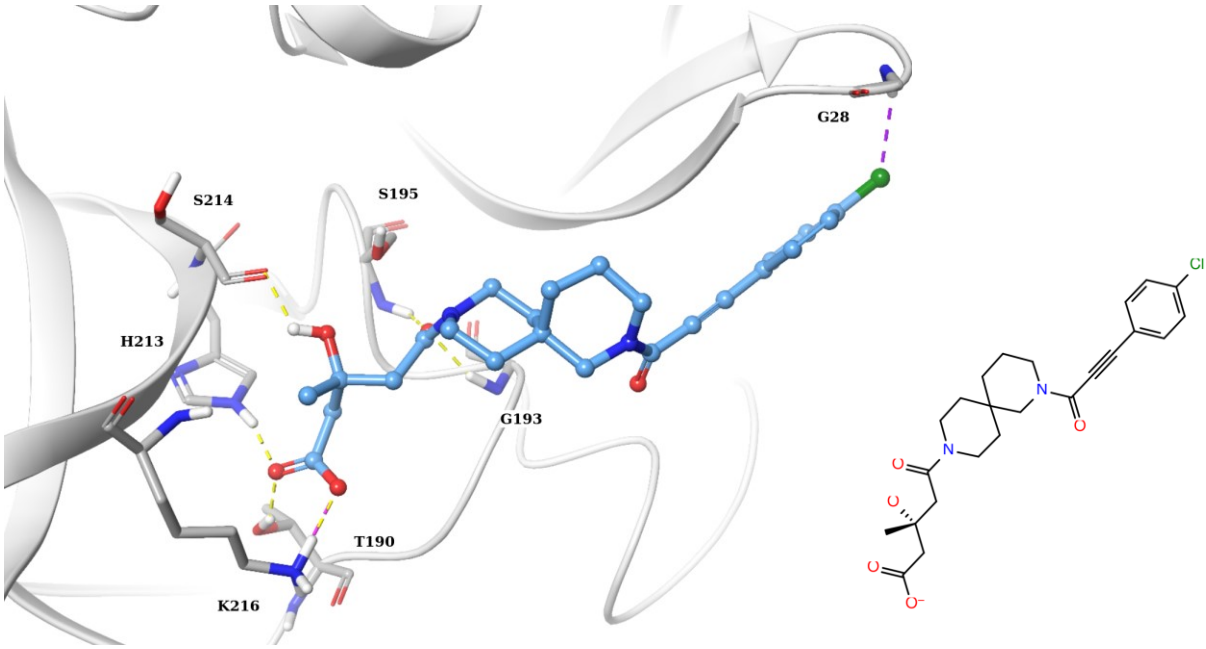

Figure 10: ETA (1AGJ) represented as a cartoon, with bound compound D, shown as blue sticks as well as the amino acids forming interactions represented as gray sticks. The predicted interactions are visible as dashed lines: H-bonds in yellow, salt bridge in pink, halogen bond in purple.

The last compound with this scaffold is compound E. It shows interactions with the most important amino acids as well as an additional H-bond with the nitrogen in S41 as depicted in Figure 11.

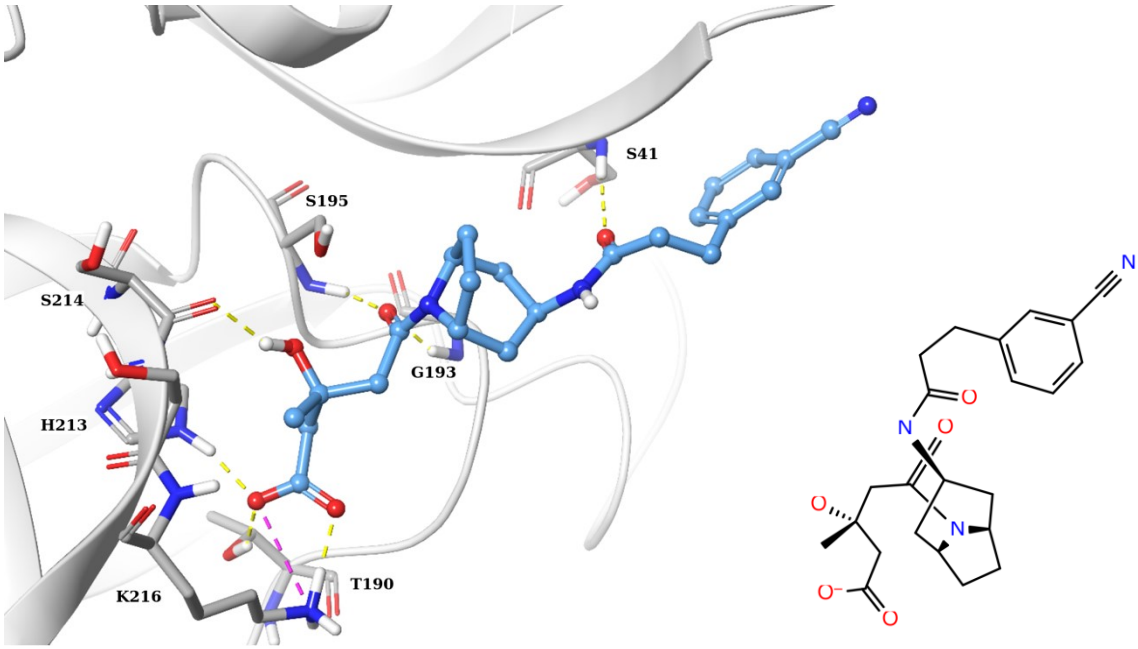

Figure 11: ETA (1AGJ) represented as a cartoon, with bound compound E, shown as blue sticks as well as the amino acids forming interactions represented as gray sticks. The predicted interactions are visible as dashed lines: H-bonds in yellow, salt bridge in pink.

The second set of compounds is similar to the ones which were discussed above. The main difference lies in the nitro functional group, which is present instead of the carboxylic acid. Nitro groups are known to be mutagenic and cause genotoxicity as they are reduced *in vivo* into nitrosamines and hydroxylamines.[65] Nevertheless, these compounds were chosen as they present interesting interactions and if bioactivity were to be detected, the toxic moieties could be exchanged by bioisosteres.

For compound F, interactions with the main amino acids H213, K216, T190 and G193 can be observed in Figure 12, as well as two additional interactions made by both nitrogens of the thiourea functional group with the two oxygens present in S41.

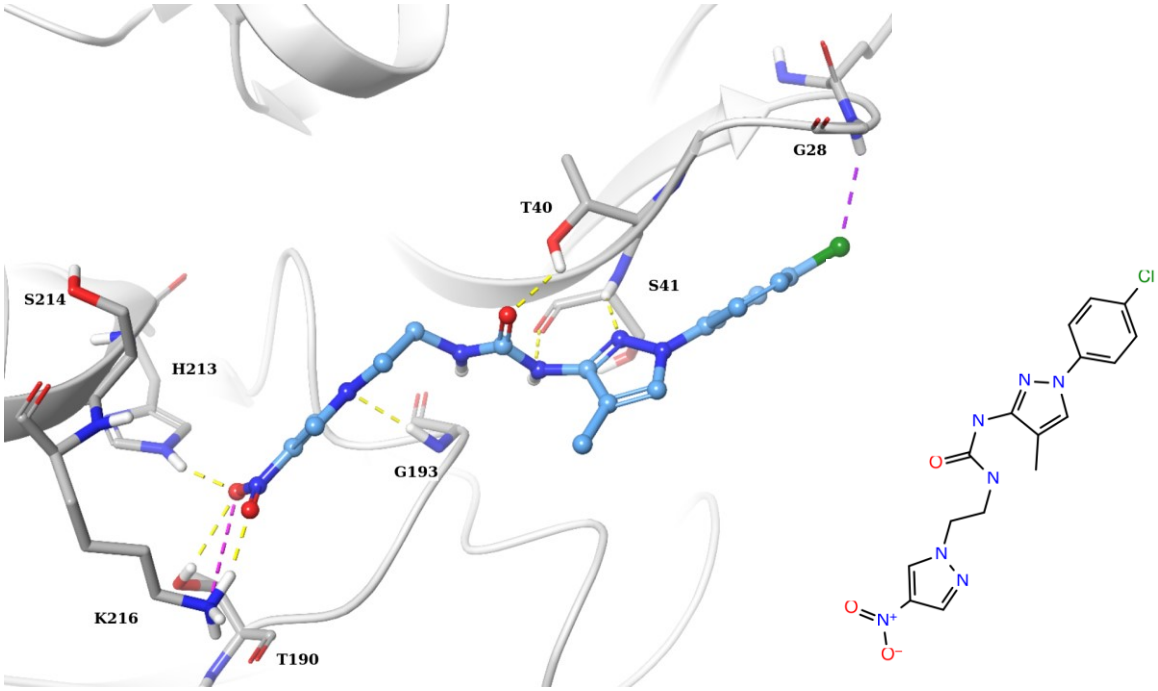

Figure 12: ETA (1AGJ) represented as a cartoon, with bound compound F, shown as blue sticks as well as the amino acids forming interactions represented as gray sticks. The predicted interactions are visible as dashed lines: H-bonds in yellow, salt bridge in pink, halogen interactions in purple.

Compound G shows a halogen interaction with G28 as seen in compound D in addition to the already discussed interactions (Figure 13).

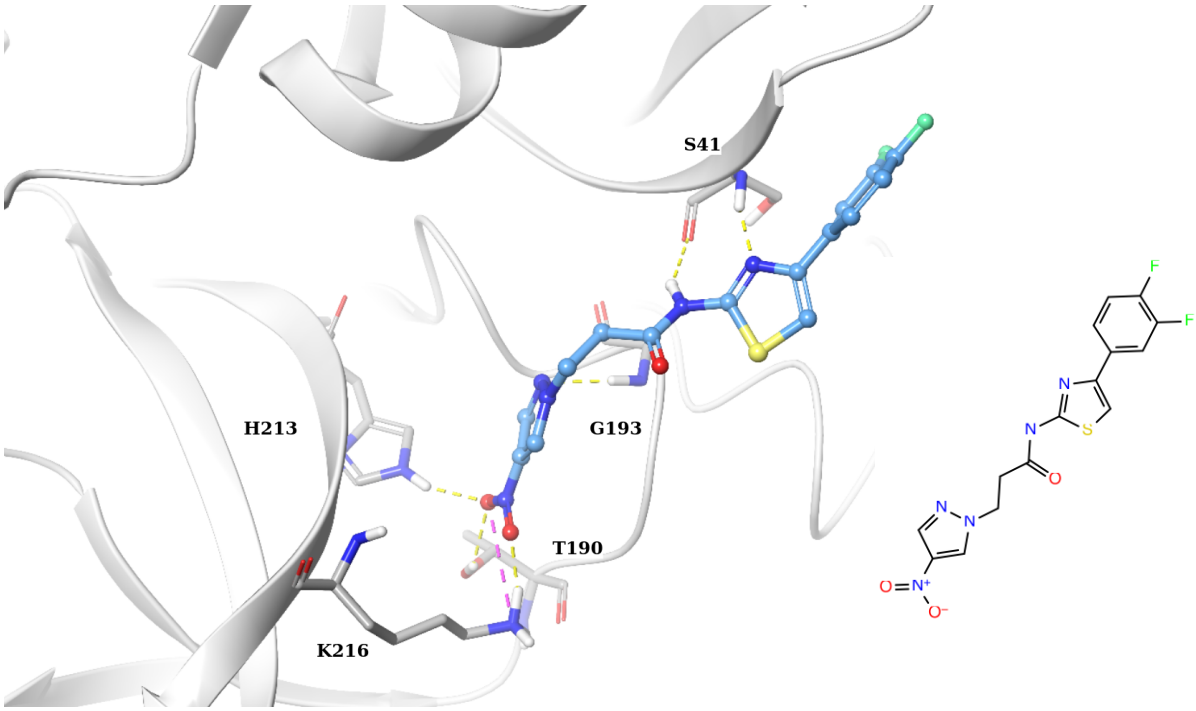

Figure 13: ETA (1AGJ) represented as a cartoon, with bound compound G, shown as blue sticks as well as the amino acids forming interactions represented as gray sticks. The predicted interactions are visible as dashed lines: H-bonds in yellow, salt bridge in pink.

The last of the selected molecules to contain a nitro group is compound H (Figure 14). Here again the halogen interaction with the same glycine as above can be seen, indicating that it might be an additional important interaction to exploit.

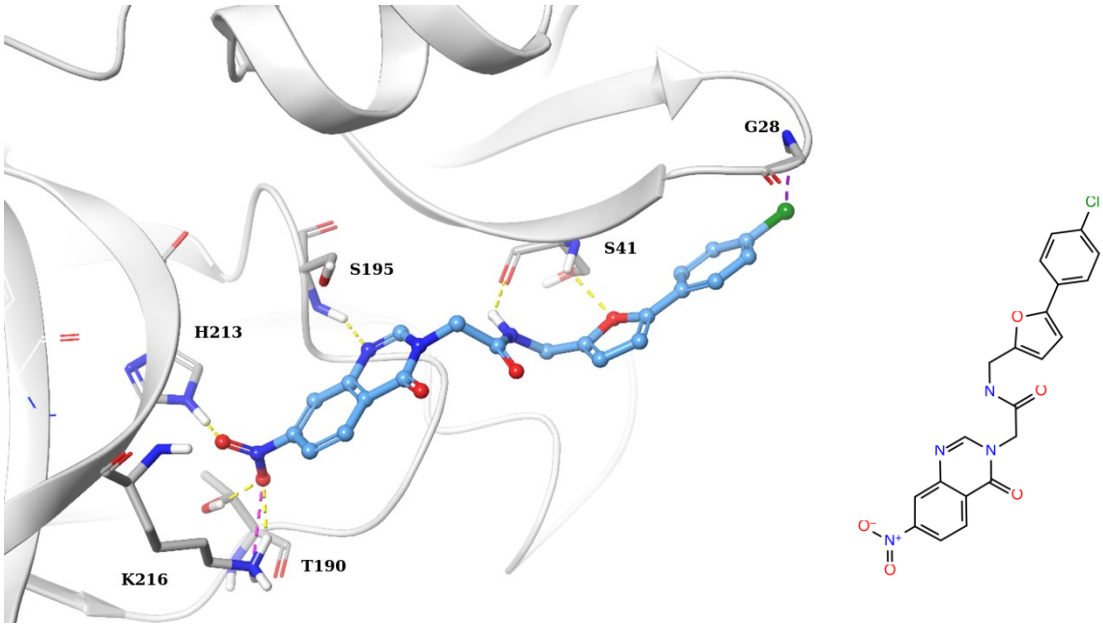

Figure 14: ETA (1AGJ) represented as a cartoon, with bound compound H, shown as blue sticks as well as the amino acids forming interactions represented as gray sticks. The predicted interactions are visible as dashed lines: H-bonds in yellow, salt bridge in pink, halogen interactions in purple.

Compound I shows an alcohol functional group instead of the carboxylic acid or nitro group. By exchanging this group the interactions with either K216 and/or T190, which were seen in all previous compounds are lost. This could have negative repercussions for ligand affinity as salt-bridges are powerful interactions which might play an important role in bringing the ligand closer to the protein. Nevertheless, this compound has the same scaffold as described above and shows hydrogen-bond interactions with four amino acids (H213, S195, G193, S41) as depicted in Figure 15.

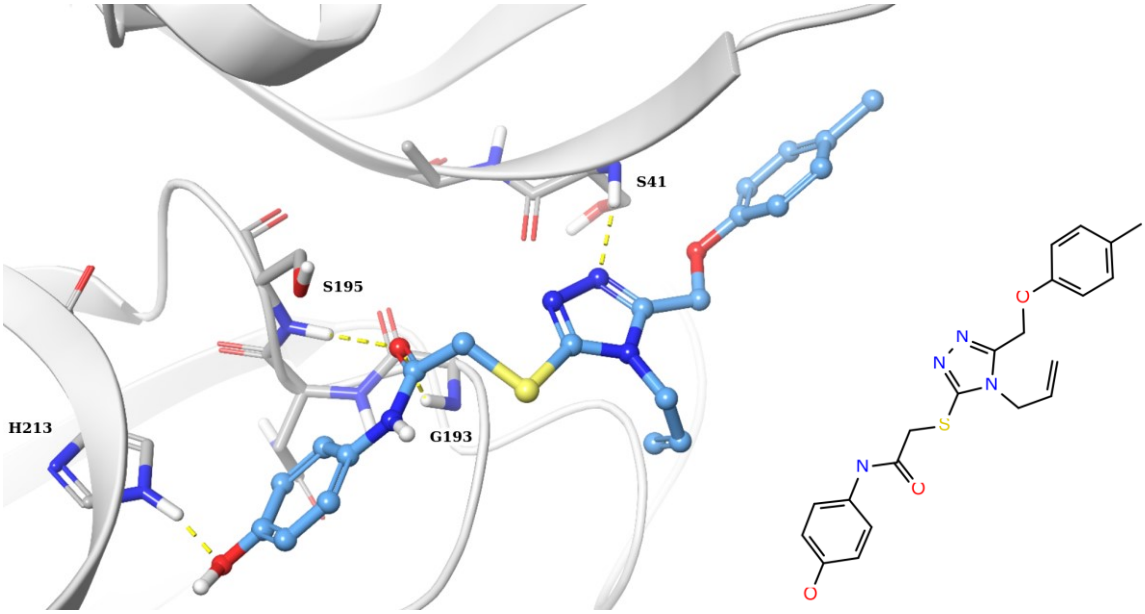

Figure 15: ETA (1AGJ) represented as a cartoon, with bound compound I, shown as blue sticks as well as the amino acids forming interactions represented as gray sticks. The predicted interactions are visible as dashed lines: H-bonds in yellow.

The last compound retrieved with the Deep Docking protocol is compound J (Figure 16). Here a carboxylic acid interacting with H213, K216 and T190 is present. However, the scaffold differs from that of the above-mentioned compounds as there is no planar moiety at the other end from the carboxylic acid. However, all important interactions are fulfilled.

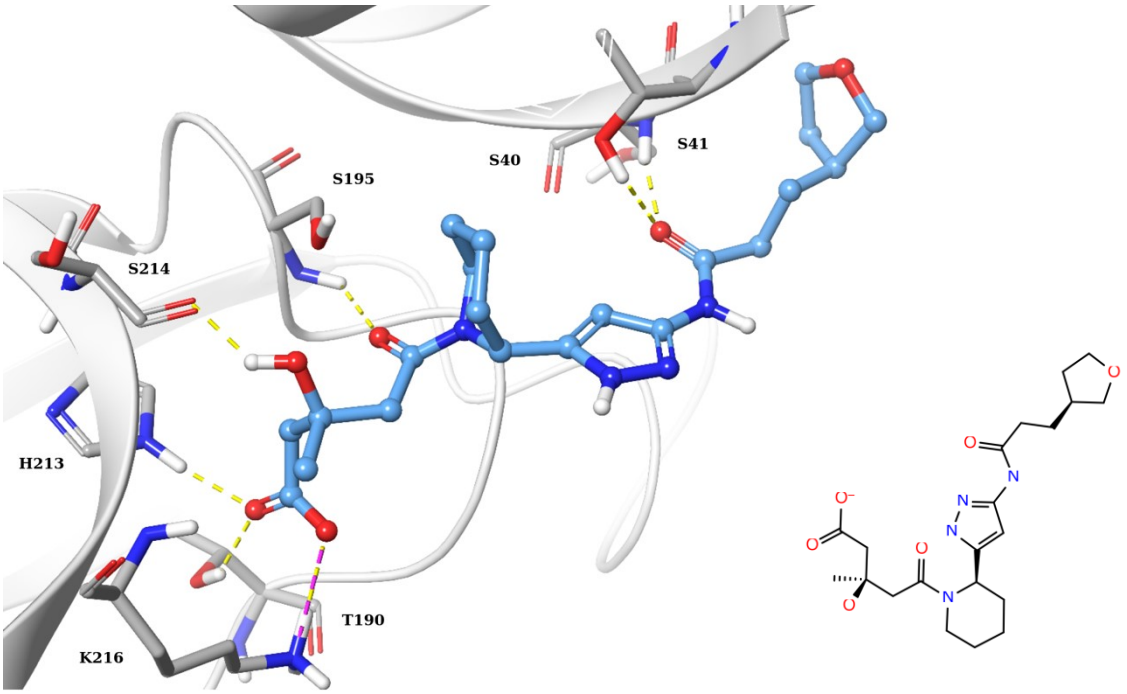

Figure 16: ETA (1AGJ) represented as a cartoon, with bound compound J, shown as blue sticks as well as the amino acids forming interactions represented as gray sticks. The predicted interactions are visible as dashed lines: H-bonds in yellow, salt bridge in pink.

4. Conclusions and outlook

In this work, different protocols for DL accelerated docking-based VS were presented and two of the most promising ones were explained in more detail, modified in order to accommodate docking with GOLD, and applied to dataset sizes of 2.1 million and 880 million compounds, respectively. A thorough evaluation of the Deep Docking and MolPal protocols demonstrates that they both substantially decrease time and the need of computational resources for VS. Indeed, when applying Deep Docking to the ultra-large library, screening was 14 times faster than conventional docking on the same workstation. A similar tendency was observed for MolPal, where the screening procedure was 7 times faster than conventional docking.

In addition to the increased efficiency, the Deep Docking protocol also performed well with respect to hit selection, as compounds showing important patterns and interactions for the protein-ligand binding were detected. Also, diverse scaffolds with similar interactions with the target could be retrieved. These results led to the identification of virtual hits for the yet undrugged *Staphylococcus aureus* ETA.

However, during the analysis of the Deep Docking results a large number of large, hydrophobic compounds were detected. These findings correspond to the already observed bias of docking scores, which tend to predict molecules with high molecular weight as being more active.[66] The MLP model seems to catch this bias and manifest it in the results. For future inquiries, the chosen library needs to be carefully filtered in order to exclude molecules, which are too large and lipophilic. For the scope of this work, this step was omitted as a library containing as many compounds as possible needed to be screened for better evaluation of the protocol performance. This issue was partially mitigated by selecting the last set of active compounds based on their ligand efficiency before visual inspection and manual selection.

In spite of the previously discussed issues, as long as the exact set of top-scoring compounds is not needed, these protocols represent an extraordinary alternative to conventional ultra-large virtual screening campaigns. Additionally, these tools could also help reduce the cost of *in vitro* HTS campaigns. Thus, improving time and cost expenses in the early stages of drug discovery.

Lastly, for future applications, these protocols could be combined with other structure-based methods, such as molecular dynamics and free energy calculations. In addition, the use of other surrogate models or docking algorithms could be explored to gain a more customized protocol. Instead of the docking score, which seems to show a bias toward larger compounds, the DL models docking scores.

References

(1) Irwin, J. J.; Tang, K. G.; Young, J.; Dandarchuluun, C.; Wong, B. R.; Khurelbaatar, M.; Moroz, Y. S.; Mayfield, J.; Sayle, R. A. ZINC20—A Free Ultralarge-Scale Chemical Database for Ligand Discovery. *J. Chem. Inf. Model.* **2020**, *60* (12), 6065–6073. https://doi.org/10.1021/acs.jcim.0c00675.

(2) Tingle, B. I.; Tang, K. G.; Castanon, M.; Gutierrez, J. J.; Khurelbaatar, M.; Dandarchuluun, C.; Moroz, Y. S.; Irwin, J. J. ZINC-22─A Free Multi-Billion-Scale Database of Tangible Compounds for Ligand Discovery. *J. Chem. Inf. Model.* **2023**, *63* (4), 1166–1176. https://doi.org/10.1021/acs.jcim.2c01253.

(3) Bohacek, R. S.; McMartin, C.; Guida, W. C. The Art and Practice of Structure-Based Drug Design: A Molecular Modeling Perspective. *Med. Res. Rev.* **1996**, *16* (1), 3–50. https://doi.org/10.1002/(SICI)1098-1128(199601)16:1<3::AID-MED1>3.0.CO;2-6.

(4) *REAL Database - Enamine*. https://enamine.net/compound-collections/real-compounds/real-database (accessed 2023-03-27).

(5) *REAL Space - Enamine*. https://enamine.net/compound-collections/real-compounds/real-space-navigator (accessed 2023-03-27).

(6) Lyu, J.; Wang, S.; Balius, T. E.; Singh, I.; Levit, A.; Moroz, Y. S.; O'Meara, M. J.; Che, T.; Algaa, E.; Tolmachova, K.; Tolmachev, A. A.; Shoichet, B. K.; Roth, B. L.; Irwin, J. J. Ultra-Large Library Docking for Discovering New Chemotypes. *Nature* **2019**, *566* (7743), 224–229. https://doi.org/10.1038/s41586-019-0917-9.

(7) Gorgulla, C.; Boeszoermenyi, A.; Wang, Z.-F.; Fischer, P. D.; Coote, P. W.; Padmanabha Das, K. M.; Malets, Y. S.; Radchenko, D. S.; Moroz, Y. S.; Scott, D. A.; Fackeldey, K.; Hoffmann, M.; Iavniuk, I.; Wagner, G.; Arthanari, H. An Open-Source Drug Discovery Platform Enables Ultra-Large Virtual Screens. *Nature* **2020**, *580* (7805), 663–668. https://doi.org/10.1038/s41586-020-2117-z.

(8) Gorgulla, C. Recent Developments in Structure-Based Virtual Screening Approaches. arXiv November 6, 2022. http://arxiv.org/abs/2211.03208 (accessed 2023-03-27).

(9) Scior, T.; Bender, A.; Tresadern, G.; Medina-Franco, J. L.; Martínez-Mayorga, K.; Langer, T.; Cuanalo-Contreras, K.; Agrafiotis, D. K. Recognizing Pitfalls in Virtual Screening: A Critical Review. *J. Chem. Inf. Model.* **2012**, *52* (4), 867–881. https://doi.org/10.1021/ci200528d.

(10) Jahn, A.; Hinselmann, G.; Fechner, N.; Zell, A. Optimal Assignment Methods for Ligand-Based Virtual Screening. *J. Cheminformatics* **2009**, *1* (1), 14. https://doi.org/10.1186/1758-2946-1-14.

(11) Alon, A.; Lyu, J.; Braz, J. M.; Tummino, T. A.; Craik, V.; O'Meara, M. J.; Webb, C. M.; Radchenko, D. S.; Moroz, Y. S.; Huang, X.-P.; Liu, Y.; Roth, B. L.; Irwin, J. J.; Basbaum, A. I.; Shoichet, B. K.; Kruse, A. C. Structures of the Σ2 Receptor Enable Docking for Bioactive Ligand Discovery. *Nature* **2021**, *600* (7890), 759–764. https://doi.org/10.1038/s41586-021-04175-x.

(12) Kaplan, A. L.; Confair, D. N.; Kim, K.; Barros-Álvarez, X.; Rodriguiz, R. M.; Yang, Y.; Kweon, O. S.; Che, T.; McCorvy, J. D.; Kamber, D. N.; Phelan, J. P.; Martins, L. C.; Pogorelov, V. M.; DiBerto, J. F.; Slocum, S. T.; Huang, X.-P.; Kumar, J. M.; Robertson, M. J.; Panova, O.; Seven, A. B.; Wetsel, A. Q.; Wetsel, W. C.; Irwin, J. J.; Skiniotis, G.; Shoichet, B. K.; Roth, B. L.; Ellman, J. A. Bespoke Library Docking for 5-HT2A Receptor Agonists with Antidepressant Activity. *Nature* **2022**, *610* (7932), 582–591. https://doi.org/10.1038/s41586-022-05258-z.

(13) Bajorath, J.; Chávez-Hernández, A. L.; Duran-Frigola, M.; Fernández-de Gortari, E.; Gasteiger, J.; López-López, E.; Maggiora, G. M.; Medina-Franco, J. L.; Méndez-

Lucio, O.; Mestres, J.; Miranda-Quintana, R. A.; Oprea, T. I.; Plisson, F.; Prieto-Martínez, F. D.; Rodríguez-Pérez, R.; Rondón-Villarreal, P.; Saldívar-Gonzalez, F. I.; Sánchez-Cruz, N.; Valli, M. Chemoinformatics and Artificial Intelligence Colloquium: Progress and Challenges in Developing Bioactive Compounds. *J. Cheminformatics* **2022**, *14* (1). https://doi.org/10.1186/s13321-022-00661-0.

(14) Gentile, F.; Yaacoub, J. C.; Gleave, J.; Fernandez, M.; Ton, A.-T.; Ban, F.; Stern, A.; Cherkasov, A. Artificial Intelligence–Enabled Virtual Screening of Ultra-Large Chemical Libraries with Deep Docking. *Nat. Protoc.* **2022**, *17* (3), 672–697. https://doi.org/10.1038/s41596-021-00659-2.

(15) Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Žídek, A.; Potapenko, A.; Bridgland, A.; Meyer, C.; Kohl, S. A. A.; Ballard, A. J.; Cowie, A.; Romera-Paredes, B.; Nikolov, S.; Jain, R.; Adler, J.; Back, T.; Petersen, S.; Reiman, D.; Clancy, E.; Zielinski, M.; Steinegger, M.; Pacholska, M.; Berghammer, T.; Bodenstein, S.; Silver, D.; Vinyals, O.; Senior, A. W.; Kavukcuoglu, K.; Kohli, P.; Hassabis, D. Highly Accurate Protein Structure Prediction with AlphaFold. *Nature* **2021**, *596* (7873), 583–589. https://doi.org/10.1038/s41586-021-03819-2.

(16) McNutt, A. T.; Francoeur, P.; Rishal, A.; Tomohide, M.; Rocco, M.; Matthew, R.; Jocelyn, S.; Ryan, K. D.; Link to external site, this link will open in a new window. GNINA 1.0: Molecular Docking with Deep Learning. *J. Cheminformatics* **2021**, *13* (1). https://doi.org/10.1186/s13321-021-00522-2.

(17) Trott, O.; Olson, A. J. AutoDock Vina: Improving the Speed and Accuracy of Docking with a New Scoring Function, Efficient Optimization, and Multithreading. *J. Comput. Chem.* **2010**, *31* (2), 455–461. https://doi.org/10.1002/jcc.21334.

(18) Bukowski, M.; Wladyka, B.; Dubin, G. Exfoliative Toxins of Staphylococcus Aureus. *Toxins* **2010**, *2* (5), 1148–1165. https://doi.org/10.3390/toxins2051148.

(19) Gismene, C.; Hernández González, J. E.; Santisteban, A. R. N.; Ziem Nascimento, A. F.; dos Santos Cunha, L.; de Moraes, F. R.; de Oliveira, C. L. P.; Oliveira, C. C.; Jocelan Scarin Provazzi, P.; Pascutti, P. G.; Arni, R. K.; Barros Mariutti, R. Staphylococcus Aureus Exfoliative Toxin E, Oligomeric State and Flip of P186: Implications for Its Action Mechanism. *Int. J. Mol. Sci.* **2022**, *23* (17), 9857. https://doi.org/10.3390/ijms23179857.

(20) Imanishi, I.; Nicolas, A.; Caetano, A.-C. B.; Castro, T. L. de P.; Tartaglia, N. R.; Mariutti, R.; Guédon, E.; Even, S.; Berkova, N.; Arni, R. K.; Seyffert, N.; Azevedo, V.; Nishifuji, K.; Le Loir, Y. Exfoliative Toxin E, a New Staphylococcus Aureus Virulence Factor with Host-Specific Activity. *Sci. Rep.* **2019**, *9* (1), 16336. https://doi.org/10.1038/s41598-019-52777-3.

(21) Cavarelli, J.; Prévost, G.; Bourguet, W.; Moulinier, L.; Chevrier, B.; Delagoutte, B.; Bilwes, A.; Mourey, L.; Rifai, S.; Piémont, Y.; Moras, D. The Structure of Staphylococcus Aureus Epidermolytic Toxin A, an Atypic Serine Protease, at 1.7 Å Resolution. *Structure* **1997**, *5* (6), 813–824. https://doi.org/10.1016/S0969-2126(97)00235-9.

(22) Svensson, F.; Norinder, U.; Bender, A. Improving Screening Efficiency through Iterative Screening Using Docking and Conformal Prediction. *J. Chem. Inf. Model.* **2017**, *57* (3), 439–444. https://doi.org/10.1021/acs.jcim.6b00532.

(23) Ahmed, L.; Link to external site, this link will open in a new window; Georgiev, V.; Capuccini, M.; Toor, S.; Schaal, W.; Laure, E.; Spjuth, O. Efficient Iterative Virtual Screening with Apache Spark and Conformal Prediction. *J. Cheminformatics* **2018**, *10* (1), 8. https://doi.org/10.1186/s13321-018-0265-z.

(24) Gentile, F.; Agrawal, V.; Hsing, M.; Ton, A.-T.; Ban, F.; Norinder, U.; Gleave, M. E.; Cherkasov, A. Deep Docking: A Deep Learning Platform for Augmentation of Structure Based Drug Discovery. *ACS Cent. Sci.* **2020**, *6* (6), 939–949. https://doi.org/10.1021/acscentsci.0c00229.

(25) Mehta, S.; Laghuvarapu, S.; Pathak, Y.; Sethi, A.; Alvala, M.; Deva Priyakumar, U. MEMES: Machine Learning Framework for Enhanced MolEcular Screening.

Chem. Sci. **2021**, *12* (35), 11710–11721. https://doi.org/10.1039/D1SC02783B.
(26) Graff, D. E.; Shakhnovich, E. I.; Coley, C. W. Accelerating High-Throughput Virtual Screening through Molecular Pool-Based Active Learning. *Chem. Sci.* **2021**, *12* (22), 7866–7881. https://doi.org/10.1039/D0SC06805E.
(27) Yang, Y.; Yao, K.; Repasky, M. P.; Leswing, K.; Abel, R.; Shoichet, B. K.; Jerome, S. V. Efficient Exploration of Chemical Space with Docking and Deep Learning. *J. Chem. Theory Comput.* **2021**, *17* (11), 7106–7119. https://doi.org/10.1021/acs.jctc.1c00810.
(28) Kalliokoski, T. Machine Learning Boosted Docking (HASTEN): An Open-source Tool To Accelerate Structure-based Virtual Screening Campaigns. *Mol. Inform.* **2021**, *40* (9), 2100089. https://doi.org/10.1002/minf.202100089.
(29) Martin, L. J. State of the Art Iterative Docking with Logistic Regression and Morgan Fingerprints.
(30) *RDKit*. https://www.rdkit.org/ (accessed 2023-08-28).
(31) Mysinger, M. M.; Carchia, M.; Irwin, John. J.; Shoichet, B. K. Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and Decoys for Better Benchmarking. *J. Med. Chem.* **2012**, *55* (14), 6582–6594. https://doi.org/10.1021/jm300687e.
(32) Li, W. A Fast Clustering Algorithm for Analyzing Highly Similar Compounds of Very Large Libraries. *J. Chem. Inf. Model.* **2006**, *46* (5), 1919–1923. https://doi.org/10.1021/ci0600859.
(33) Graff, D. E.; Aldeghi, M.; Morrone, J. A.; Jordan, K. E.; Pyzer-Knapp, E. O.; Coley, C. W. Self-Focusing Virtual Screening with Active Design Space Pruning. *J. Chem. Inf. Model.* **2022**, *62* (16), 3854–3862. https://doi.org/10.1021/acs.jcim.2c00554.
(34) Dixon, S. L.; Duan, J.; Smith, E.; Von Bargen, C. D.; Sherman, W.; Repasky, M. P. AutoQSAR: An Automated Machine Learning Tool for Best-Practice Quantitative Structure-Activity Relationship Modeling. *Future Med. Chem.* **2016**, *8* (15), 1825–1839. https://doi.org/10.4155/fmc-2016-0093.
(35) Morgan, H. L. The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service. *J. Chem. Doc.* **1965**, *5* (2), 107–113. https://doi.org/10.1021/c160017a018.
(36) Talevi, A.; Morales, J. F.; Hather, G.; Podichetty, J. T.; Kim, S.; Bloomingdale, P. C.; Kim, S.; Burton, J.; Brown, J. D.; Winterstein, A. G.; Schmidt, S.; White, J. K.; Conrado, D. J. Machine Learning in Drug Discovery and Development Part 1: A Primer. *CPT Pharmacomet. Syst. Pharmacol.* **2020**, *9* (3), 129–142. https://doi.org/10.1002/psp4.12491.
(37) Jones, G.; Willett, P.; Glen, R. C. Molecular Recognition of Receptor Sites Using a Genetic Algorithm with a Description of Desolvation. *J. Mol. Biol.* **1995**, *245* (1), 43–53. https://doi.org/10.1016/s0022-2836(95)80037-9.
(38) Liebeschuetz, J. W.; Cole, J. C.; Korb, O. Pose Prediction and Virtual Screening Performance of GOLD Scoring Functions in a Standardized Test. *J. Comput. Aided Mol. Des.* **2012**, *26* (6), 737–748. https://doi.org/10.1007/s10822-012-9551-4.
(39) Radaeva, M.; Ho, C.-H.; Xie, N.; Zhang, S.; Lee, J.; Liu, L.; Lallous, N.; Cherkasov, A.; Dong, X. Discovery of Novel Lin28 Inhibitors to Suppress Cancer Cell Stemness. *Cancers* **2022**, *14* (22), 5687. https://doi.org/10.3390/cancers14225687.
(40) Tang, M.; Wen, C.; Lin, J.; Chen, H.; Ran, T. Discovery of Novel A2AR Antagonists through Deep Learning-Based Virtual Screening. *Artif. Intell. Life Sci.* **2023**, *3*, 100058. https://doi.org/10.1016/j.ailsci.2023.100058.
(41) Jing, Y.; Bian, Y.; Hu, Z.; Wang, L.; Xie, X.-Q. S. Deep Learning for Drug Design: An Artificial Intelligence Paradigm for Drug Discovery in the Big Data Era. *AAPS J.* **2018**, *20* (3), 58. https://doi.org/10.1208/s12248-018-0210-0.
(42) Kim, J.; Park, S.; Min, D.; Kim, W. Comprehensive Survey of Recent Drug Discovery Using Deep Learning. *Int. J. Mol. Sci.* **2021**, *22* (18), 9983. https://doi.org/10.3390/ijms22189983.

(43) Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R. P.; de Freitas, N. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proc. IEEE* **2016**, *104* (1), 148–175. https://doi.org/10.1109/JPROC.2015.2494218.
(44) Torres, P. H. M.; Sodero, A. C. R.; Jofily, P.; Silva-Jr, F. P. Key Topics in Molecular Docking for Drug Design. *Int. J. Mol. Sci.* **2019**, *20* (18), 4574. https://doi.org/10.3390/ijms20184574.
(45) *Case: What is the difference between the GoldScore, ChemScore, ASP and ChemPLP scoring functions provided with GOLD? - The Cambridge Crystallographic Data Centre (CCDC)*. https://www.ccdc.cam.ac.uk/support-and-resources/support/case/?caseid=5d1a2fc0-c93a-49c3-a8e2-f95c472dcff0 (accessed 2023-09-02).
(46) *ZINC*. https://zinc20.docking.org/ (accessed 2023-08-23).
(47) Hawkins, P. C. D.; Skillman, A. G.; Warren, G. L.; Ellingson, B. A.; Stahl, M. T. Conformer Generation with OMEGA: Algorithm and Validation Using High Quality Structures from the Protein Databank and Cambridge Structural Database. *J. Chem. Inf. Model.* **2010**, *50* (4), 572–584. https://doi.org/10.1021/ci100031x.
(48) *pKa and Tautomer Enumeration Software | Quacpac*. https://www.eyesopen.com/quacpac (accessed 2023-08-23).
(49) Shelley, J. C.; Cholleti, A.; Frye, L. L.; Greenwood, J. R.; Timlin, M. R.; Uchimaya, M. Epik: A Software Program for PKaprediction and Protonation State Generation for Drug-like Molecules. *J. Comput. Aided Mol. Des.* **2007**, *21* (12), 681–691. https://doi.org/10.1007/s10822-007-9133-z.
(50) Lu, C.; Wu, C.; Ghoreishi, D.; Chen, W.; Wang, L.; Damm, W.; Ross, G. A.; Dahlgren, M. K.; Russell, E.; Von Bargen, C. D.; Abel, R.; Friesner, R. A.; Harder, E. D. OPLS4: Improving Force Field Accuracy on Challenging Regimes of Chemical Space. *J. Chem. Theory Comput.* **2021**, *17* (7), 4291–4300. https://doi.org/10.1021/acs.jctc.1c00302.
(51) *Hermes | CCDC*. https://www.ccdc.cam.ac.uk/solutions/software/hermes/ (accessed 2023-08-23).
(52) Kluyver, T.; Ragan-Kelley, B.; Pérez, F.; Granger, B.; Bussonnier, M.; Frederic, J.; Kelley, K.; Hamrick, J.; Grout, J.; Corlay, S.; Ivanov, P.; Avila, D.; Abdalla, S.; Willing, C.; Jupyter development team. Jupyter Notebooks – a Publishing Format for Reproducible Computational Workflows; Loizides, F., Scmidt, B., Eds.; IOS Press, 2016; pp 87–90. https://doi.org/10.3233/978-1-61499-649-1-87.
(53) Sapundzhi, F.; Dzimbova, T.; Pencheva, N.; Milanov, P. Comparative Evaluation of Four Scoring Functions with Three Models of Delta Opioid Receptor Using Molecular Docking. **2016**.
(54) Friesner, R. A.; Banks, J. L.; Murphy, R. B.; Halgren, T. A.; Klicic, J. J.; Mainz, D. T.; Repasky, M. P.; Knoll, E. H.; Shelley, M.; Perry, J. K.; Shaw, D. E.; Francis, P.; Shenkin, P. S. Glide: A New Approach for Rapid, Accurate Docking and Scoring. 1. Method and Assessment of Docking Accuracy. *J. Med. Chem.* **2004**, *47* (7), 1739–1749. https://doi.org/10.1021/jm0306430.
(55) McGann, M. FRED Pose Prediction and Virtual Screening Accuracy. *J. Chem. Inf. Model.* **2011**, *51* (3), 578–596. https://doi.org/10.1021/ci100436p.
(56) Pyscreener, 2023. https://github.com/coleygroup/pyscreener (accessed 2023-08-25).
(57) Alhossary, A.; Handoko, S. D.; Mu, Y.; Kwoh, C.-K. Fast, Accurate, and Reliable Molecular Docking with QuickVina 2. *Bioinformatics* **2015**, *31* (13), 2214–2216. https://doi.org/10.1093/bioinformatics/btv082.
(58) Ng, M. C. K.; Fong, S.; Siu, S. W. I. PSOVina: The Hybrid Particle Swarm Optimization Algorithm for Protein–Ligand Docking. *J. Bioinform. Comput. Biol.* **2015**, *13* (03), 1541007. https://doi.org/10.1142/S0219720015410073.
(59) Koes, D. R.; Baumgartner, M. P.; Camacho, C. J. Lessons Learned in Empirical Scoring with Smina from the CSAR 2011 Benchmarking Exercise. *J. Chem. Inf. Model.* **2013**, *53* (8), 1893–1904. https://doi.org/10.1021/ci300604z.

(60) Allen, W. J.; Balius, T. E.; Mukherjee, S.; Brozell, S. R.; Moustakas, D. T.; Lang, P. T.; Case, D. A.; Kuntz, I. D.; Rizzo, R. C. DOCK 6: Impact of New Features and Current Docking Performance. *J. Comput. Chem.* **2015**, *36* (15), 1132–1156. https://doi.org/10.1002/jcc.23905.
(61) Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; Chintala, S. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc., 2019; Vol. 32.
(62) Falcon, W.; The PyTorch Lightning team. PyTorch Lightning, 2019. https://doi.org/10.5281/zenodo.3828935.
(63) Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M.; Palmer, A.; Settels, V.; Jaakkola, T.; Jensen, K.; Barzilay, R. Analyzing Learned Molecular Representations for Property Prediction. *J. Chem. Inf. Model.* **2019**, *59* (8), 3370–3388. https://doi.org/10.1021/acs.jcim.9b00237.
(64) Nienaber, V. L.; Breddam, K.; Birktoft, J. J. A Glutamic Acid Specific Serine Protease Utilizes a Novel Histidine Triad in Substrate Binding. *Biochemistry* **1993**, *32* (43), 11469–11475. https://doi.org/10.1021/bi00094a001.
(65) Nepali, K.; Lee, H.-Y.; Liou, J.-P. Nitro-Group-Containing Drugs. *J. Med. Chem.* **2019**, *62* (6), 2851–2893. https://doi.org/10.1021/acs.jmedchem.8b00147.
(66) Carta, G.; Knox, A. J. S.; Lloyd, D. G. Unbiasing Scoring Functions: A New Normalization and Rescoring Strategy. *J. Chem. Inf. Model.* **2007**, *47* (4), 1564–1571. https://doi.org/10.1021/ci600471m.