UNIVERSITÀ
DI PAVIA

UNIVERSITÀ DEGLI STUDI DI PAVIA

Dipartimento di Chimica

Direttore Prof. Mauro Freccero

Corso di Laurea Magistrale in Chimica

Generazione di ligandi specifici per TRAP1 mediante un modello di Deep Learning basato su un'architettura RNN-LSTM

Deep Generative Modeling of TRAP1-Specific Ligands
using an RNN-LSTM architecture

Relatore:
Prof. Giorgio Colombo

Laureanda:
Holly Mae Thurston

Correlatore:
Dott.ssa Elisabetta Moroni

Correlatore:
PhD Cristiano Sciva

Anno Accademico 2025/2026

Contents

1 Introduction **14**

1.1 TRAP1 chaperone 14

1.1.1 Structure and HSP90 Family 14

1.1.2 Role of TRAP1 in cancer cells 16

1.1.3 Targeting TRAP1 in anticancer therapy 17

1.2 Deep Learning and Drug Design 19

1.2.1 Introduction to Machine Learning 19

1.2.2 Evolution of Machine Learning Models 21

1.2.3 Drug Design and Machine Learning 25

1.2.4 Deep Learning and Molecular Generation 27

2 Materials and Methods **33**

2.1 CLMs and RNN-LSTM networks 33

2.2 Molecular Docking 47

2.3 Molecular Dynamics 49

3 Results and Discussion **53**

3.1 LSTM: training, finetuning and generation 53

3.2 Postprocessing and receptor ensemble preparation 61

3.2.1 Filtering of generated molecules 61

3.2.2 Molecular dynamics simulations of ATP-bound TRAP1 . . . 63

3.3 Ensemble docking-based prioritization of LSTM-generated TRAP1 Ligands 66

3.3.1 Generation, filtering and quality assessment of the decoy set 66

3.3.2 Docking on MD-derived TRAP1 conformations 69

3.3.3 Enrichment-based selection of the final receptor ensemble . . 70

3.3.4 Prioritization of LSTM-generated molecules 72

3.3.5 Chemical diversity and ADMET analysis of prioritizedgenerated candidates 74

3.3.6 Conclusions and perspectives 82

Riassunto

Le proteine sono biomolecole fondamentali per la vita cellulare: le loro funzioni sono molto diverse, dalla catalisi di reazioni metaboliche al trasporto di molecole, dalla formazione di strutture supramolecolari alla trasmissione di informazione tra cellule. Per poter svolgere correttamente queste funzioni, le proteine devono ripiegarsi in una struttura tridimensionale ordinata, tipicamente associata ad una certa attività biologica. Questo processo di ripiegamento, chiamato folding, è assistito da una serie di proteine specializzate note come chaperoni molecolari, che aiutano altre proteine a ripiegarsi correttamente e a mantenere una conformazione funzionale nell'ambiente estremamente affollato dell'interno della cellula.

Tra le principali famiglie di chaperoni vi è quella della Heat Shock Protein 90 (HSP90), che nell'uomo comprende quattro isoforme localizzate in diversi compartimenti cellulari ed associate al folding di specifiche classi di proteine substrato, dette clients. Queste isoforme sono HSP90α e HSP90β nel citosol, Glucose-Regulated Protein 94 (GRP94) nel reticolo endoplasmatico e TRAP1 (Tumor Necrosis Factor Receptor Associated Protein 1) nei mitocondri.

L'obiettivo di questa tesi è stato quello di sviluppare un protocollo computazionale che combina Deep Learning e physics-based modeling per la generazione e prioritizzazione di nuovi candidati ligandi selettivi per l'isoforma mitocondriale TRAP1.

Le funzioni di TRAP1 vanno oltre il controllo del folding delle proteine. Nei mitocondri infatti, TRAP1 partecipa alla regolazione di processi metabolici e bioenergetici, modulando l'attività di componenti della catena respiratoria e degli enzimi metabolici, contribuendo alla regolazione della fosforilazione ossidativa, della produzione di specie reattive dell'ossigeno (ROS) e delle vie di sopravvivenza cellulare. La disregolazione di TRAP1 è stata associata allo sviluppo di diverse patologie tumorali, rendendola quindi un target promettente per lo sviluppo di strategie antitumorali.

Lo sviluppo di inibitori selettivi di TRAP1 rappresenta tuttavia una sfida rilevante. Gli approcci più comuni si sono concentrati sul blocco dell'attività ATPasica della proteina, associata al sito di legame dell'ATP localizzato nel dominio

N-terminale. Tuttavia il sito di legame per ATP è altamente conservato tra i diversi membri della famiglia HSP90, di conseguenza molecole progettate per legare questo sito possono inibire anche altre isoforme, con possibili effetti aspecifici e conseguente tossicità. La principale sfida nella progettazione di inibitori specifici per TRAP1 consiste quindi nell'identificare nuove classi di composti, o scaffold chimici alternativi, capaci di sfruttare differenze strutturali e conformazionali specifiche di TRAP1, pur agendo su un sito di legame altamente conservato.

Nell'ultimo decennio, lo sviluppo di metodi di Artificial Intelligence (AI), e in particolare il Deep Learning (DL), ha avuto un impatto significativo nella progettazione di farmaci. Il DL è una branca del Machine Learning basata su reti neurali organizzate in più livelli, o strati, di elaborazione. Ogni strato trasforma l'informazione ricevuta dal precedente, permettendo al modello di apprendere relazioni complesse e non lineari nei dati. Una delle applicazioni più promettenti e in rapida evoluzione del DL è il de novo drug design, in cui reti neurali generative vengono utilizzate per creare nuovi composti con proprietà farmacologiche e fisico-chimiche desiderate.

In questo progetto è stato sviluppato un Chemical Language Model (CLM), un modello generativo basato sul DL, con l'obiettivo di generare nuovi ligandi specifici per TRAP1, successivamente selezionati mediante un approccio structure-based basato sulla valutazione della loro interazione predetta con il sito di legame della proteina.

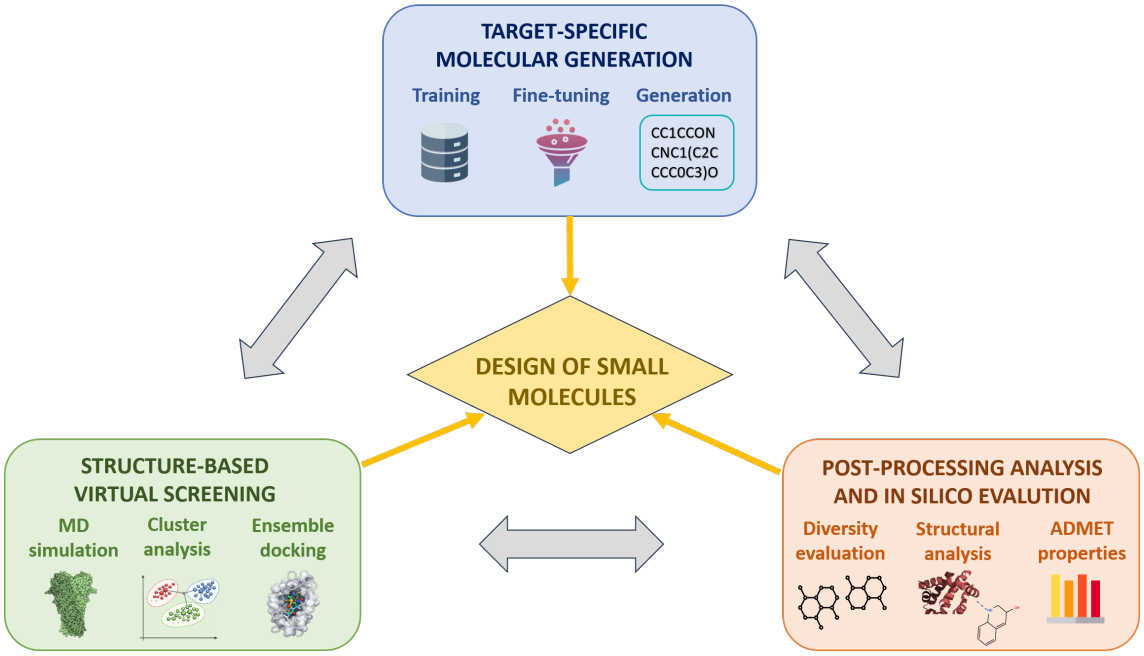


Figure 1: Strategia computazionale implementata nel progetto di tesi

Nel modello generativo CLM, le molecole sono rappresentate come SMILES, cioè stringhe testuali lineari che codificano la struttura chimica in una sequenza di caratteri, e generate tramite una Recurrent Neural Network con unità Long Short-Term Memory (RNN-LSTM). Le RNN apprendono a generare una sequenza un token alla volta, condizionando ogni nuova predizione sui token precedenti, risultando così efficaci strumenti per la generazione di molecole rappresentate in forma sequenziale; l'architettura LSTM permette inoltre di superare alcune delle limitazioni delle RNN standard (vanishing/exploding gradient), come la difficoltà nel mantenere dipendenze a lungo raggio nella sequenza SMILES. Una volta addestrata, la rete viene usata in modalità generativa, campionando istanze in sequenza per produrre nuove stringhe SMILES.

La generazione può essere orientata verso composti con proprietà desiderate tramite fine-tuning del modello su un sottoinsieme più ristretto di molecole con una specifica attività biologica. In questa tesi, l'addestramento si è articolato in due fasi: pre-training (apprendimento generale sullo spazio chimico) e fine-tuning (specializzazione sul target di interesse).

Nella prima fase di pre-training, è stato sviluppato un modello generico che apprende la sintassi chimica e la semantica degli SMILES di molecole drug-like, cioè ligandi aventi caratteristiche simili a quelle dei farmaci, utilizzando circa 1,5 milioni di composti presenti nel database ChEMBL. Nella seconda fase di fine-tuning, il modello pre-addestrato è stato ottimizzato, tramite transfer learning, su un insieme più ristretto di attivi noti e selettivi per TRAP1. Questo approccio permette di orientare la generazione di molecole con caratteristiche simili a quelle dei farmaci da uno spazio chimico generale verso regioni più rilevanti per il target di interesse. Questo spostamento dalla generazione generale alla generazione target specifica rappresenta il vantaggio chiave del transfer learning nella progettazione di molecole, consentendo un'esplorazione efficiente di regioni chimiche di composti biologicamente rilevanti anche quando è disponibile solo un numero limitato di composti attivi noti.

Il modello CLM fine-tuned è stato quindi utilizzato per generare 100.000 composti. Le molecole generate sono state filtrate sulla base di tre metriche: validity (correttezza chimico-strutturale, es. connettività), uniqueness (assenza di duplicati tra i generati) e novelty (distanza chimica da composti usati per l'addestramento). Dopo questi filtri, è stato quindi ottenuto un insieme di molecole valide, uniche e nuove rispetto ai dati di partenza. Prima della valutazione structure-based e drug-likeness, è stato applicato un ulteriore filtro basato sulla log-likelihood, ovvero il valore con cui la rete LSTM quantifica quanto una sequenza generata sia probabile secondo la distribuzione appresa. Sono stati selezionati i candidati con maggiore log-

likelihood, cioè le molecole più coerenti con la distribuzione del modello fine-tuned, ottenendo un set finale di 9.482 SMILES da sottoporre alle successive analisi.

La log-likelihood del modello generativo, tuttavia, non misura l'affinità di legame, ma solo la probabilità che la molecola generata rispetti le regole della distribuzione appresa. Per questo motivo è stato necessario introdurre uno step indipendente basato sulla struttura del target, per verificare la compatibilità delle molecole generate con il sito ATP di TRAP1, in termini di posa e interazioni. Per ottenere una stima preliminare della compatiblità di legame di queste molecole per il recettore è stato eseguito un ensemble molecular docking sulle conformazioni più rappresentative di TRAP1, estratte da traiettorie di dinamica molecolare (MD). L'uso di più conformazioni del recettore derivate da MD permette di tenere conto della natura dinamica della proteina e della flessibilità del sito di legame.

Tramite un clustering focalizzato a distinguere le posizioni delle catene laterali dei residui entro 5 Å dall'ATP, sono state ottenute dodici conformazioni rappresentative distinte del sito di legame. Poiché non tutte avrebbero potuto essere ugualmente informative per il riconoscimento del ligando (alcune potrebbero introdurre rumore nel docking), è stata condotta una fase di selezione di queste conformazioni tramite virtual screening retrospettivo, usando degli attivi noti di TRAP1 e i relativi decoy, cioè molecole con proprietà fisico-chimiche simili agli attivi ma topologicamente distinte e presumibilmente non leganti. L'obiettivo è stato di identificare le conformazioni, singole o in combinazione, più capaci di discriminare attivi da decoy, collocando preferenzialmente i primi nelle posizioni più alte della classifica di docking; l'ensemble finale di recettori è stato quindi selezionato sulla base della capacità di arricchimento dei ligandi attivi noti. I decoy sono stati generati tramite DeepCoy, un metodo basato su Graph Neural Networks (GNN) che codifica il ligando attivo in una rappresentazione latente catturandone le caratteristiche essenziali (dimensioni, gruppi funzionali, connettività). Per ciascun attivo sono stati generati 1.000 decoy, da cui sono stati selezionati i 40 migliori in base alla distanza di Mahalanobis. Questa metrica è stata scelta per misurare la somiglianza delle proprietà chimico-fisiche tra ciascun attivo e i corrispondenti decoy, in quanto considera sia la varianza dei singoli descrittori sia le correlazioni tra essi: un valore basso indica un decoy molto simile al corrispondente attivo. La qualità del set finale è stata verificata tramite analisi degli scaffold di Bemis–Murcko e un classificatore Random Forest, confermando diversità strutturale rispetto agli attivi e l'assenza di evidenti bias di classificazione, ovvero differenze sistematiche tra attivi e decoy che potrebbero influenzare artificialmente le prestazioni del virtual screening.

Il set di benchmark attivi/decoy è stato quindi sottoposto a docking nelle dod-

ici conformazioni di TRAP1, valutando le prestazioni tramite "Enrichment Factor" (EF), una metrica utilizzata nel virtual screening per misurare la capacità di posizionare i composti attivi noti nelle prime posizioni di una lista ordinata rispetto a quanto ci si aspetterebbe da una selezione casuale. In questo contesto, un valore elevato di EF indica che il protocollo assegna agli attivi posizioni migliori rispetto ai decoy nella classifica di docking. Poiché i docking score ottenuti tra conformazioni recettoriali diverse non sono direttamente confrontabili in termini assoluti, le prestazioni sono state valutate in base al ranking relativo all'interno di ciascuna conformazione, calcolando l'EF1%, che misura l'arricchimento degli attivi nel primo 1% della classifica, per tutte le possibili combinazioni recettoriali. Il valore massimo di EF1% è stato ottenuto con una combinazione di sole due conformazioni, tuttavia questo ensemble è troppo poco rappresentativo della variabilità conformazionale del sito di legame, come emerso dalla MD. Per questo motivo, la selezione finale è stata effettuata tra le combinazioni con EF1% immediatamente inferiore al valore massimo, ma caratterizzate da un numero maggiore di conformazioni recettoriali. Tra le combinazioni con prestazioni comparabili è stato scelto un ensemble di cinque conformazioni, privilegiando le combinazioni fatte da strutture che ricorrevano più frequentemente negli ensemble ad alto arricchimento. Questo approccio ha permesso di mantenere prestazioni di virtual screening elevate, bilanciandole con un'adeguata rappresentazione della flessibilità del sito di legame. L'ensemble finale ha raggiunto un EF1% = 18,93, valore che supporta l'utilizzo di questo ensemble per il docking e la prioritizzazione delle molecole generate dal modello LSTM. Le molecole generate selezionate sono state sottoposte a docking usando lo stesso protocollo applicato al set di benchmark attivi/decoy. A partire dalla distribuzione dei docking score dei decoy sono state definite le soglie FPR1% e FPR5%, corrispondenti rispettivamente a 9,26 e 7,29 kcal/mol. La False Positive Rate (FPR) indica la frazione di decoy che supera una determinata soglia di score, venendo quindi classificata come active-like: una soglia FPR1% corrisponde a un valore di docking score più favorevole, cioè più negativo, raggiunto solo dall'1% dei decoy, mentre una soglia FPR5% è raggoiunta dal 5% dei decoy. Le stesse soglie sono state poi valutate sugli attivi noti, calcolando il True Positive Rate (TPR), cioè la frazione di attivi recuperata a ciascuna soglia. Il TPR è risultato pari al 22,2% alla soglia FPR1% e al 31,1% alla soglia FPR5%, indicando che una quota significativa di attivi noti ricade nelle regioni di docking score più favorevoli definite dalla distribuzione dei decoy. Applicando queste soglie alle molecole generate, 549 composti (5,99%) hanno raggiunto docking score migliori della soglia FPR1%, mentre 1.074 (11,72%) hanno superato anche la soglia FPR5%. Questi composti costituiscono quindi due sot-

toinsiemi di molecole generate con docking score particolarmente favorevoli, poiché superano soglie che, nel benchmark, erano raggiunte solo da una piccola frazione dei decoy e consentivano di recuperare una parte degli attivi noti. Il sottoinsieme di 549 molecole (soglia FPR1%) è stato quindi analizzato in termini di similarità strutturale rispetto ai ligandi attivi noti di TRAP1, usando la similarità di Tanimoto. Per ciascuna molecola generata sono state considerate sia la similarità di Tanimoto massima rispetto all'attivo più vicino, sia la similarità media rispetto all'intero set di attivi. Rispetto al dataset generato completo, entrambe le metriche risultano aumentate, indicando che il sottoinsieme selezionato è mediamente più vicino allo spazio chimico degli attivi noti, pur mantenendo una certa diversità strutturale; la maggior parte delle molecole selezionate mostra una similarità intermedia o alta con almeno un ligando attivo noto. Tuttavia, non è stata osservata una correlazione tra similarità strutturale e ranking di docking, suggerendo che la prioritizzazione non dipende esclusivamente dalla similarità coi ligandi noti, ma anche dalla compatibilità prevista delle molecole con il sito di legame, in termini di posa e interazioni ligando-recettore.

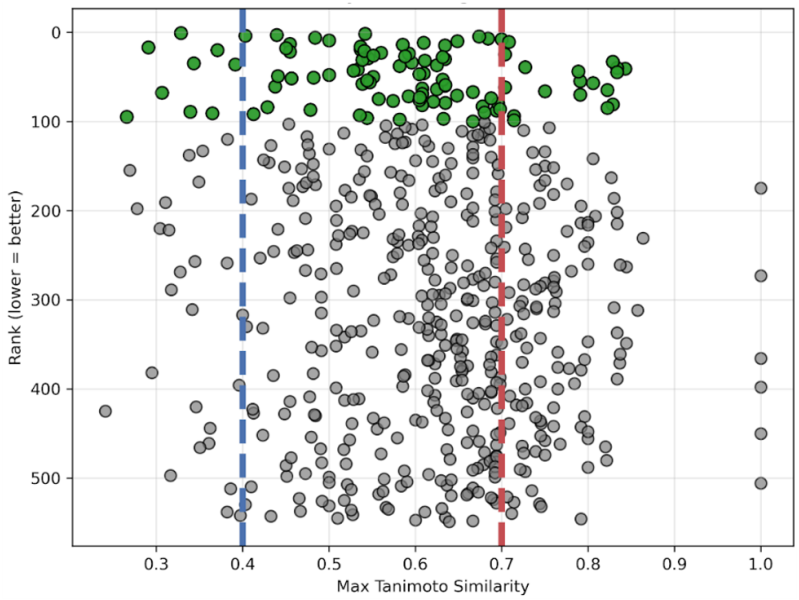


Figure 2: Massima similarità di Tanimoto in funzione del rank del docking per le molecole generate selezionate tramite FPR1%. I punti verdi indicano i composti con il rank più elevato (docking score < -10.2076 kcal/mol); i punti grigi indicano le restanti molecole selezionate

Infine, le molecole selezionate sono state analizzate anche in termini di proprietà fisico-chimiche e ADME preliminari, al fine di ottenere una prima valutazione della loro drug-likeness e sviluppabilità. Nella Figura 3.10 sono riassunti i risultati delle principali proprietà calcolate:

Property	Mean	Range	Recommended range	% of molecules within recommended range
Molecular Weight (MW)	359.02	245.28 – 576.69	≤ 500	96.5
Octanol-water partition coefficients (logP)	2.13	-2.10 – 4.81	-2 – 6.5	99.8
Topological Polar Surface Area (TPSA), Å²	110.61	61.04 – 236.21	≤ 140	88.5
Hydrogen-bonding donors (HBD)	3	0 – 8	≤ 5	99.5
Hydrogen-bonding acceptors (HBA)	6	4 – 14	≤ 10	95.4
Rotable bonds	4	1 – 22	≤ 10	99.1
Water solubility (QplogS)	-4.21	-7.61 – -1.03	-6.5 – 0.5	97.3
Predicted apparent permeability across Caco-2 cells (QPPCaco), nm/s	170.43	0.01 – 1945.81	≥ 500	2.2

Figure 3: Principali proprietà fisico-chimiche calcolate via Qikprop

Nel complesso, le molecole generate presentano un profilo chimicofisico favorevole con peso molecolare, lipofilicità (logP), capacità di formare legami a idrogeno e solubilità che, nella maggior parte dei casi, rientrano negli intervalli raccomandati. Le molecole mostrano inoltre una polarità moderata, mentre la permeabilità cellulare Caco-2 risulta mediamente inferiore rispetto ai valori ottimali. Questo aspetto potrebbe rappresentare un limite per l'assorbimento e fornisce un'informazione importante per la successiva ottimizzazione dei candidati da sintetizzare. L'analisi della carica molecolare ha evidenziato una netta prevalenza di composti con carica negativa (509 molecole negative, 37 neutre e 3 positive). Tale caratteristica può favorire le interazioni elettrostatiche con il sito ATP di TRAP1 e migliorare la solubilità acquosa, ma potrebbe penalizzare la permeabilità di membrana e l'accumulo mitocondriale. Anche questa caratteristica dovrà essere valutata nella fase successiva di sintesi dei candidati più promettenti.
La valutazione della drug-likeness mediante le regole di Lipinski ha mostrato che oltre il 94% delle molecole non presenta alcuna violazione, indicando che la libreria generata e prioritizzata dal docking è costituita prevalentemente da composti con caratteristiche compatibili con lo spazio chimico dei farmaci orali. Il confronto con gli inibitori noti di TRAP1 ha inoltre evidenziato un'elevata sovrapposizione dei principali descrittori, confermando che le molecole generate occupano uno spazio chimico coerente con quello dei composti bioattivi noti. Per verificare la plausibilità delle pose di docking, è stata effettuata un'ispezione visiva dei complessi proteina-ligando per alcuni composti rappresentativi della libreria selezionata.

Tra questi, il composto gen5634, appartenente alla classe di similarità massima intermedia, ha mostrato in tutte le conformazioni di TRAP1 un orientamento nel sito di legame altamente comparabile a quello dell'ATP. La posa di legame di gen5634 nel sito ATP di TRAP1 è rappresentata in Fig 4:

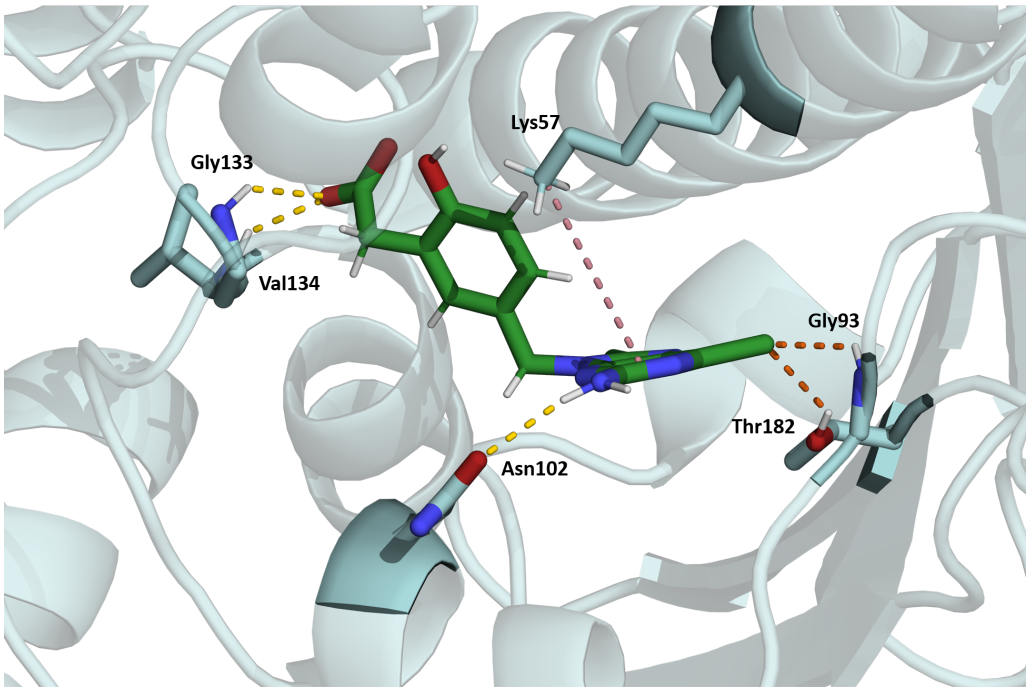


Figure 4: Posa di docking di gen5634 nella conformazione del recettore TRAP1 (sito1_rep2_eps1_15_c0). I legami a idrogeno sono indicati in giallo, le interazioni π-catione in rosa e i legami alogeno in arancione.

Il composto conserva infatti le principali interazioni con residui coinvolti nel riconoscimento del nucleotide, tra cui Asp89, Asn102, Ser109, Ser111, Gly133 e Val134, oltre a instaurare ulteriori contatti che potrebbero contribuire ulteriormente alla stabilizzazione del complesso. Di particolare rilievo è l'interazione con il residuo Asn102, un residuo proposto come determinante per la selettività verso TRAP1 rispetto alle altre isoforme HSP90. Sono stati inoltre analizzati altri composti con diversi livelli di similarità rispetto agli inibitori noti di TRAP1. Anche molecole meno simili, come ad esempio gen4947, occupano il sito ATP con orientazioni compatibili e mantengono interazioni con residui chiave, incluso Asn102.
Nel complesso, questa valutazione strutturale suggerisce che il workflow sia stato in grado di generare e selezionare molecole strutturalmente diversificate ma che possiedono le principali caratteristiche chimico-fisiche utili al riconoscimento del sito ATP di TRAP1. La successiva selezione mediante docking ha inoltre privilegiato composti che mantengono le interazioni essenziali con il sito ATP pur introducendo nuovi pattern di interazione.

Conclusioni e prospettive future

I risultati mostrano che il workflow proposto integra efficacemente un modello generativo target-specifico (CLM) e metodi structure-based (ensemble docking) per la progettazione de novo e la prioritizzazione di nuovi candidati ligandi di TRAP1. Il workflow sviluppato ha permesso di selezionare un sottoinsieme di molecole generate con diversità strutturale rispetto ai ligandi attivi noti e con caratteristiche promettenti in termini di compatibilità prevista con il sito ATP, comportamento di docking rispetto al benchmark attivi/decoy, e proprietà fisico-chimiche preliminari.

Il passo successivo necessario sarà la validazione sperimentale dei composti selezionati, tramite sintesi chimica e saggi biochimici e cellulari, per confermarne il legame a TRAP1, caratterizzarne l'attività inibitoria, la selettività rispetto alle altre isoforme e le proprietà cellulari rilevanti, come permeabilità e accumulo mitocondriale. Una direzione di sviluppo particolarmente promettente è l'integrazione del protocollo in una strategia di active learning. In questo scenario, il benchmark attivi/decoy sviluppato in questa tesi potrebbe costituire un primo set di riferimento per addestrare modelli predittivi come Graph Neural Network, basati su informazioni molecolari e descrittori structure-based derivati dal docking. I composti prioritizzati dovrebbero poi essere testati sperimentalmente, e i risultati ottenuti, distinguendo molecole confermate come attive e inattive, potrebbero essere reinseriti nel modello per migliorarne progressivamente la capacità di generare e selezionare candidati TRAP1 più promettenti. Infine, il protocollo sviluppato è potenzialmente generalizzabile ad altri target farmacologici, purché siano disponibili dati sui ligandi attivi e informazioni strutturali adeguate per la modellistica e il docking. In questa prospettiva, il lavoro rappresenta un esempio di integrazione tra intelligenza artificiale generativa, dinamica molecolare e virtual screening structure-based per la progettazione computazionale de novo di molecole con potenziale terapeutico.

Abstract

TRAP1 (Tumor Necrosis Factor Receptor Associated Protein 1) is the mitochondrial paralog of the HSP90 family of molecular chaperones, which in humans also includes Hsp90α, Hsp90β, Grp94, with each localized in a distinct subcellular compartment. TRAP1 is involved in the regulation of mitochondrial metabolism, oxidative phosphorylation, reactive oxygen species production and cell survival pathways. Its dysregulation has been associated with several cancer contexts, making TRAP1 a promising target for anticancer drug discovery. One of the main approaches for developing TRAP1 inhibitors is to target the N-terminal ATP-binding domain and interfere with ATPase activity. However, the ATP-binding site is highly conserved among Hsp90 family members, making the design of TRAP1-selective orthosteric inhibitors challenging. In addition, effective TRAP1-targeted compounds must be able to reach the mitochondrial compartment. For this reason, the identification of new chemical scaffolds capable of exploiting TRAP1-specific structural and conformational features remains an important objective. In the past decade, deep learning (DL) has increasingly impacted drug discovery, particularly in the field of de novo molecular design, where generative models are used to propose novel compounds with desired properties. In this project a target specific Chemical Language Model (CLM) based on DL was developed to generate candidate TRAP1 specific ligands. The model uses SMILES strings as input data and is based on a recurrent neural network with Long Short-Term Memory units (RNN-LSTM), an architecture well suited for generating molecules represented in sequential form. In this thesis, training consisted of two main steps: pretraining and finetuning. First, a general model was trained on approximately 1.5 million drug-like compounds from the ChEMBL database, allowing it to learn the syntax and general chemical features of drug-like molecular space. This model was then finetuned, through transfer learning, on a small set of known TRAP1 active compounds. As a result, the model shifted its learned molecular distribution from a broad drug-like chemical space toward molecular features associated with known TRAP1 ligands, enabling target-oriented generation even when only a limited number of known actives is available.

Subsequently, the finetuned CLM was used to generate 100,000 SMILES. The generated molecules were filtered for validity, uniqueness, and novelty, and a subset of high-confidence candidates was selected based on the likelihood distribution of the model. Since likelihood reflects how consistent a generated molecule is with the learned model distribution, but does not provide information on binding to TRAP1, a structure-based evaluation step was introduced to prioritize the most promising candidate binders. The selected generated molecules were then evaluated by ensemble molecular docking against the most representative TRAP1 conformations extracted from molecular dynamics (MD) simulations. This structure-based strategy was used to assess the predicted compatibility of the molecules with the TRAP1 ATP-binding site, while accounting for protein flexibility. In order to achieve this, firstly, a cluster analysis was carried out to identify the most populated TRAP1 binding-site conformations, and secondly, such receptors were tested in a retrospective virtual screening experiment using known TRAP1 active ligands and corresponding decoys. Enrichment factor analysis was performed on the active/decoys docking results to select the ensemble of receptor conformations that most effectively enriched known TRAP1 actives over decoys in the docking ranking. The combination of receptors that yielded the best enrichment factor (EF1%=18.93) was therefore used for docking as a structure-based prioritization tool for the selected subset of LSTM-generated molecules. False Positive Rate thresholds were derived from the decoy score distribution and applied to the generated molecules. The stringent FPR1% threshold identified 549 compounds, which were further analyzed. Tanimoto similarity analysis showed that this selected subset was closer to the chemical space of known TRAP1 ligands than the full LSTM-generated library, while still maintaining an intermediate level of structural diversity. Moreover, no clear relationship between molecular similarity and docking rank was observed, suggesting that the docking-based prioritization was not driven solely by ligand similarity, but also by predicted compatibility with the ATP-binding site.

Preliminary physicochemical and ADME-related properties were calculated to provide an initial assessment of the selected compounds as potential drug candidates. The selected generated molecules showed an overall physicochemical profile broadly consistent with that of known TRAP1 inhibitors, particularly in terms of molecular weight, lipophilicity and solubility-related descriptors. However, predicted cell permeability was generally lower for the generated compounds, consistent with the slightly higher polarity of part of the generated subset. This result provides useful information for the subsequent optimization of the selected candidates, highlighting the need to balance binding-site compatibility with physicochemical properties

relevant to cellular and mitochondrial access. Finally, visual inspection of representative docking poses was performed to qualitatively assess the plausibility of the predicted binding modes. Selected generated molecules occupied the TRAP1 ATP-binding pocket with orientations comparable to ATP and established interactions with residues involved in nucleotide recognition, while also introducing additional contacts and structurally diverse chemotypes. These observations support the complementarity between the generative model, which produced molecules bearing chemical features compatible with ATP-site recognition, and the docking-based prioritization, which selected compounds with predicted binding modes compatible with the TRAP1 ATP-binding site. Overall, this work provides an integrated and versatile computational strategy that combines target-specific molecular generation with structure-based, dynamics-informed prioritization through MD-derived ensemble docking and active/decoy benchmarking, enabling the identification of novel TRAP1-oriented candidate ligands for future experimental validation and optimization. More broadly, the workflow could be extended to other therapeutic targets for which ligand data and structural information are available, supporting its potential applicability as a general framework for target-specific de novo molecular design.

Chapter 1

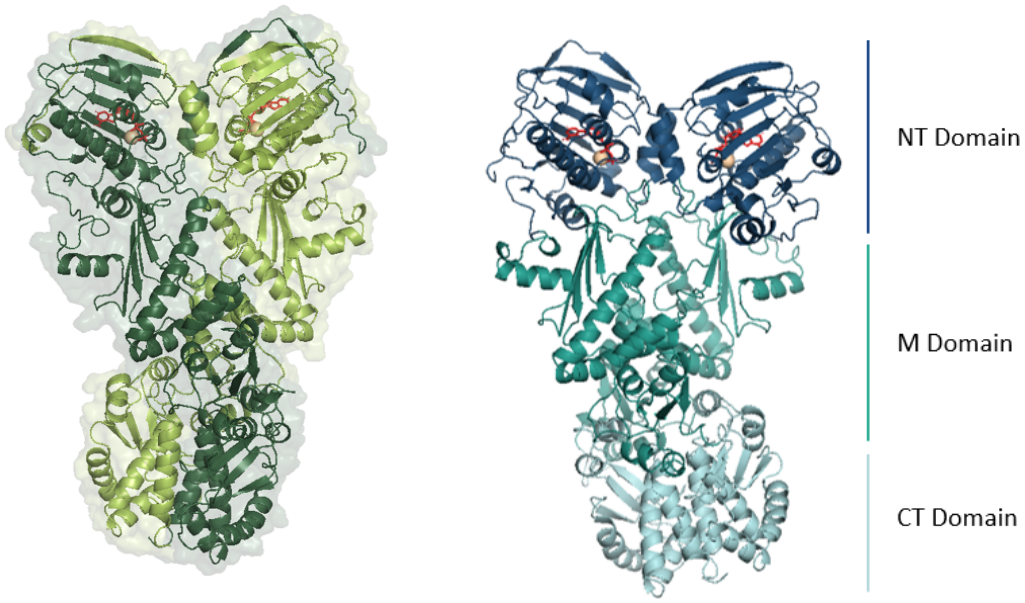
Introduction

1.1 TRAP1 chaperone

1.1.1 Structure and HSP90 Family

TRAP1 (Tumor Necrosis Factor Receptor Associated Protein 1) is a molecular chaperone belonging to the family of heat shock proteins 90, known as HSP90. The human HSP90 family comprises four proteins: Hsp90α (inducible) and Hsp90β (constitutive form) in the cytoplasm, glucose-regulated protein (Grp94) in the endoplasmic reticulum (ER), and TRAP1 in the mitochondrial matrix. HSP90 family members are ATP-dependent molecular chaperones that play essential roles in late-stage protein quality control by assisting the folding, stabilization, maturation, and assembly of a wide range of so-called client proteins. Their activity is often regulated by co-chaperones, proteins that assist the chaperone cycle by modulating ATPase activity, facilitating client protein recruitment and stabilizing specific conformational states required for client maturation. Whilst this is true for cytosolic Hsp90 isoforms, no co-chaperones have been conclusively identified for TRAP1, which suggests that TRAP1 may function independently compared to the characteristic co-chaperone network of other HSP90 family members. Whilst TRAP1 is capable of stabilizing and regulating mitochondrial client proteins, accumulating evidence suggests that its primary functions extend beyond general protein folding. In mitochondria, TRAP1 modulates the activity of respiratory chain components and metabolic enzymes, contributing to the regulation of oxidative phosphorylation, reactive oxygen species (ROS) production, and cell survival pathways.

All Hsp90 isoforms share the same overall homodimer structure.



(a) TRAP1 dimer structure (b) TRAP1 domain structure

Figure 1.1: olo-form TRAP1

Each protomer consists of three main domains: the N-Terminal Domain (NTD), containing the ATP binding site, the Middle Domain (MD), which contributes to ATP hydrolysis and client interaction, and the C-Terminal Domain (CTD) for protomer dimerization. However, unlike cytosolic Hsp90 isoforms, TRAP1 lacks both the charged flexible region (CR) between NTD and MD, and the MEEVD sequence essential for the interaction with several co-chaperones. Also it has an unusually long N-terminal β strand, the so called NTD strap, which crosses between protomers in the closed dimeric state. This stabilizes the closed conformation and regulates ATP hydrolysis and ATPase activity, and also acts as a thermal regulator of protein function, inhibiting it at low temperatures. One of the most distinctive structural features of TRAP1 is the asymmetric configuration of the two protomers in the closed-form with one protomer in a buckled state at the MD-CTD interface whilst the other one in a straight state. In its apo-form, TRAP1, similar to Hsp90, is a dimeric structure with protomers interacting only between CTDs, this is the so-called open conformation. Unlike Hsp90, upon ATP binding (one molecule for each NTD), both protomers undergo structural modifications leading to the formation of a strained closed asymmetric conformation with one protomer "straight" and the other "buckled". Such asymmetry generates a different ATP hydrolysis rate for each monomer. The strained buckled protomer hydrolyzes the bound ATP and the MD-CTD interface rearranges resulting in the ADP protomer assuming a straight

state whilst the ATP protomer a buckled one. This buckled protomer can therefore undergo ATP hydrolysis generating the apo-open form. The first hydrolysis leads to a rearrangement of the client-binding site (in the MD) and thus conformational changes of the client protein. The second hydrolysis, instead, releases the client, as well as ADP, thus resetting TRAP1 for the next client.

1.1.2 Role of TRAP1 in cancer cells

Hsp90 family members are overexpressed in many cancer cell lines thereby supporting tumor progressing pathways. However, the function and regulation of these chaperones vary significantly due to their distinct subcellular localizations, which influence their roles and mechanisms of action. Several studies have shown that TRAP1 can contribute to mitochondrial pathways rewiring in cancer cells, although its effects are strongly dependent on tumor type and metabolic context. Many cancer cells depend on modifications in mitochondrial functions concerning energy production, redox modulation, anabolic pathways and apoptosis pathway regulation, all of which are linked to TRAP1. One protein that has been shown to interact with TRAP1 is SDH or succinate dehydrogenase, however, the functional outcome of this interaction appears to be highly context-dependent. For example, in cancer cells from human colon, bone, ovary, and cervix, TRAP1 inhibits SDH[1], which is part of both the electron transport chain and the Krebs cycle, thus suppressing OXPHOS (Mitochondrial oxidative phosphorylation), the primary metabolic process for ATP production. Consequently, metabolism is shifted towards aerobic glycolysis; this increased glucose uptake and fermentation of glucose to lactate, observed in cancer cells, is also known as the Warburg effect[2]. OXPHOS suppression is also assisted by the inhibition of cytochrome c oxidase[3], the complex IV of the respiratory chain. The subsequent accumulated succinate leads to gene activation that enhances tumor invasion and angiogenesis (development of new blood vessels). On the other hand, in cancer cells from human brain, prostate and breast, TRAP1 stabilizes and activates SDH, therefore, sustaining mitochondrial respiration and providing stress resistance[4].

Another aspect to consider is that, under stress conditions correlated with neoplastic growth, such as redox disequilibria or nutrient shortage, TRAP1, help preserve mitochondrial proteostasis and bioenergetic function. Through this activity, TRAP1 can suppress the permeability of transition pore (PTP) , a process that can lead to cell death[5]. When mitochondrial TRAP1 activity is inhibited, however, cancer cells activate autophagy as a compensatory pro-survival response to mitochondrial bioenergetic stress.

Furthermore, by inhibiting mitochondrial respiration, ROS or reactive oxygen species (which are a natural byproduct of this process) levels go down so cells experience less oxidative damage. Such decline, however, can have different effects based on tumor context, thus type of cancer, stage, environment, ect... For example, in early tumors, cells are fragile so a ROS reduction could promote survival[6], instead, in advanced tumors, ROS helps create mutations and therefore a lack of these could in fact slow down cancer progression. For instance, with clear cell renal cell carcinoma[7] and high-grade ovarian cancer[7], lower levels of TRAP1, therefore higher ROS levels, lead to more aggressive tumors.
TRAP1 also plays a central role in cell cycle modulation via the ubiquitination of client proteins, such as CDK1 (Cyclin-Dependent Kinase 1), a protein that is required for cells to enter mitosis. At low or normal levels of TRAP1, ubiquitination of CDK1 takes place, therefore this protein is tagged for degradation and cell division is suppressed. Instead, at elevated levels of TRAP1, ubiquitination is prevented and cells enter mitosis promoting uncontrolled proliferation. This is especially true for breast, lung, and colorectal cancers. For these, increased expression of TRAP1 has been suggested to contribute to cancer resistance to chemotherapies that target cell cycle regulation[8]. As a result, TRAP1 could also be used as a potential diagnostic/prognostic marker for tumors with deregulated cell cycle control.

1.1.3 Targeting TRAP1 in anticancer therapy

Taking into account the above observations, it is reasonable to conclude that TRAP1 represents a promising target for the development of novel drugs for cancer treatment. The four Hsp90 paralogs share a highly conserved amino-acid sequence within the N-terminal ATP binding site, making selective inhibition of one isoform over the others a difficult task. Another challenge in creating TRAP1 inhibitors is mitochondrial accumulation. To reach the mitochondrial matrix, small molecules must traverse not only the plasma membrane but also both mitochondrial membranes, including the highly selective inner mitochondrial membrane. This membrane represents a particularly stringent barrier due to its high protein density, tightly regulated transport systems, and low permeability. As a result, compounds that readily enter cells often fail to accumulate within mitochondria at sufficient concentrations to interact with their targets. In addition, small changes in physicochemical properties can lead to large variations in mitochondrial uptake thus complicating the optimization of potency, selectivity, and pharmacokinetics. On the other hand, excessive mitochondrial accumulation can result in off-target toxicity, particularly by disrupting oxidative phosphorylation and inducing mitochondrial dysfunction.

The first strategy employed to develop TRAP1 specific inhibitors was the optimization of existing HSP90 inhibitors that target the N-terminal domain. Shepherdin, a peptide-based ligand, was among the earliest reported TRAP1-interacting compounds, able to disrupt mitochondrial function in cancer cells and induce apoptosis. Later studies, however, showed that the compound was not highly selective for TRAP1 but instead acted as a broad Hsp90 inhibitor. Subsequently, the strategy evolved towards the conjugation of mitochondrial-targeting moieties, such as triphenylphosphonium (TPP), to HSP90 inhibitory scaffolds. Mitochondria generate an inner negative electrochemical gradient across the inner membrane, therefore, the mitochondrion-targeting vehicle TPP, with its non polar and delocalized positive charge, is able to penetrate the membrane enabling drug accumulation. One of the first compounds developed by exploiting this strategy was Gamitrinib, which incorporates geldanamycin, a potent Hsp90 inhibitor, and TPP. This proved to suppress mitochondrial functions, cause PTP opening and induce cancer cell death. Gaminitrib remains though a preclinical drug as critical issues must still be resolved, such as compound formulation (it is water insoluble), tolerability, and potential side effects on tissues characterized by high levels of mitochondrial respiration.
The geldanamycin moiety was then substituted with a purine-based scaffold PU-H71, and the isopropyl amine of this was replaced with TPP using a C6 carbon linker; this resulted in SMTIN-P01[9]. This compound displayed selectivity for TRAP1 over its isoforms and a slight improvement in cytotoxity compared to gamitrinib. A later study, demonstrated how the length of the TPP linker influences inhibitor behavior. When the TPP was modified to have a C10 chain (as opposed to the C6 one), generating the so-called SMTIN-C10[10], inhibition of TRAP1 was increased. This results from the interaction of SMTIN-10 to an allosteric binding site at E115 in the NTD, in addition to binding to the ATP pocket, leading to a closed conformation of TRAP1. Alternatively, the purine ring of Hsp90 inhibitors was further optimized to increase TRAP1 binding, and a pyrazolopyrimidine derivative DN401 was found[11]. In earlier studies, this was shown to display mitochondria permeability and potent anticancer activity in cell-based and animal models, later works, however, suggested that DN401 behaves more like a pan-Hsp90 family inhibitor, affecting TRAP1, Grp94, and Hsp90 paralogs simultaneously, although it does not induce HSP70. A key limitation associated with N-terminal inhibition of Hsp90 is,in fact, the induction of the heat shock response. This is mediated by activation of heat shock factor 1 (HSF1), leading to upregulation of cytoprotective heat shock proteins, such as Hsp70, that promote cell survival and can counteract the intended anti-cancer effects. To overcome this limitation, several C-terminal Hsp90

inhibitors, such as novobiocin, have been developed. Also for these inhibitors, conjugation with mitochondrial targeting vehicles is a common strategy used; an example of a promising candidate is 6BrCaQ-C10-TPP[12] which manifests nanomolar GI-50 values against various human cancer cell lines, including breast (MDA-MB-231) and colorectal (HT-29, HCT-116).
An alternative approach to ATP-competitive inhibitor is allosteric targeting. This represents a compelling strategy as allosteric modulators, instead of binding to the active site of the protein, interact with distint regulatory sites that are typically less conserved between isoforms, allowing for greater specificity. Additionally, while orthosteric inhibitors often result in complete inhibition of protein activity, allosteric compounds can modulate function more subtly, providing a finer level of control and potentially reducing toxicity. However, despite these advantages, allosteric drug design remains challenging due to the difficulty in identifying suitable binding sites and achieving sufficient potency. An example of this strategy is HDCA[13] which specifically inhibits TRAP1 by binding to an allosteric pocket in the MD region.

1.2 Deep Learning and Drug Design

1.2.1 Introduction to Machine Learning

Machine learning (ML) is a branch of Artificial Intelligence (AI) in which a model learns patterns from data and then uses these to make predictions or decisions. This is used in numerous areas including image recognition, speech processing, medical imaging, fraud detection and of course drug design.
ML is commonly divided into three categories based on how the model learns from data: Supervised, Unsupervised and Reinforcement Learning. Supervised Learning involves training a model on labeled data, meaning each input comes with a correct output. The goal is for the model to learn a mapping from inputs to outputs so it can make accurate predictions on new data. Common tasks include classification and regression; such strategy is used in medical diagnosis and speech recognition. Unsupervised learning works with unlabeled data, where the model tries to find patterns on its own (ex. clustering): application examples include fraud detection and recommendation systems. Reinforcement learning is based on interaction with an environment. An agent learns by taking actions and receiving rewards or penalties, gradually improving its strategy to maximize long-term rewards. This is often used in robotics, gaming, and decision-making systems.
In this thesis, we will focus on supervised learning, where the starting point is a dataset where each data point is associated with features, that are measurable

properties or inputs that describe each data point, and a label which is the "ground truth" outcome that the model is trained to predict. Not all features are equally useful, some are highly predictive, meaning they strongly influence the predicted outcome, while others may be irrelevant or even harmful by adding noise. Part of a good model is selecting relevant features and removing unhelpful ones, this is known as feature selection. Selected features must then be transformed, through a process called feature engineering, into inputs that the model can actually learn from effectively. This can involve scaling numerical values, encoding categorical variables into numbers, or even creating entirely new features that better capture underlying patterns. The dataset is typically divided into training, validation, and test sets: the training set is used to learn model parameters, the validation set to optimise hyperparameters and guide model selection, and the test set to evaluate generalisation performance on unseen data. During training, the model repeatedly looks at the input features, makes predictions, and compares them to the true values. Based on the error it makes, it adjusts its internal parameters to better capture the relationship between inputs and outputs, this is known as model fitting. Alongside parameters, there are also hyperparameters, which are not learnt from the data but instead are set beforehand to control how the training process behaves, such as how fast the model learns or how complex it is allowed to be. After training, the model is evaluated on the separate validation set, this serves to estimate how well the model generalizes to adjust hyperparameters accordingly. Techniques like cross-validation improve this process by repeating it across multiple splits of the data, which reduces the dependence of the performance estimate on a single data split and also enables an efficient use of limited data: in a simple train/validation split part of the data is "wasted" for validation and cannot contribute to training. Even after validation, there is still a risk that the model has been indirectly tuned to perform well on the validation data. For this reason, a final test set is used. The test set is kept completely separate and untouched during both training and validation, and it is only used at the very end to evaluate the final model. This provides an unbiased estimate of how the model will perform on truly unseen data and serves as the most reliable indicator of real-world performance. After training and validation, and finally testing, the model is evaluated using appropriate performance metrics. The choice of metric depends on the type of problem: in classification tasks, measures like accuracy, precision, recall, and F1-score are common, while regression tasks often use metrics such as mean squared error or mean absolute error. These metrics provide a quantitative way to compare models. Throughout this process, several common problems can arise. One of the most important is overfitting, where the

model learns not only the true patterns in the training data but also the noise, resulting in excellent performance on the training set but poor performance on new data. The opposite problem, underfitting, occurs when the model is too simple to capture the underlying structure, leading to poor performance everywhere. Other pitfalls include data leakage, where information from outside the training process unintentionally influences the model, and imbalanced datasets, where one class dominates and can bias predictions. Overall the success of an ML architecture depends as much on how well the whole pipeline is designed and executed as it does on the specific model being used.

1.2.2 Evolution of Machine Learning Models

The development of machine learning has undergone a significant transition from classical "shallow" ML algorithms to neural network–based approaches. This was driven largely by the increase of data size and complexity, advances in computational power and novel algorithm implementations. Early machine learning methods employ classical statistics so derive relationships from data using mathematical probability models, such as linear regression. The aim is therefore to fit predefined functional forms.

With the rise of computational systems, early algorithmic learning methods started to emerge. One of the simplest early approaches is k-nearest neighbors or k-NN[14], developed in the 1950s–1960s. It is an instance-based algorithm, meaning it does not build a mathematical model during a training step but classification or regression is performed by identifying the most similar examples in the training data. As there is no training step, all computation happens at prediction time: distance between new point and all training points is computed and the k closest points are selected, these then drive the prediction.

One of the next steps in ML was the introduction of kernel methods that instead of taking k nearest points, use all points and weigh them by similarity, this is used to construct functions. These gained ground with Support Vector Machines (SVM), introduced by Vladimir Vapnik[15], which add a global optimization function. Such model is based on finding the hyperplane that maximizes the margin between classes. Although SVM is global, in the end it depends on a small subset of data, called support vectors, which are the points closest to the decision boundary. Such framework was later extended to regression problems through support vector regression (SVR).

Another type of model is Decision trees which are rule-based learning models where predictions are made through a sequence of if-then rules learned directly from data. For each feature, the dataset is split and different thresholds are tested, then the

best split is chosen. The best split is the one that separates the data into two groups where each group contains mostly one class. The dataset is then split into two child nodes and the same procedure is recursively applied to each subset.This recursive process continues until stopping criteria are met, such as node purity, maximum depth, or minimum sample size.

While single decision trees are capable of capturing complex and non-linear relationships in data, they show high variance: small perturbations in the training data can lead to substantially different splits, resulting in a completely different tree structure. This makes them prone to overfitting. To overcome this limitation, ensemble learning techniques were developed, which combine multiple models to produce more robust and accurate predictions. One of the most important models was the Random Forest algorithm[16]. A random forest constructs a large number of trees during training and aggregates their predictions, typically through averaging (for regression) or majority voting (for classification). At each split in the tree, only a random subset of features is considered, rather than evaluating all possible features, to determine the best split. This means that individual trees are de-correlated, reducing the risk that they all make the same errors.

Despite the success of these shallow ML algorithms, such approaches share a fundamental limitation: they rely heavily on manually engineered features. In these frameworks, raw data cannot typically be processed directly; instead, domain-specific knowledge must be used to construct descriptors that capture the relevant properties of the input, this is known as feature engineering. In fields such as cheminformatics, this has traditionally involved the design of molecular fingerprints or physicochemical descriptors. This means that performance of the model depends on completeness and correctness of prior assumptions about the problem. Feature engineering therefore becomes a limitation when underlying relationships are unknown or partially understood.

Neural networks (NNs) differ from traditional machine learning models in their ability to perform representation learning. Rather than relying on predefined features, these models learn hierarchical representations of data through multiple layers of nonlinear transformations. Lower layers capture simple patterns, whilst deeper layers progressively encode more abstract and task-relevant features. This hierarchical structure allows neural networks to model complex relationships.

NNs are inspired by biological neural systems as the brain is arguably the most powerful computational engine known today. In the brain, information is transferred by the interaction of neurons through synapses.

Therefore, NNs consists of artificial neurons, called nodes, connected by edges which

represent the synapses. They are based on the principle of applying non linear, differentiable activation functions to the linear combination of inputs, this enables the network to learn complex patterns and produce an output.
The first NN architecture was attributed to Frank Rosenblatt in 1958 with Perceptron[17], consisting of an input and output layer, that was designed to perform binary classification tasks. The Perceptron takes several input values, multiplies them by corresponding weights, sums them, and applies a threshold function to produce an output of either 0 or 1. The limitation of such model was that it could only solve linearly separable problems.
Subsequently, a single hidden layer was added between the input and output layer, enabling the model to learn nonlinear complex patterns.

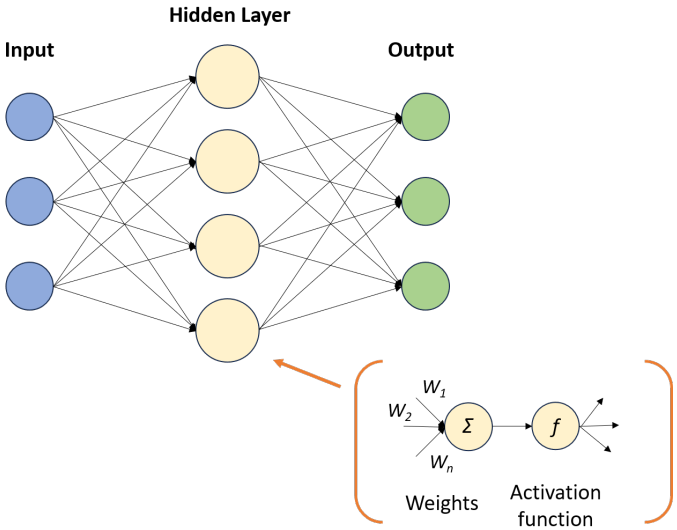


Figure 1.2: Single hidden layer NN

The input layer receives input data directly by putting a feature into each node. Then, each node in the hidden layer receives a weighted linear combination as input from all the units in the input layer and then uses an activation function to perform a nonlinear transformation. This same mechanism happens from hidden to output layer. The learning process iteratively adjusts the weights in order to correctly make the prediction. While single hidden layer NNs can theoretically approximate many complex functions, in practice they often require a very large number of neurons to do so efficiently. This makes them computationally expensive and harder to train. Additionally, they struggle to capture hierarchical patterns in data, such as those found in images or language. During the 1960s-1970s, researchers realized that adding multiple hidden layers could overcome these limitations, the issue then became that there was no efficient way to train multi-layer networks. The real break-

through came with Geoffrey Hinton, David Rumelhart and Ronald Williams who demonstrated that backpropagation[18] could be applied to the training of multi-layer neural networks. This consists of computing the output error and then systematically propagating that error backward through each layer using the chain rule. The chain rule allows us to determine how each hidden-layer weight affects the final loss by expressing this contribution as a product of gradients flowing backward through the network. After BP, each weight is updated using gradient descent to minimize loss. Backpropagation, therefore, established modern neural network training and allowed the transition from "shallow" neural systems to deep neural networks (DNNs). Deeper architectures significantly increase computational complexity and multilayer neural networks with many layers are prone to the vanishing gradient problem, which can hinder effective weight updates and make training difficult. To address these challenges, the development of DNN models has relied heavily on GPU acceleration to improve computing power. Furthermore, network architectures were modified to optimize weight initialization and optimization, and different transfer functions and regularization techniques were adopted to minimize overfitting.

1.2.3 Drug Design and Machine Learning

Drug design is a central pillar of pharmaceutical research, aiming to identify and optimize compounds with therapeutic potential while minimizing adverse effects.

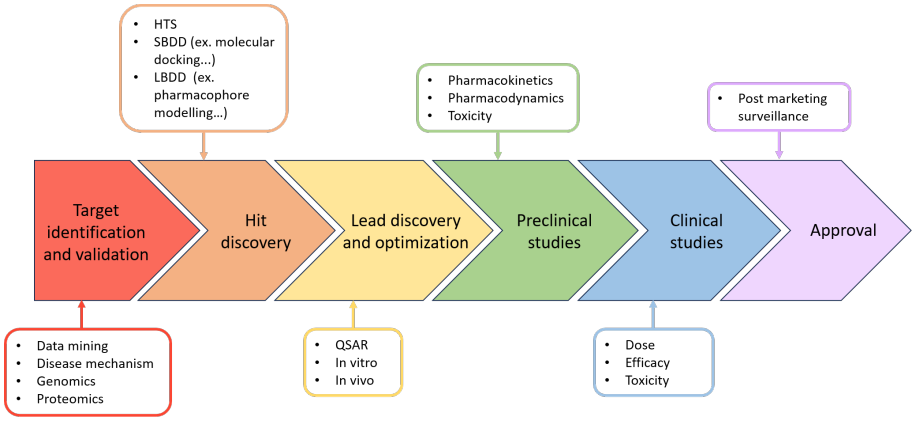


Figure 1.3: Drug design and development pipeline

The first step of the process is identifying the biological target, which can be a protein, a specific nucleic acid sequence or a pathway. This is followed by the discovery of molecules, the so-called "hits", that show target activity. After identifying the potential drug candidates, these must be transformed into "leads" by increasing their drug suitability. The main approach is to alter the core structure of a molecule and adjust substituents. In this hit-to-lead stage, analyses, such as physical-chemical properties, SAR and ADMET, are carried out. Then lead compounds are optimized through the same property and pharmaco-dynamic and kinetic analyses plus also in vitro and in vivo testing.

Traditionally, the drug design process has heavily relied on human expertise, combining experimental screening and medicinal chemistry intuition. Despite enormous progress in the field, drug development remains an expensive and time-consuming process, as it can take approximately 10-15 years, usually with low success rates. One of the primary challenges in drug design lies in the immense size and complexity of chemical space. It is estimated that the number of possible drug-like molecules can exceed 10^{60}, making exhaustive exploration infeasible. Furthermore, biological systems introduce additional layers of complexity: drug candidates must not only bind effectively to their targets but also exhibit suitable pharmacodynamic and ADMET properties (pharmacokinetics plus toxicity). Whilst, early-stage drug discovery has historically relied on high-throughput screening (HTS) and structure–activity relationship (SAR) studies, both are resource-intensive. A key development in addressing the scale limitations of experimental screening was virtual screening (VS). This enables the in silico evaluation of millions of molecules, significantly reducing

the number of candidates that must be experimentally validated. Virtual screening can be divided in structure-based (SBVS) and ligand-based (LBVS). SBVS relies on the three-dimensional structure of a biological target and employs molecular docking techniques to predict how candidate molecules bind within the active site. LBVS, instead, does not require explicit structural information about the target but uses known active compounds to identify new candidates with similar physicochemical or structural features (pharmacophore modeling, QSAR...). Although virtual screening represented a major advance in computational drug discovery, early implementations were largely based on physics-inspired models and empirical scoring functions.

Over the past decade, progresses in information technology and automation have enabled the production and collection of enormous quantities of compound activity and bio-medical data. However, it is not feasible to process such amounts of data and discover underlying patterns through classical statistical approaches and human evaluation alone. In recent years, Artificial Intelligence (AI) and its subset Machine Learning, have increasingly been adopted, also in the pharmaceutical industry, to help address this issue. Early ML tasks were supervised learning tasks where algorithms are trained using explicitly labeled datasets. Such tasks are distinguished into classification, where a class prediction is made, and regression tasks, in which a numerical value is predicted. The other two types of ML are unsupervised learning and reinforcement learning. In the former, the model discovers hidden patterns in unlabeled data, instead for the latter, an agent is trained to make decisions, through trial-and-error, by maximizing rewards. Supervised learning methods, such as random forests, are generally applied to predict drug-target interactions and ADMET properties. Unsupervised learning, such as clustering, help with grouping compounds, whilst reinforcement learning methods, although less common, is gaining ground in optimizing chemical synthesis pathways.

ML techniques have also determined a significant transformation in virtual screening. ML-based scoring functions have been developed to improve the prediction of binding affinity by learning complex patterns from experimental data. Hybrid approaches have also emerged, where classical docking is combined with machine learning-based rescoring or filtering, leading to improved prioritization of candidate compounds.

A branch of ML that has had a huge impact in the pharmaceutical field is Deep Learning (DL), which utilizes neural networks with multiple hidden layers to capture complex, non linear relationships in data. This makes DL networks capable of handling very large, high-dimensional datasets with billions of parameters that pass through nonlinear functions. The first application of deep learning in drug de-

sign consisted of adapting existing neural network architectures to classical QSAR and QSPR problems. Then DL architectures were extended to a vast number of applications such as target identification and validation, small molecule design, protein–protein interactions, reaction prediction, synthesis planning, ect...

1.2.4 Deep Learning and Molecular Generation

The integration between DL and computational chemistry has led to the development of de novo design methods which have found widespread applications, including drug design and has been increasingly explored in areas such as enzyme engineering, bio-manufacturing and materials science.
De novo drug design consists of generating novel compounds with desirable pharmacological and physicochemical properties. Unlike virtual screening, de novo methods allow for goal-directed molecular optimization and discovery of novel scaffolds without needing to depend on large pre-existing libraries. De novo molecular generation primarily employs two strategies. The first, "Holistic Generation", creates an entire molecule from scratch, whilst the second approach, "Iterative Generation", builds molecules step-by-step.
Molecules cannot be directly processed and must be transformed into a machine-readable format. Molecular representation is a critical step in the modeling pipeline as this determines not only the choice of architecture but also syntax constraints, generation speed and post-generation evaluation. The earliest and still most-widely used representations encode molecules into one-dimensional strings. The most popular string notation is SMILES which is generated by traversing a two-dimensional molecular representation, called molecular graph, and recording the atomic symbols and bond types along the traversal path. Side branches are enclosed inside parentheses, and closed rings are indicated by matching numeric labels at the bond break points. To address limitations in the generation process, for SMILES, such as chemical validity, notations like DeepSMILES and SELFIES were later developed.
DeepSMILES removes paired notation for rings and branches and only uses a single token at the closure of the ring or the end of the branch; this way the generative model cannot produce an "unmatched pair" error and result in an invalid string. SELFIES, instead, make use of predefined semantic tokens together with a grammar that enforces chemical rules, such as atomic valency, during the generation process; this ensures that every possible string corresponds to a valid molecule. However, enforcing chemical validity at the representation level may remove useful learning signals that teach models to avoid chemically implausible regions.

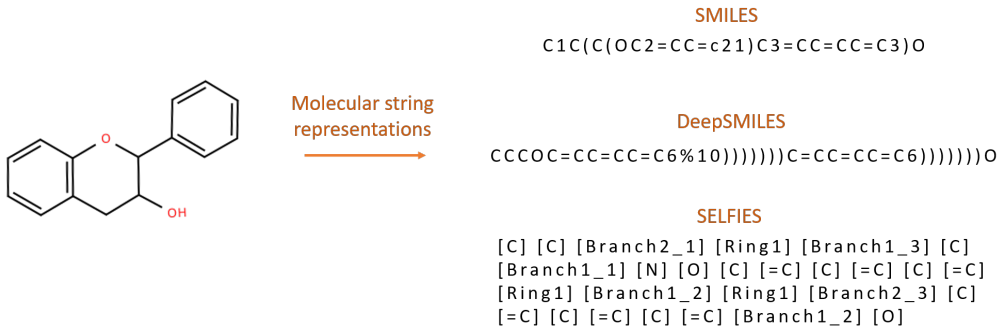


Figure 1.4: Most popular molecular string representations

A more intuitive way of representing molecules is through 2D and 3D molecular graphs, where atoms are nodes and chemical bonds are edges. In 3D molecular graphs, conformational information can be included through atomic coordinates or derived geometric features such as distances, bond angles and torsional angles.
Once a molecular representation has been chosen, the next step is molecule encoding, so transforming this structural notation into numerical representations that can be processed by machine learning models. The amount of molecular information preserved during this conversion depends on the chosen representation and encoding scheme. Ideally, this step should preserve the molecular information relevant to the task. For molecular strings common encoding methods include one-hot encoding, which converts each token into a unique binary vector, and learnable embeddings, where tokens are represented as vectors of unique numbers that are updated during training. Graphs are encoded using a node feature matrix capturing atomic properties (e.g., element type, charge, chirality), an edge feature matrix describing bond characteristics (e.g., bond type, ring membership), and an adjacency matrix defining connectivity.
There are currently four main strategies that can be used to achieve goal-directed molecular generation toward a specific target:

1. generate-and-filter approaches, where large molecular libraries are produced and subsequently screened (e.g., via virtual screening);

2. data-driven biasing, such as fine-tuning generative models on known active compounds for a target;

3. objective-driven optimization, where scoring functions (e.g., QSAR models or docking predictors) are integrated into reinforcement learning or related frameworks

4. structure-based generation, where protein–ligand interactions or binding pocket information are explicitly incorporated into the model.

These strategies are not mutually exclusive and are frequently combined in practical workflows. For example, generative models are often fine-tuned on target-specific data and subsequently optimized using reinforcement learning with QSAR or docking-based scoring functions.

The most common ML architectures used to build generative models are: Recurrent Neural Networks (RNN), Variational Autoencoders, Transformer-based models, Generative Adversarial Networks (GAN) and Diffusion models. Recurrent Neural Networks typically operate on 1D string representations and are well suited for molecular generation as they process sequential information, one token at a time. The type of RNN unit typically used is either the long short term memory (LSTM) unit or the newer gated recurrent unit (GRU), both of which mitigate the exploding and vanishing gradient problems that occur during training when capturing long-term dependencies. An exhaustive description of this architecture, including the training process, will be detailed in the next chapter (Materials and Methods). RNNs can be considered the first widely adopted generative model in the field of drug design. Early studies[19,20] quickly demonstrated that such models could learn syntactically valid molecular structures and generate novel compounds, establishing RNNs as a baseline architecture for generative chemistry. Then Segler et al.[21] showed that fine-tuning an LSTM-RNN generative model on target-specific ligands could bias molecular generation toward such chemical space, this was one of the first approaches for target-directed molecular generation via fine-tuning. In this approach, the model is first trained on a large and diverse dataset of molecules to learn the general syntax and structure of valid SMILES strings. Subsequently, additional training, known as finetuning, is carried out on a smaller, target-specific dataset to produce focused libraries containing compounds with similar structural scaffolds and physicochemical properties to those associated with the target.

Gupta et al.[22] formalized this two-stage process within a transfer learning framework, thereby establishing a general and reusable methodology for target-directed de novo molecular design. This is the approach that was employed in the present project, in which, to develop a target specific CLM, a RNN-LSTM was trained first on ~1.5 million bioactive compounds extracted from the ChEMBL database and this pretrained model was then finetuned, by transfer learning, on TRAP1 known active binders.

Transformers, instead, look at all parts of a sequence at the same time learning which parts are important for each other (self-attentive mechanism), it is therefore particularly powerful at capturing long-range dependencies within molecular structures. Recently, a comparison between RNNs and Transformers has shown that the former generates more structurally diverse design libraries, while the latter better captures complex properties of molecules[23]. Examples of Transformer based frameworks are MolGPT and ChemBERTa, although the latter is not used for molecular generation but rather for the prediction of chemical properties by learning statistical relationships within SMILES strings and producing embeddings that capture molecular similarity and activity patterns.

Variational autoencoders (VAEs) comprise a dual neural network architecture with an encoder, that maps molecular structures into a probability distribution over a latent space, and a decoder that learns to reconstruct molecules from such distribution mapping them back to valid representations. The encoder and decoder can be any deep learning model, with convolutional neural networks (CNNs) and RNNs being typical choices for molecular sequences. A major training issue in VAEs is posterior collapse, where they generate only molecules with low diversity, this is driven by the decoder that becomes so strong that it ignores the latent vector, therefore, latent space stops carrying meaningful information. One of the earliest and most influential applications of variational autoencoders to molecular generation was introduced by Gómez-Bombarelli et al[24] .

Generative Adversarial Networks (GANs) also involve a dual architecture, these are two competing networks: a generator and a discriminator. The role of the discriminator is to distinguish whether the molecule it is processing was generated by the generator or came from the training data. In GAN training, the objective of the generative model becomes to try to fool the discriminator rather than maximizing likelihood. Examples of these are ORGANIC and MolGAN (this improves the rate of valid molecules produced), such methodologies have mostly been applied in the field of molecular optimization.

Finally, Diffusion models, inspired by nonequilibrium statistical physics, represent the state-of-the-art for 3D molecular generation. They function in two stages: a fixed "forward process" that adds noise and a learned "reverse process" that iteratively denoises samples into valid molecules. During training, noise is added to real molecules and the model learns how to reverse this process step by step. During generation, the model starts from random noise and repeatedly denoises it until a valid molecular structure is formed. The primary limitation of this framework is computational cost; the iterative denoising process results in slow sampling speeds

compared to VAEs or GANs making them less suitable for ultrahigh-throughput virtual screening pipelines.

Clearly no single generative model is universally optimal, therefore a common trend is to employ hybrid approaches that combine strengths of different model families to mitigate their individual weaknesses. An example is the Sc2Mol framework which uses, to begin with, a VAE to generate a molecular scaffold and then a Transformer to decorate it with functional groups. Furthermore, RL is frequently integrated with generative models, particularly GANs, to create goal-directed systems.

An emerging approach in de novo drug design is Active Learning (AL) which is built on the principle that models can achieve higher accuracy with less data by intelligently selecting training samples. Traditionally AL was used to improve model predictions but is now exploited to explore chemical space. The core idea is an iterative loop where a model generates molecules, evaluates them based on predicted performance and uncertainty, and selectively retrains on the most informative candidates. By doing so, AL enables a more efficient exploration with fewer experiments. A key component of this process is uncertainty estimation, where the model assesses how confident it is in its predictions. Techniques such as Bayesian modeling or ensemble learning are often used to quantify this uncertainty. Predictions from the model, along with their associated uncertainties, are used to prioritize compounds for further study. The molecular candidates can originate from various sources, including existing databases, combinatorial libraries, or generative models, making the approach flexible and compatible with both existing and de novo design workflows.

Despite extraordinary progress, several challenges hinder the widespread adoption of generative molecular design frameworks. A fundamental limitation common to deep learning approaches, both in general and in the context of de novo molecular generation, is data scarcity. Even comparatively large bioactivity datasets typically comprise only a few thousand molecules, which represents a significant bottleneck for training data-intensive models. Transfer learning has been proposed as a strategy to mitigate this limitation. Experimentally confirmed inactive molecules can also be considered a valuable resource in low-data drug discovery, as they are typically more abundant than active compounds. One of the most pressing challenges in de novo drug design is out-of-distribution (OOD) generation. Whilst generative models can theoretically produce molecules outside the training distribution, current models often struggle to generate truly novel and practical scaffolds that fall outside known chemistry. On the one hand, similarity to known bioactive compounds increases the probability of identifying viable drug candidates, as structurally similar molecules often share biological activity. In the other hand, excessive similarity

hinders chemical innovation by restricting the explored novel chemical space. This limitation can be mitigated by aiming to minimize scaffold similarity while maintaining three-dimensional information; bioactivity is strongly influenced driven by three-dimensional shape and electrostatics complementarity. For example, some strategies condition the generation using pharmacophore or three-dimensional pocket information. Another challenge is evaluating generative models. Whilst predictive models can be evaluated on held-out molecules that were previously tested, de novo generated molecules have no associated tested properties. This makes it difficult to compare architectures, choose which model is suitable for a particular task and identify learning gaps. To overcome this limitation, in the last few years, several benchmarking attempts have been made to supply standardized datasets and evaluation metrics. The most well-known benchmarking platforms are GuacaMol and MOSES. The former is centered on distribution-learning and goal-directed models while the latter focuses only on distribution-learning. Although DL architectures allow the production of novel molecules, a difficulty that arises is the synthetic feasibility of such compounds. Common strategies include synthetic metric calculations, such as synthetic accessibility score (SASscore), and retro-synthetic route predictions. In order to convert computationally discovered hits into viable drug candidates, experimental validation is necessary. Nevertheless, experimental testing is expensive and time-consuming so in reality only a small fraction of generated compounds can be synthesized and evaluated. Such process presents two main difficulties: selection of molecules for evaluation and experimental validation. Generative models typically produce large libraries of compounds that achieve comparable scores under computational evaluation metrics, making it difficult to prioritise specific candidates for synthesis. Such scoring functions are often noisy and may not reliably capture small but meaningful differences between molecules. Furthermore, predictive models may exhibit systematic errors, particularly when evaluating out-of-distribution compounds. Therefore, to identify the optimal subset robust selection strategies must be carried out in order to balance the exploitation of high-scoring compounds with the need to maintain diversity thus mitigating prediction bias. To address the second challenge, experimental validation, the following must be considered. Predictive models are evaluated on test sets containing hundreds to thousands of examples which enables statistically robust performance estimates. On the other hand, for generative models, the fraction of experimentally validated molecules is small, therefore, evaluation may be subject to high variance and potential selection bias.

Chapter 2

Materials and Methods

2.1 CLMs and RNN-LSTM networks

A Chemical Language Model or CLM is a deep learning model that is able to process molecular information in string format and generate novel molecules with desired properties, which can be chemical-physical or pharmacological characteristics or properties related to the biological activity, and depends on the chosen ML model and generative strategy. This, therefore, falls within the field of de novo drug design. CLMs adapt natural language processing (NLP) algorithms to learn chemical language from one-dimensional molecular data. By chemical language we mean both syntax, so chemical validity, and semantics, therefore molecular properties which depend on the atoms present and their connectivity.
Since computers process data and learn through matrix operations, deep generative models used in chemistry require a precise molecular representation. Molecules are generally represented as SMILES (Simplified Molecular Input Line Entry Specification) or SELFIES (Self-Referencing Embedded Strings). In SMILES notation, atoms are indicated with their atomic letters, bonds and branching with symbols, and ring opening/closure with numbers. SELFIES, instead, use predefined semantic bracketed tokens that enforce chemical rules, ensuring the production of only valid molecules. Whilst with SMILES, each token meaning depends on its position so on the surrounding characters, for SELFIES each token carries built-in constraints. The latter notation consequently solves the chemical invalidity problem of SMILES. This means that from a machine learning perspective, with SELFIES representation, compared to SMILES tokens, the syntax-learning task is simplified. Despite this advantage, SMILES is generally used over SELFIES as it is more human readable, has a shorter sequence and more importantly is deeply integrated into tooling and datasets (CHEMBL, PubChem...).

CLMs can be broadly grouped into three categories based on how they learn and generate molecular structures: distribution learning, conditional learning and goal-directed learning. In distribution learning, the model learns the overall statistical distribution of molecules in the training dataset so the generated molecules share the same chemical space. In goal-directed learning, the objective is to generate molecules that optimize a specific scoring function, ex. predicted binding affinity, drug-likeness (QED)... In conditional learning, the model generates molecules given a specified condition, this can be property based or a structural constraint (scaffold). Such classification is not exhaustive, also a common approach is to combine these strategies, for example, a model can be firstly trained with distribution learning and then optimized or conditioned towards particular properties.

In this project, a RNN of LSTM type is used for generating TRAP1 specific compounds. Firstly, the model is pretrained to learn the general distribution of syntactically valid and chemically relevant molecules, subsequently the pretrained model is fine-tuned on a dataset of TRAP1 active compounds to bias the learned distribution towards molecules associated with this target protein. As a result, the model performs distribution learning over SMILES representations, progressively shifting from a general chemical space to a TRAP1-focused chemical subspace. RNNs are trained to generate one character at a time, based on the preceding portions of the molecular string, so they can be powerful generative tools.

Other popular architectures for de novo molecular generation are transformers, encoder-decoder frameworks, VAEs and GANs.

RNNs are neural networks used to generate and manipulate sequentially structured data, such as one dimensional molecular representations (SMILES). RNNs receive a sequence input and use a set of hidden layers connected in a recurrent manner to transform the discrete token representation (generated from one-dimensional data), into a continuous representation. The continuous representation is then fed into a feedforward network to predict the adjacent token in the sequence. The advantages of RNNs lie in the variable-length molecule generation and their straightforward training procedure, typically based on maximum likelihood estimation, compared to other architectures, such as GANs and VAEs.

In this project, the training consists of two basic steps. First, in the pretraining step, we develop a generic model that learns the chemical syntax and semantics of drug-like molecules from a large unfocused compound set. In a second step, we fine-tune this generic model, by transfer learning, on a small target-focused library of actives. For pretraining we used ~1.5 million drug-like compounds from the ChEMBL database and for finetuning 43 known TRAP1 selective actives.

The core idea of a RNN is to process sequential data by updating a hidden state at every step of the sequence. Each current hidden state depends on all the previous ones; this way sequence order is captured as well as long-range dependencies. Such procedure can be described by the following diagram:

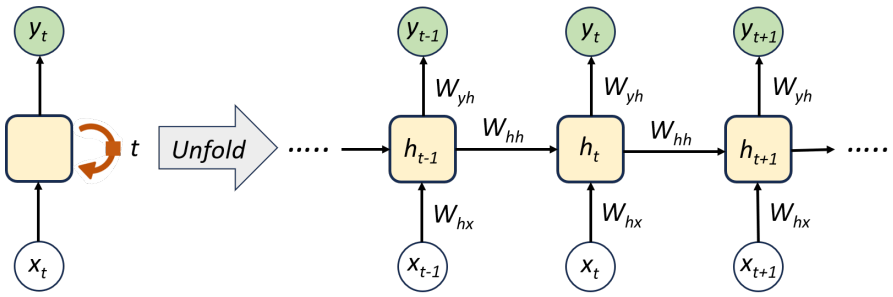


Figure 2.1: RNN network unfolded

At time t, the input x_t is passed to the hidden state which is updated as:

$$h_t = a_h(W_{hx}x_t + W_{hh}h_{t-1} + b_h) \tag{2.1}$$

where a_h is the activation function, W_{hx} is the matrix containing the weights of the connections between input and hidden nodes, W_{hh} is the matrix containing the weights of the internal connections in the hidden layer, h_{t-1} is the previous hidden layer and b_h is the vector containing biases.
The output $\hat{y}_t$ is then calculated as:

$$\hat{y}_t = a_y(W_{yh}h_t + b_y) \tag{2.2}$$

where a_y is the activation function, W_{yh} is the matrix containing the weights of the connections between hidden nodes and output and b_y is the vector containing biases. During training, the goal is to optimize the weights W and biases b to best fit the data and therefore minimize the loss between the prediction (model output) and the ground truth values (in the training set). One of the most widely used loss functions is cross entropy:

$$L = -\sum_{i=0}^{n-1} y_i \; log \; (\hat{p}_i) \tag{2.3}$$

where L is the loss, y_i is the real class of the sample x_i, and $\hat{p}_i$ is the probability estimate, obtained by applying the softmax function (which converts a given number to a value between 0 and 1) to the observed answer of the model. To be able to optimize the model's parameters, we must know how each parameter contributes to the model's predictions at every time step. To achieve this, the chain rule is applied and the gradient of the loss function is decomposed into a product of local

derivatives; this gradient computation is called BPTT (Back Propagation Through Time)[25]. BPTT starts from the last time step and continues until reaching the initial time step, t = 0.
Computation, therefore, proceeds in two phases: a forward pass, in which inputs are propagated through the network to compute the output and evaluate the loss, and a backward pass, in which gradients are propagated from the output layer back toward the input layer. During this backward phase, each layer is assigned a loss gradient with respect to its output, which quantifies how changes in its output affect the overall loss.

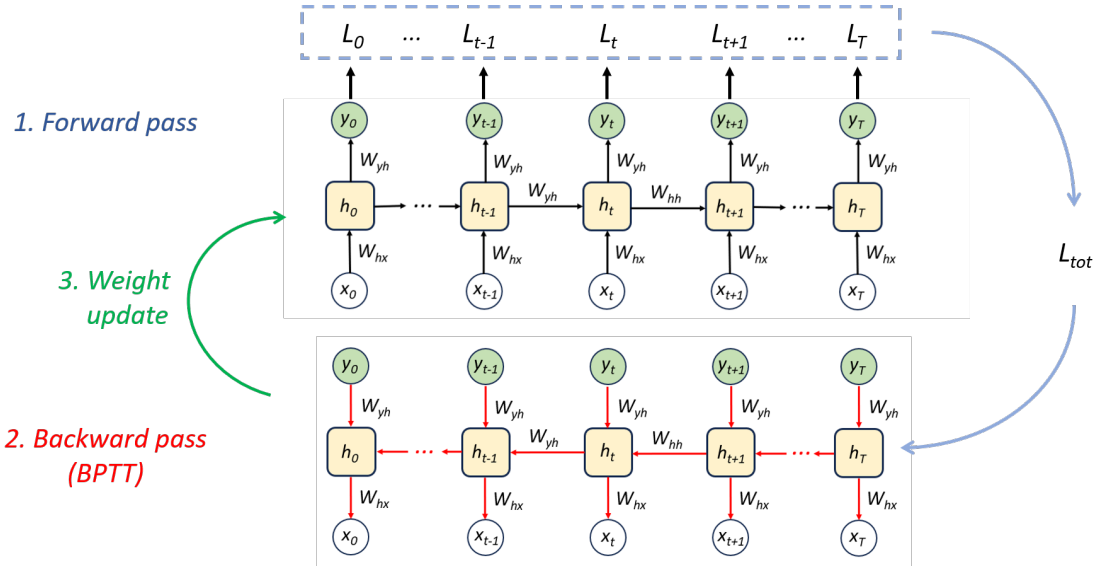


Figure 2.2: RNN training diagram

More in detail, with BPTT, starting from the final timestep:

$$\frac{\partial L}{\partial h_t} = \frac{\partial L_T}{\partial h_t} \tag{2.4}$$

at each step, the recursive formula is applied:

$$\frac{\partial L}{\partial h_t} = \frac{\partial L_t}{\partial h_t} + \frac{\partial L}{\partial h_{t+1}} \cdot \frac{\partial h_{t+1}}{\partial h_t} \tag{2.5}$$

By recursively substituting the same expression for $\frac{\partial L}{\partial h_{t+1}}$, $\frac{\partial L}{\partial h_{t+2}}$ and so on, the temporal dependence is unfolded across all timesteps up to T. The full expanded form is therefore:

$$\frac{\partial L}{\partial h_t} = \sum_{k=t}^{T} \frac{\partial L_k}{\partial h_k} \cdot \prod_{j=t+1}^{k} \frac{\partial h_j}{\partial h_{j-1}} \tag{2.6}$$

where each hidden state receives a gradient from its own loss and gradients of future timesteps via a chain of derivatives. This way the contribution of each hidden state h_t to the total loss is calculated, by combining its immediate effect on the current timestep's loss with its indirect effect on future timesteps through recurrence. The inner product over time is a chain of Jacobian matrixes, thus the expanded BPTT formula can also be written as:

$$\frac{\partial L}{\partial h_t} = \sum_{k=t}^{T} \frac{\partial L_k}{\partial h_k} \cdot \prod_{j=t+1}^{k} J_j \tag{2.7}$$

We are now able to compute the loss gradients with respect to the model's parameters, so weights and biases. For each parameter θ, again, the chain rule is applied:

$$\frac{\partial L}{\partial \theta} = \sum_{t=1}^{T} \frac{\partial L}{\partial h_t} \cdot \frac{\partial h_t}{\partial \theta} \tag{2.8}$$

The goal of the first stage, computation of $\frac{\partial L}{\partial h_t}$ for all hidden states, is to identify which hidden states contribute to the prediction error whilst the second stage, calculation of $\frac{\partial L}{\partial \theta}$ for each parameter, determines how weights and biases must be changed to minimize the error.
BPTT computation is followed by loss optimization, which is performed using gradient-based optimization.
For example, the gradient descent algorithm[26] calculates the gradient of the loss L with respect to each parameter θ and updates the latter according to the equation:

$$\theta \leftarrow \theta - \eta \frac{\partial L}{\partial \theta} \tag{2.9}$$

where η is a positive scalar called learning rate, which determines the magnitude of the change. The choice of η is crucial for the identification of the minimum in the loss function: if the learning rate is too small, convergence becomes slow and computationally expensive, whereas if it is too large, overshooting may occur, preventing convergence to a good minimum. Thus, different η values are tested during hyperparameter search, to identify the learning rate that best minimizes the loss. Usually, the optimization is repeated until the gradient of loss L with respect to θ is close to zero or until a set maximum number of steps is reached. That said, local minima and saddle points are also associated with a gradient value equal to zero, such condition therefore is not sufficient to indicate that global minimum is reached. Also, in gradient descent the loss L is computed using all training data so calculation time increases linearly with the size of the training set.

To overcome these problems, stochastic gradient descent[26] is usually preferred:

$$\theta \leftarrow \theta - \eta \sum_{i \epsilon B} \frac{\partial L_i}{\partial \theta} \tag{2.10}$$

Here, at each training iteration, optimization is carried out only on a small batch of randomly selected training samples. The calculation is therefore much faster compared to the classic gradient descent algorithm. Also by introducing stochasticity into the process, the optimization trajectory can continue to evolve, as these noisy gradient estimates are unlikely to be exactly zero, allowing the model to escape flat regions and saddle points.

As already mentioned, BPTT computation involves repeated multiplication of Jacobian matrices across timesteps. If the matrix norms are smaller than 1, then the multiplication causes the gradient to shrink exponentially, resulting in vanishing gradients. If, instead, the Jacobians have values greater than 1, the gradient will grow exponentially leading to exploding gradients. Consequently, early timesteps, in recurrent neural networks, are particularly susceptible to vanishing or exploding gradients because their associated gradients are propagated through a longer sequence of Jacobian multiplications. Vanishing gradients hinder long-range dependency learning whilst exploding gradients destabilize training by causing excessively large parameter updates.

To overcome these limitations, Long Short-Term Memory (LSTM) networks were introduced[27]. The key to LSTMs is the cell state which stores long-term memory through gates that control the flow of information through the network:

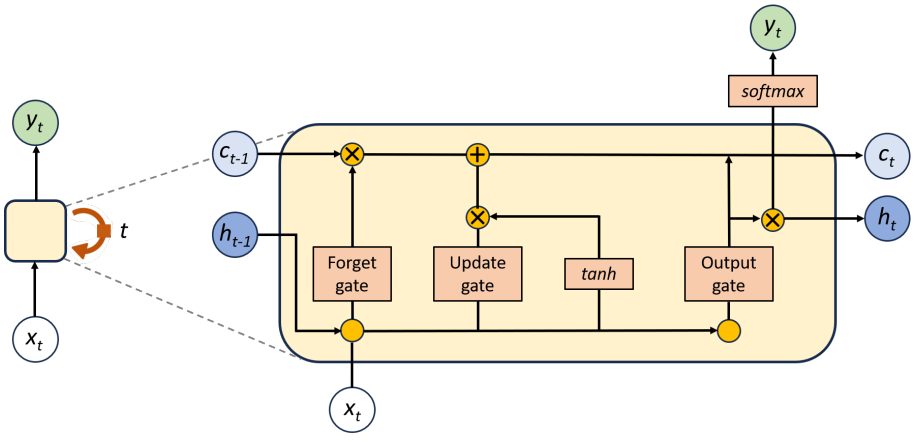


Figure 2.3: LSTM cell

An LSTM cell contains three gates and one cell state. The first gate is the forget gate :

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + bf\right) \tag{2.11}$$

that multiplies the previous cell state by a value between 0 and 1, deciding how much information from the previous time step to retain or discard.
The next step is to decide how much new information from the current input is to be added to the memory. This consists of two parts. First, the input gate decides which values will be updated:

$$i_t = \sigma\ (W_i \cdot [h_{t-1}, x_t] + bi) \tag{2.12}$$

and then a new candidate cell state is generated:

$$\tilde{C}_t = tanh\ (W_C \cdot [h_{t-1}, x_t] + b_C) \tag{2.13}$$

which represents the potential new memory content based on current input and past hidden state. At this point the old cell state C_{t-1} is updated into the new one C_t:

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{2.14}$$

Finally, the output gate multiplies the updated cell state (after passing through a tanh) by a gating value to determine how much of the long-memory will be part of the output and so how much information is passed on to the next timestep:

$$o_t = \sigma\ (W_o \cdot [h_{t-1}, x_t] + bo) \tag{2.15}$$

The hidden state of the current step is therefore updated as:

$$h_t = o_t \cdot tanh(C_t) \tag{2.16}$$

In LSTMs, therefore, the hidden state h_t represents the "short-term memory" as it is used to predict the output of the current timestep. Instead, the cell state C_t is the "long-term-memory" which stores information across many timesteps and is not directly used for predictions.
By introducing a gated cell state with additive updates, LSTMs mitigate vanishing and exploding gradients, as they enable gradients to propagate through time with controlled scaling. During training and gradient optimization, besides weights and biases (typical for classic RNNs), gates are also updated.
As already mentioned, in this project, an LSTM network was implemented to generate target specific ligands.
We shall now describe in detail the two steps of pretraining and finetuning, the sequential procedure for both of these is nearly identical.

The pretraining dataset consists of ~1.5 million bioactive compounds from the ChEMBL database whilst the finetuning set a small ensemble of TRAP1 known actives (43 molecules); both are written in SMILES notation.
Firstly, as for most supervised ML models, the initial dataset must be split: the pretraining dataset is split into training (~95 %), validation (~2.5%) and test (~2.5%) set, instead, the finetuning set is only split into training and validation with a 80:20 ratio.
Starting from pretraining, which from here on we shall call simply training, each SMILES string is tokenized: each character of the string is treated as a token. This is necessary as neural networks, which process data through matrix operations, cannot directly operate on raw text strings but these must be converted to numbers. In order to perform tokenization, a character vocabulary must be defined which contains all possible characters that appear in the dataset. For SMILES this typically includes atoms (C,N,O...), bonds, structure symbols and numbers. Special tokens are also defined, such as <SOS> and <EOS> , respectively, "start of SMILES" and "end of SMILES".
Each character (and so token) is then mapped to an integer ID using a fixed user-defined vocabulary. Each molecular string therefore is encoded into a tensor:

$$x = [x_1, x_2, x_3, ..., x_T] \tag{2.17}$$

This tensor is split into an input sequence:

$$x = [x_1, x_2, ..., x_{T-1}] \tag{2.18}$$

and a target sequence:

$$x = [x_2, x_3, ..., x_T] \tag{2.19}$$

To enable the model to process these discrete symbols, each integer index is mapped to a continuous vector representation using an embedding matrix. An embedding matrix is a trainable parameter table that maps each discrete token to a continuous vector of real numbers. Formally, it is a matrix of size $V \times D$, where V is the vocabulary size and D is the embedding dimension, and each row corresponds to a learnable representation of a token. At the start of training, these vectors are initialized randomly, meaning they carry no chemical or linguistic meaning. During training, through backpropagation, the loss gradients are used to update not only the weights, biases and gates of the network but also the rows of the embedding matrix. Over many iterations, tokens that appear in similar contexts receive similar gradient updates, causing their vectors to move closer together in the continuous

space.
Therefore the input of the LSTM network is a sequence of continuous representations for each molecular string. The embedded sequence is then processed step by step. At each step, the x_t token enters the network, along with the previous states h_{t-1} and c_{t-1}, and after the h_t and c_t update through gate computation, an output is produced. The output for each timestep is a vector of real numbers, called logits:

$$z = [z_1, z_2, z_3, ..., z_V] \tag{2.20}$$

Logits are then converted into a probability by the softmax function:

$$\hat{y}_t = softmax(z_t) \tag{2.21}$$

where

$$\hat{y}_t[i] = \frac{e^{z_t[i]}}{\sum_j e^{z_t[j]}} \tag{2.22}$$

After softmax, at each timestep t, we have a probability distribution over all possible next tokens.
So each element

$$\hat{y}_t(i) = P(next\ token = i \mid x_1, x_2, ...x_t) \tag{2.23}$$

is the probability that the next token is i given all the tokens the model has seen so far. At this point, we compare the probability vector $\hat{y}_t$ to the ground-truth token y_t by calculating the token loss (cross entropy):

$$L_t = -log\ \hat{y}_t\ [y_t] \tag{2.24}$$

this is a measurement of how well the model is predicting the next character. At the end of the sequence prediction, the total loss is calculated:

$$L_{tot} = \sum_{t=1}^{T} L_t \tag{2.25}$$

Now the total loss is propagated backwards in time through BPTT. This means a calculation of gradients with respect to hidden states for each timestep, which are then used to compute a single gradient, across all timesteps:

$$\frac{\partial L}{\partial \theta} \tag{2.26}$$

for all trainable parameters, including the weights and biases associated with each

LSTM gate, the embedding matrix, and the output layer.
The model then uses these gradients to update the parameters by means of an optimizer. In the implemented LSTM network, the Adam optimizer is used which belongs to the family of stochastic gradient-based, or SGD, optimizers. Adam optimizer extends SGD by incorporating momentum (remembering the direction of past gradients) and adaptive learning rates.
To sum up, LSTM training consists of three main steps: a forward pass that computes predictions and losses, a backward pass that computes gradients and finally a parameter update step. This entire process is carried out on all of the SMILES of the training set. Instead, for the validation set, we only have a forward pass, with the computation of the validation loss, so no BPTT or weight update takes place. This is because the purpose here is to evaluate the model on unseen data without allowing it to learn from the input. Validation loss is used to detect overfitting and decide early stopping. Also it is helpful for hyperparameter tuning, such as choosing learning rate, hidden size of LSTM, number of layers, dropout rate, batch size.
Finally, after training, the test set is used for the final evaluation of the trained model, as it provides an unbiased estimate of the model's generalization performance on unseen data.
Another thing to note is that training of an LSTM model is performed using batches B so the full training set is divided into smaller subsets of fixed size. This is necessary as calculating the loss function and gradient over the entire training set would be computationally expensive in terms of both memory and processing time. Furthermore, batching allows for more frequent parameter updates within a single epoch, enabling faster convergence and thus learning. Also, since each batch consists of a different subset of the data, the gradients computed at each step vary accordingly, such variability introduces stochasticity into the optimization and helps prevent overfitting.
Each batch contains a set of SMILES strings that are processed in parallel, therefore, after the forward pass, we compute a batch loss, followed by BPTT and parameter update. This happens for all batches of the dataset. A complete pass over the training set is called an epoch, the training process is repeated for N epochs. Thus the model incrementally refines its parameters through repeated exposure to the dataset. At early epochs the model learns basic SMILES syntax and local dependencies whilst later epochs lead to capturing long range dependencies, reducing prediction error and improving generalization.
The output of this whole pretraining step is a pretrained model that has learnt SMILES syntax, structural constraints, chemical patterns and local and long range

dependencies. Such model is therefore able to generate syntactically plausible molecular strings that reflect the training distribution, of course chemical validity is not guaranteed and must be verified through post-processing. At this stage, however, the model is task-agnostic.

The second phase of training is finetuning the pretrained model to shift the SMILES distribution towards molecules with desired biological properties. For this, as mentioned, we use a small dataset of TRAP1 known actives containing 43 ligands. Finetuning, which follows the same sequential procedure as pretraining, is performed by transfer learning. This means that the weights of the parameters are not reinitialized but the pretrained weights are used as a starting point. During finetuning optimization, the learning rate is small so as to not erase pretrained knowledge. In addition, fewer epochs are needed as the finetuning dataset is often small so we must prevent overfitting. For this reason, careful regularization is required. One technique, employed here, is "early stopping"[?] , which automatically stops training when the model stops improving on the validation set. Generally, in fact, training loss almost always decreases whereas validation loss decreases at first but then may increase if the model starts overfitting on the training data. So if the validation loss does not improve for a predefined number of iterations, known as the patience parameter, then training is stopped: this is the best validation epoch. Its parameters are stored and this final model represents the finetuned model.

With finetuning, the model shifts its learned molecular distribution from a broad drug-like chemical space toward a distribution enriched in molecules associated with TRAP1 activity, as defined by the dataset of known TRAP1 binders.

During finetuning, model evaluation is performed using repeated random subsampling validation, also referred to as Monte Carlo cross-validation.

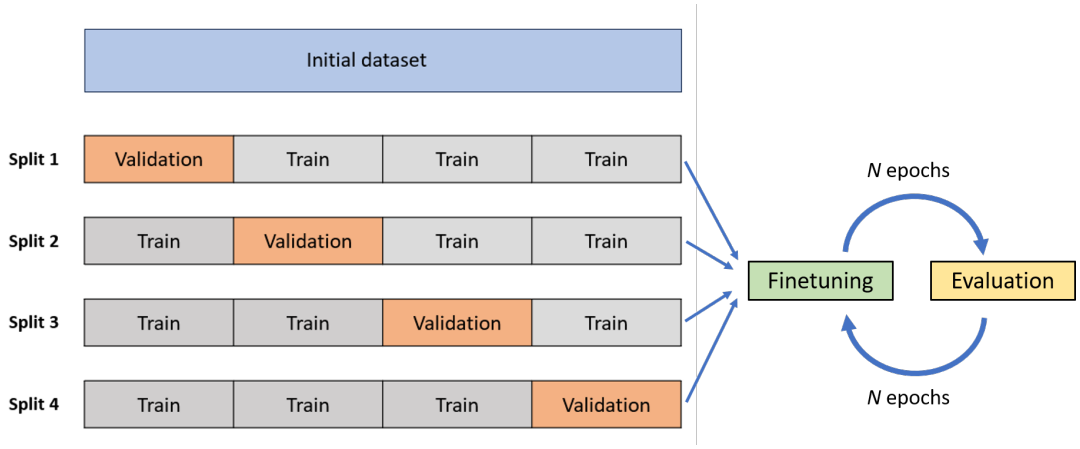


Figure 2.4: Cross-validation training

The finetuning dataset is randomly divided into training (80%) and validation (20%) subsets, such splitting procedure is repeated N times. Therefore, for each split, the model is fine-tuned on the training subset and evaluated on the held-out validation subset.

This procedure is necessary because we have only 43 TRAP1 active compounds so we are in a low-data regime. This means that the model is highly sensitive to the specific composition of the validation subset (high variance). This is even more true for LSTMs, and more in general RNNs, as they are high capacity models that comprise a large number of learnable parameters. Therefore, the evaluation on multiple random splits of the dataset provides a more robust estimate of model generalization. Also, repeated splitting enables the detection of performance variability and potential overfitting.

Furthermore, to address the limited size of the TRAP1 active dataset and improve model generalization, data augmentation[28;29] was employed through randomized enumeration of molecular representations. This means that a single molecule is encoded by multiple valid SMILES strings. In this project, after dividing the dataset into training and validation, for each compound, ten isomeric SMILES strings are generated via random atom ordering.

This strategy is particularly important for recurrent neural network architectures, such as LSTMs, which operate on sequential token inputs and are therefore sensitive to input ordering. Without augmentation, the model may overfit to specific SMILES patterns rather than learning chemically meaningful features. By introducing randomized representations, the model is able to generalize across different valid encodings of the same molecule.

The finetuned model of the best epoch, so associated with the lowest validation loss, is final model that is subsequently used as a generative tool to generate TRAP1 specific SMILES. Generation means sampling a sequence, token by token, from the finetuned model's probability distribution. Instead of predicting the correct next token, as in training, we sample one. The first hidden state h_t and cell state c_0 is initialized with the special token <SOS> and a probability vector over the vocabulary is computed. A token is sampled from this distribution and becomes the next input. This is repeated iteratively; therefore, at each timestep, the probability distribution is computed, the next token is sampled and appended to the sequence, and hidden and cell states are updated. Such process stops when the <EOS> token is generated or when the maximum length is reached.

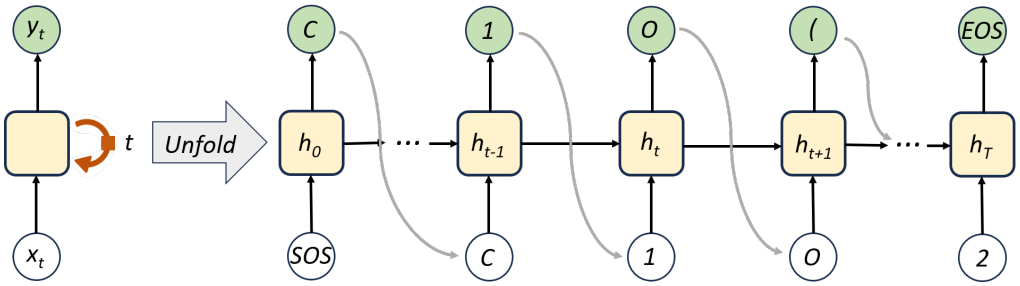


Figure 2.5: RNN generation unfolded

The sampling is probabilistic or stochastic: a token is sampled randomly but according to the probability distribution predicted by the model. This allows to explore a wider chemical space enabling a higher diversity of generated compounds and the discovery of novel scaffolds. The main disadvantages are: less stable output (different molecules at each run which can hinder reproducibility and make benchmarking harder) and chemical invalidity of a significant fraction of generated sequences. In reality, this second issue is dealt with, after generation, by downstream filtering and validation to refine the generated set.
If instead, at each timestep we choose the most likely token (so the one with the highest probability), thus applying deterministic sampling, we would get high syntactic stability and reproducibility, but poor exploration of chemical space and many generated molecules become structurally similar.
After the generation of a sequence, the per token log probabilities are computed:

$$logP_\theta(x_t \mid x_{<t}) \tag{2.27}$$

and then summed over the sequence:

$$logP_\theta(x) = \sum_{t=1}^{T} logP_\theta(x_t \mid x_{<t}) \tag{2.28}$$

This is the log-likelihood of the generated molecule and represents how confident the model is in generating the sequence. In most cases, as here, the log-likelihood is normalized by sequence length so as to have a fair comparison across molecules. Otherwise with raw log-likelihood, longer sequences would be penalized as these have a more negative value since we are summing more negative log-probabilities.
A high log-likelihood means that the molecule closely aligns with the distribution learnt during training, whilst a low log-likelihood indicates a weak overlap with the training distribution space. It is important, therefore, to consider that high-likelihood does not reflect true chemical validity or biological activity of a molecule,

analogously low-likelihood compounds can also display the desired molecular properties . This means that downstream evaluation methods are necessary when assessing molecules for target-specific applications such as TRAP1 binding.
Initial post-processing entails the removal of invalid sequences, chemical sanitization and deduplication. This is followed by physicochemical and structural analysis, such as evaluation of drug-likeness criteria (Lipinski's Rule of Five), ADMET prediction... Finally, target-specific evaluation is carried out, such as molecular docking against the target, pharmacophore matching and activity prediction models, in order to predict binding affinity and thus specific biological activity. It is also important to assess synthetic accessibility to ensure that candidate molecules can effectively be produced, as ultimately, experimental validation is the decisive step that converts computationally generated candidates into viable therapeutic leads.
To assess the performance of generative models, Polykovskiy, et al.[30] proposed a set of evaluation metrics. These help identify common problems, such as overfitting, imbalance of frequent structures, and model collapse.
Firstly validity of generated SMILES strings is evaluated, usually, using RDKit's molecular structure parser that checks atom's valency, bonding and correct ring closures.
Validity is computed as:

$$Validity = \frac{Number\ of\ valid\ molecules}{Total\ generated\ molecules} \tag{2.29}$$

Then uniqueness is assessed which measures how many molecules are distinct from each other:

$$Uniqueness = \frac{Number\ of\ unique\ molecules}{Number\ of\ valid\ molecules} \tag{2.30}$$

This captures if there is redundancy in generation: mode collapse.
Finally, novelty is measured, which indicates how many generated molecules are not present in the training dataset:

$$Novelty = \frac{Number\ of\ novel\ molecules}{Number\ of\ unique\ molecules} \tag{2.31}$$

This represents the ability of the model to generate new chemistry without memorizing training data.
The evaluation of molecular generative models is highly sensitive to experimental parameters, particularly the number of generated samples[31]. Small sample size produces unstable estimates and high sensitivity to randomness, and can overestimate performance. Typically large sample sizes are employed, in the order of tens to hun-

dreds of thousands of molecules, both for benchmarking frameworks[30;32] and drug design[33–35]. Whilst for benchmarking, large sample size is important to ensure fair comparison metrics and so reproducible evaluation of generative models, for drug design purposes this is necessary, more than anything, to retain a sufficient number of viable candidates after downstream filtering.

2.2 Molecular Docking

As stated above, one of the target-specific post processing methods employed is molecular docking. This is a computational method used to predict the most likely binding pose in a target macromolecule, usually a binding site of a protein. Docking involves two main components: first, the search of all possible orientations and conformations of a ligand in the target pocket, known as poses, and second, the approximate evaluation of binding affinity of poses with a scoring function. There are two main categories of search algorithms: systematic search and stochastic search algorithms. Systematic methods try to exhaustively explore the conformational space in a deterministic way. The ligand is broken into rigid fragments and rotatable bonds, these bonds are then sampled at discrete intervals and afterwards the ligand is placed into the binding site in many orientations. This approach, however, is computationally expensive as computation scales exponentially with flexibility. To avoid the problem of combinatorial explosion, what is usually performed is an incremental growth of the ligand within the binding site. One strategy is to initially dock the rigid core fragments of the ligand and afterwards add the more flexible side chains[36 37].

Stochastic methods, instead, sample orientations in a random way that is however guided by probability. The starting point is an initial ligand pose, to which a random perturbation, such as a rotation, translation or torsional change, is applied. This is evaluated with a scoring function and the move is accepted or rejected based on rules. This way, over time, the system moves through the conformational space finding low energy binding modes. One common algorithm is the Monte Carlo algorithm[?] . Here the random move is accepted if the energy of the corresponding pose is lower, although occasional "worse" moves are accepted, for example according to a Metropolis criterion, so the system can escape local minima.

The second stage of docking is evaluation of the generated poses through a scoring function. Scoring functions can be classified as force field-based, empirical or knowledge-based. Force field-based scoring functions are physics-based methods that therefore employ molecular mechanics to estimate interactions between ligand

and protein. Interaction terms include electrostatics (Coulombic terms) and Van der Waals interactions, and on a smaller scale, solvation and entropic effects. Empirical scoring functions are fitted models where the score is a weighted sum of interactions, with the weights derived from experimental data. Finally, for knowledge-based scoring functions, the aim is to reproduce experimental structures instead of binding energies, in contrast to empirical scoring. For this, large databases of known protein-ligand structures are analyzed and the frequencies of atom-atom interactions are converted into potentials using statistical mechanics.

In this project, the Glide software, in Schrödinger-Maestro, was used. This uses a hybrid approach. Firstly, a 3D grid is precomputed around the binding site, where each grid point stores information about the shape of the protein and the interaction potential. This grid-based approach eliminates the need to recompute protein–ligand interactions during docking, significantly improving computational speed. Ligand conformations are also precomputed. Subsequently, hierarchical search strategy is employed, where ligand conformations are placed in the binding site based on geometric and chemical complementarity, generating a large number of candidate poses. These poses are then subjected to successive filtering steps to eliminate sterically unfavorable or poorly fitting conformations. Surviving poses undergo further optimization, followed by local energy minimization (physics-based). Finally, poses are ranked using GlideScore, an empirical scoring function that approximates binding affinity as:

$$\begin{aligned} Glide\ score =& 0.05\ \cdot VdW + 0.15\ \cdot Coulomb + Lipo + Hbond \\ &+ Metal + Rewards + RotB + Site \end{aligned} \tag{2.32}$$

The van der Waals (VdW) term accounts for steric complementarity and dispersion interactions between ligand and receptor atoms, while the Coulomb term describes electrostatic interactions. Additional contributions include a lipophilic (Lipo) term, which rewards hydrophobic contacts, and a hydrogen-bonding (Hbond) term, which evaluates the geometry and strength of hydrogen bond interactions. The Metal term accounts for coordination interactions between the ligand and metal ions in the binding site. The scoring function also includes empirical corrections, such as the Rewards term, which captures favorable interaction patterns, a RotB penalty that accounts for the entropic cost of ligand flexibility by penalizing rotatable bonds, and a Site term that incorporates binding-site-specific effects such as desolvation. The units of GlideScore are kcal/mol.

Prior to docking, the TRAP1 receptors were prepared employing the Protein Prepa-

ration Wizard workflow, in Maestro, leaving the default settings. All ligands were prepared with the built-in Maestro module LigPrep employing the OPLS4 force field. For each ligand, tautomeric and ionization states at pH 7.0 ± 2.0 were generated, also for chiral compounds in the dataset, chirality was retained. Isomers were then ranked and filtered before docking; this is detailed further in the next chapter (Results). The grids, needed for pose prediction and scoring, were generated through Receptor Grid Generation in Glide keeping default settings. The box was centered on the ATP molecule in the NTD of TRAP1 and then expanded from 0.10 $\AA$ to 0.12 $\AA$ in all directions. The docking calculation was carried out with Ligand Docking workflow employing SP (Standard precision) and a maximum of five poses were kept for each ligand.

2.3 Molecular Dynamics

Molecular Dynamics (MD) is a computational method used for studying the spatial evolution of atoms and molecules over time. MD simulates a system of N particles by solving Newton's equations of motion for all atoms. The force (F_i) applied on an atom i by the rest of the system is expressed as the negative gradient of the potential energy function $V(r)$ with respect to the position of atom i.
The equation of motion can be written as:

$$F_i = -\frac{\partial V}{\partial r_i} \tag{2.33}$$

The motion of each atom in three-dimensional space is governed by Newton's second law:

$$\frac{d^2 r_i(t)}{dt^2} = \frac{F_i}{m_i} \tag{2.34}$$

By numerical integration of these equations, over successive small time steps δt, MD simulations generate trajectories describing the evolution of atomic positions and velocities. The potential energy of the system, which defines the atomic interactions, is approximated by a set of parameterized equations, known as a force field. This is a mathematical function, generally expressed as the sum of bonded and non-bonded interactions:

$$E_{total} = \sum E_{bonded} + \sum E_{non\ bonded} \tag{2.35}$$

The force field used in this work belongs to the AMBER family. Here, the bonded term comprises 4 types of interactions: bond stretching, angle bending, dihedral torsions and improper dihedrals; this is expressed as:

$$E_{bonded} = \sum_{bonded} K_b(b - b_0)^2 + \sum_{angles} K_\theta(\theta - \theta_0)^2 \sum_{\substack{improper \\ dihedrals}} K_\varphi(\varphi - \varphi_0)^2 + \\ + \sum_{dihedrals} K_{\phi,n}[1 + cos(n\phi - \delta_n)] \tag{2.36}$$

The non-bonded term, instead, comprises electrostatic and van der Waals forces:

$$E_{non\ bonded} = \sum_{\substack{non\ bonded \\ pairs\ i\ j}} \kappa_e \frac{q_i q_j}{\varepsilon_r r_{ij}} + \sum_{\substack{non\ bonded \\ pairs\ i\ j}} \varepsilon_{ij} \left[\left(\frac{R_{min,ij}}{r_{ij}} \right)^{12} - 2 \left(\frac{R_{min,ij}}{r_{ij}} \right)^{6} \right] \tag{2.37}$$

Furthermore, MD simulations generally employ periodic boundary conditions (PBC), where the simulation box is replicated infinitely in all directions. PBC removes boundary effects and allows the system to behave as if it were part of a bulk material. Without this, atoms at the surface of the simulation box would experience artificial forces due to missing neighbors, leading to nonphysical behavior.
After this general description of the theoretical principles underlying molecular dynamics simulations, the specific MD protocol applied in this work is described below. The crystal structure of human TRAP1 was retrieved from the Protein Data Bank (PDB:6XG6). The structure includes 2 ADP:BeF_3 residues and 2 Mg^{2+} ions in the active sites. ADP:BeF_3 was manually converted to ATP after careful RMSD-alignment of the active sites with those of a crystal structure of zTRAP1 previously prepared in the group from PDB code 4IPE. Similarly, we kept the Mg^{2+} ions present in zTRAP1 structure for the final structural model. The unresolved loops within the structure were reconstructed using the Crosslink Protein workflow in Maestro suite with Prime Energy in implicit solvent. Such loops included residues 282 to 292 and 486 to 502 in Protomer A, and residues 290 to 292, 489 to 503, and 558 to 561 in Protomer B. The protein was then prepared using the Protein Preparation Wizard in Maestro. This comprises assigning the correct protonation states of the protein at pH 7.0, adding missing hydrogen atoms and optimization of hydrogen-bonding network. Termini were modeled with charged NH_3^+ and COO^- caps. Finally, the system was briefly minimized in OPLS4 force field to relieve steric clashes and optimize the geometry.

The whole system must now be prepared for the MD simulation. Protein residues were parametrized using the Amber ff14SB force field. ATP parameters were taken from Meagher et al.[38], while Mg^{2+} parameters were adopted from Allnér et al.[39] The system was solvated using tleap (AmberTools21) with a truncated octahedral box of TIP3P water molecules, maintaining a minimum solute–box distance of 14.5 Å. The system was neutralized by adding Na^+ ions, described using parameters from previous published work[40].

Molecular dynamics simulations were performed using the *sander* and *pmemd* engines from AmberTools (version 21) and Amber (version 20), respectively. Sander was employed for the initial minimization, solvent equilibration, and heating phases, whereas the GPU-accelerated pmemd.cuda was used for the subsequent equilibration and production runs.

Before the production run, energy minimizations were performed as well as equilibration steps in order to achieve the correct temperature and pressure states.

The systems was minimised with a two-stage approach: in the first stage, a 1000-step minimisation was carried out on all hydrogens and waters using 500 steps of steepest descent algorithm, followed by 500 steps of conjugate gradient method. Positional restraints were introduced on Mg^{2+} ions and all heavy atoms of protein residues and ATPs, with a harmonic force constant equal to 500 mol^{-1} $\mathring{A}^{-2}$. In the second stage, a 2500-step minimisation was performed on the whole system without positional restraints and switching from steepest descent to conjugate gradient after 1000 steps. The cutoff for non-bonded interaction was 10 Å in both minimisation steps, meaning that after this limit only Coulomb interactions were computed with the Particle Mesh Ewald method[41]. Solvent equilibration was performed with a 9 ps simulation in the NVT ensemble (time step, 1 fs), assigning positional restraints (force constant of 10 mol^{-1} $\mathring{A}^{-2}$) to all solute atoms. This process is divided into three steps: a first 3 ps heating from 25 to 400 K, followed by a 3 ps equilibration at the constant temperature of 400 K and a third cooling to 25 K in the remaining 3 ps. Temperatures are maintained by the Berendsen thermostat[42]. A tight temperature coupling (0.2 ps) is set for the first 6 ps of the process and then relaxed during the 3 ps cooling phase (from 2.0 to 1.0 ps). The cutoff for non-bonded interactions was set to 8 Å and will remain so for the rest of the simulation. A heating step followed, consisting of a system temperature increase from 25 to 300 K in the NVT ensemble during 20 ps. Starting from this stage, the time step was increased to 2 fs in line with the introduction of the SHAKE algorithm for constraining hydrogen-containing bonds[43]. Temperature control was enforced via the Langevin thermostat from now on (collision frequency set at 0.75 ps^{-1} for this stage), whereas harmonic positional

restraints were applied to C atoms with a force constant of 5 mol^{-1} $\mathring{A}^{-2}$ to avoid large fluctuations. Next, the system was equilibrated in the NpT ensemble at 300 K with a 4-step protocol. All equilibration steps entailed an NpT ensemble, 300 K temperature enforced by the Langevin thermostat, 1 atm pressure (controlled by the Berendsen barostat, relaxation time of 1 ps) and the SHAKE algorithm. The whole process lasted 4.0 ns (time step, 2 fs). The first 20 ps of equilibration was performed by relaxing position restraints on backbone Cα to 3.75 mol^{-1} $\mathring{A}^{-2}$. The second step further relaxed the position restraints to 1.75 mol^{-1} $\mathring{A}^{-2}$ during 20 ps of simulations. The third step was 2 ns long with no backbone restraints active. In these first three relaxation steps the collision frequency of the thermostat was 1 ps^{-1}. In the last step of 2 ns all the conditions were kept except for the coupling to the thermostat which featured a collision frequency of 5.0 ps^{-1}. All simulations, from heating cycles to system equilibration, employ PBC conditions.
Four independent replicas of 1 μs were carried out using the same conditions of the fourth NPT equilibration at 300K and PBC conditions. System coordinates were stored every 50 ps, yielding 20,000 frames per replica.

Chapter 3

Results and Discussion

3.1 LSTM: training, finetuning and generation

The first stage of this thesis work is to develop a generative tool for TRAP1 binders (de novo drug design); the workflow for such process is illustrated below:

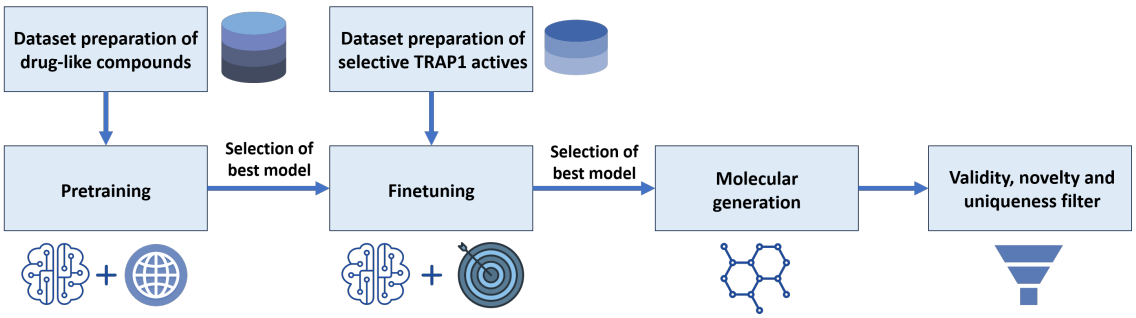
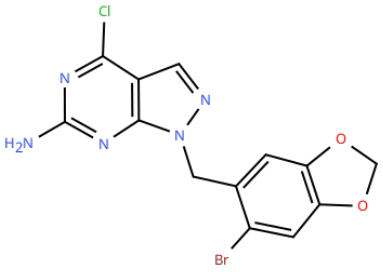


Figure 3.1: LSTM workflow

The chosen architecture is an RNN-LSTM network which is able to generate molecular strings, token by token. The adoption of this framework is viable due to the possibility of representing chemical structures as linear sequences, usually by SMILES encoding. LSTMs are well suited for de novo drug design for many reasons, such as: learning of implicit chemical syntax, capturing of long range dependencies and generation of novel molecules.
In short, an LSTM network is trained to learn the probability distribution over valid SMILES strings by predicting the next token in a sequence given its preceding context. Once trained, the LSTM can be used as a generative model by sequentially sampling tokens to produce new SMILES strings. Generation can be guided towards desired properties by finetuning the trained model on a smaller set of molecules with a specific biological activity.

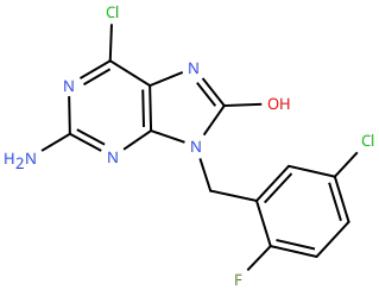
The development of the network, therefore, follows two main stages, firstly a pretraining step where the model learns to generate chemically valid SMILES, and secondly a finetuning step where the generation is shifted towards TRAP1 binders. The pretraining dataset, comprising drug-like compounds, was extracted from the ChEMBL v33 database[44], as described in Özçelik et al[31]. The first step involved the selection of compounds containing only: C, H, O, N, P, F, Cl, Br and I. The following steps included removal of salts, neutralization, sanitization, canonicalization and removal of stereochemistry annotations. The selected list of SMILES was randomly split into training (1.500.000), validation (40.000) and test (44.858) sets. The canonical SMILES of training and validation set were converted into isomeric representations prior to training. The motivation behind this lies in the data augmentation strategy employed in the subsequent fine-tuning step.
For the selection of the finetuning dataset, we used both ChEMBL and BindingDB[45]. The aim here is to build a small dataset of actives that are selective for TRAP1, specifically NTD binders, therefore, orthosteric ligands. Compounds were extracted from the databases by IC_{50}, K_i and K_d values. For all of these, specificity for TRAP1 was checked by verifying that retained compounds experimentally show specificity towards TRAP1 with respect to at least one of its isoforms (GRP94, HSP90α or HSP90β). Two of the compounds, CHEMBL4641095 and CHEMBL4128206, contain a triphenylphosphonium (TPP) moiety with an aliphatic carbon spacer. This moiety is primarily introduced to promote mitochondrial accumulation rather than to directly interact with the TRAP1 ATP-binding site. Such portion was therefore removed and and replaced with smaller capping groups. For CHEMBL4641095, we created three different compounds by installing in the first one, an amino -NH group, in the second one, a methyl-amino group -NMe and in the third one, an ethyl-amino group -NEt; the three ligands are labelled as CHEMBL4641095_NH, CHEMBL4641095_Me and
CHEMBL4641095_Et respectively. For CHEMBL4128206 we built two compounds, one with a methyl and the other with an ethyl group, resulting in CHEMBL4128206_Me and CHEMBL4128206_Et. For two compounds, ZINC908175388 and ZINC237713788, we can write two stereo-isomers, both of these, experimentally showed TRAP1 specificity, therefore, this must be considered. Since stereochemical information was removed for fine-tuning, each stereoisomeric pair collapsed to the same achiral representation; however, both stereoisomers were retained for the subsequent docking analysis (stereochemistry impacts the orientation of the ligand in the binding site).

The final finetuning dataset contains 43 ligands:



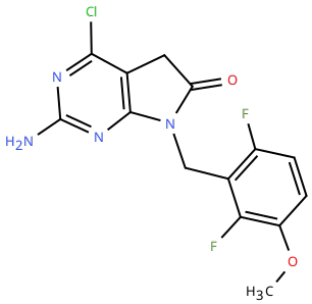
act01

CHEMBL4068596



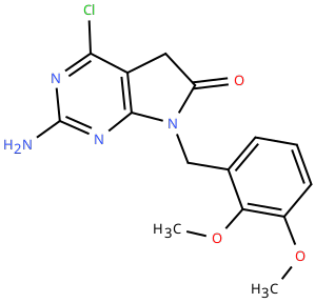
act02

CHEMBL4846028



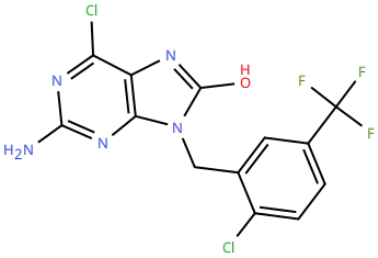
act03

CHEMBL4850126



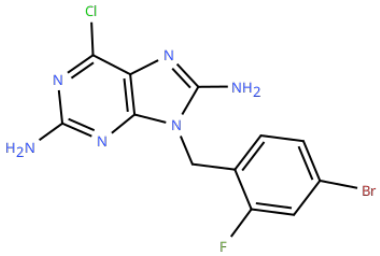
act04

CHEMBL4854861



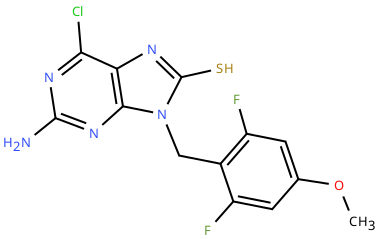
act05

CHEMBL4857569



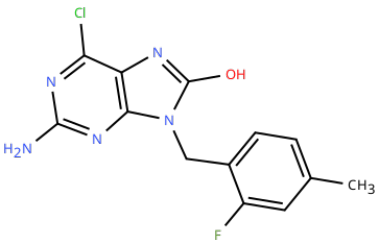
act06

CHEMBL4859919



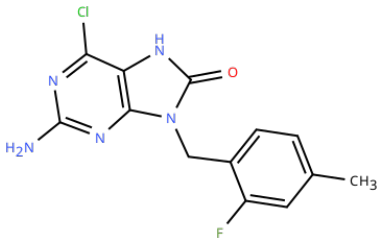
act07

CHEMBL4860652



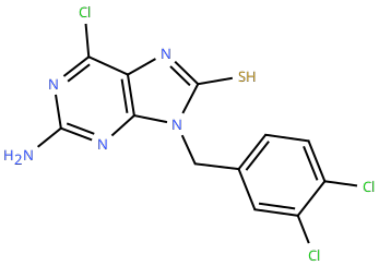
act08

CHEMBL4863672



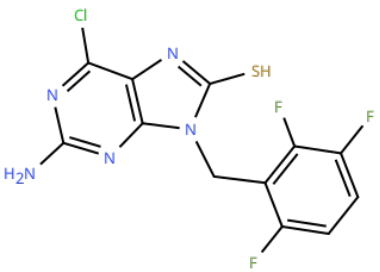
act09

CHEMBL4863988



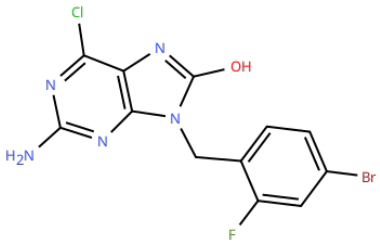
act10

CHEMBL4864922



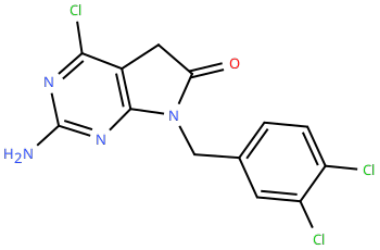
act11

CHEMBL4865186



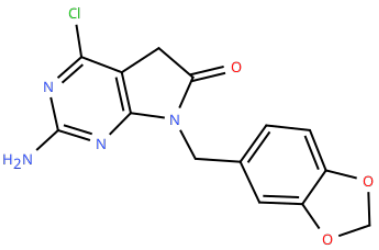
act12

CHEMBL4866705



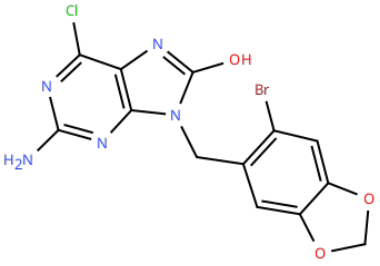
act13

CHEMBL4866958



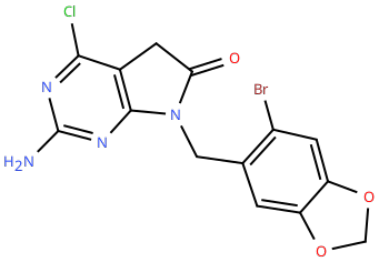
act14

CHEMBL4867384



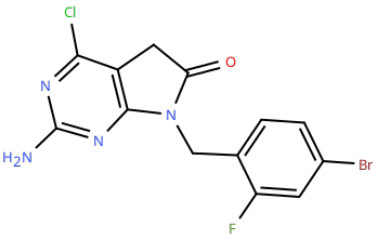
act15

CHEMBL4869481



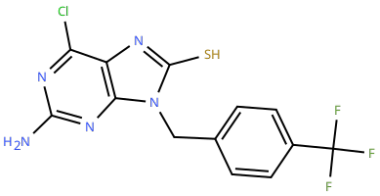
act16

CHEMBL4870087



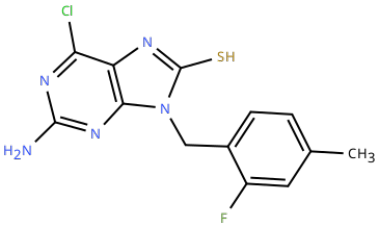
act17

CHEMBL4871630



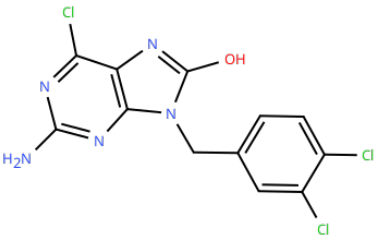
act18

CHEMBL4872173



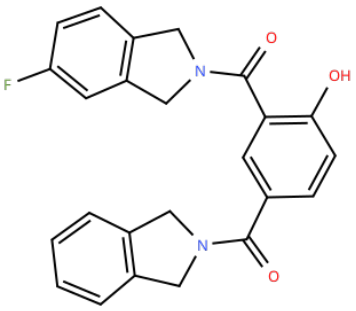
act19

CHEMBL4876871



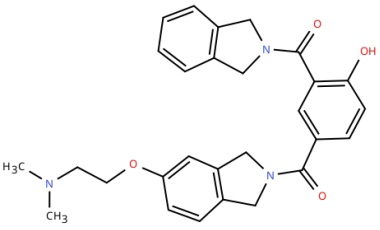
act20

CHEMBL4879021



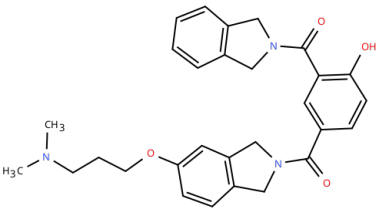
act21

CHEMBL5629930



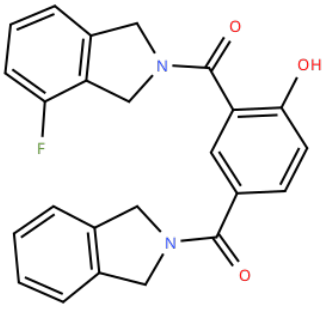
act22

CHEMBL5631423



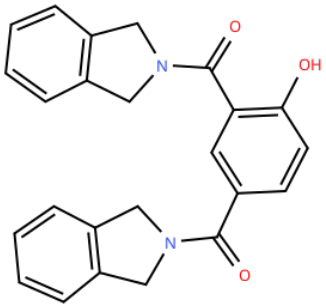
act23

CHEMBL5631613



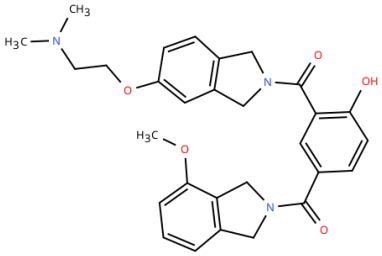
act24

CHEMBL5631707



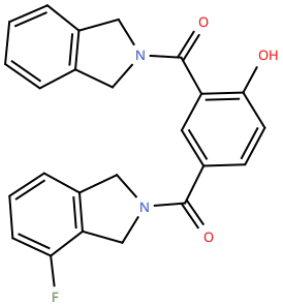
act25

CHEMBL5631756



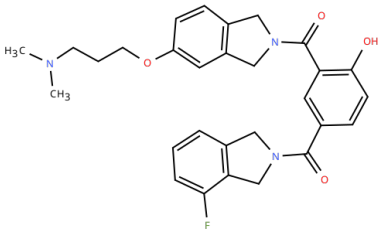
act26

CHEMBL5632075



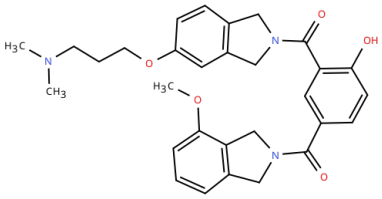
act27

CHEMBL5632194



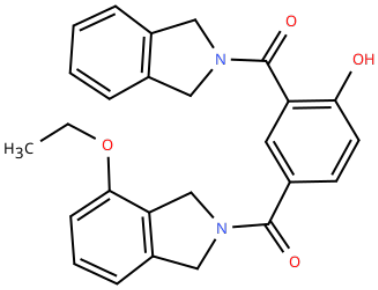
act28

CHEMBL5632244



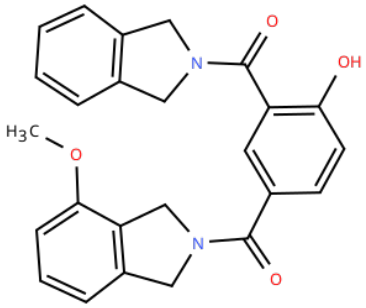
act29

CHEMBL5633012



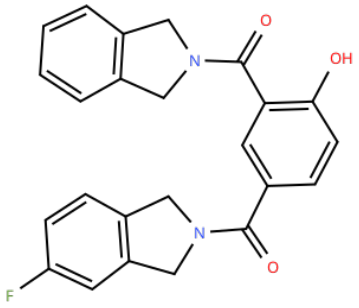
act30

CHEMBL5633571



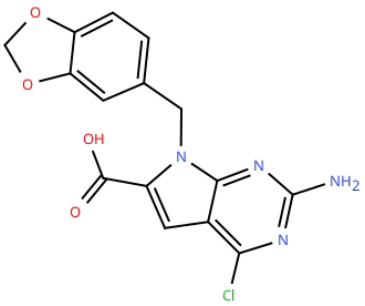
act31

CHEMBL5633903



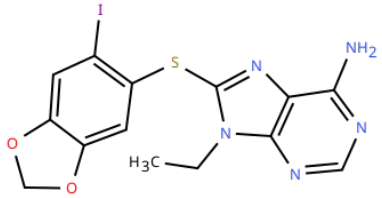
act32

CHEMBL5633971



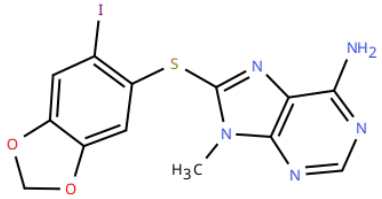
act33

CHEMBL4079410



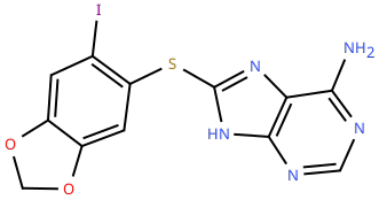
act34

CHEMBL4641095_Et



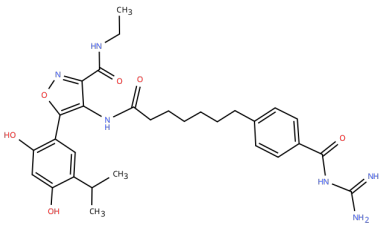
act35

CHEMBL4641095_Me



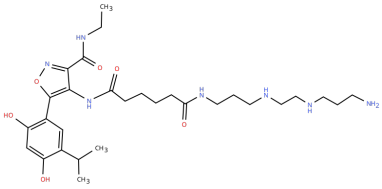
act36

CHEMBL4641095_NH



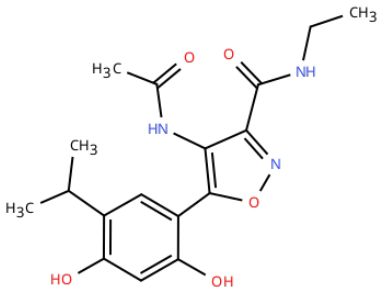
act37

CHEMBL4128056



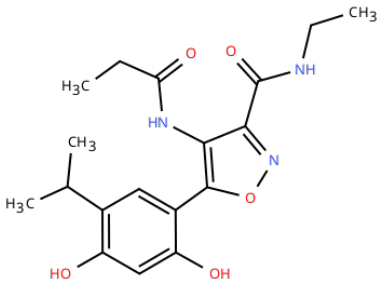
act38

CHEMBL4129436



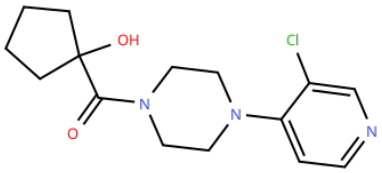
act39

CHEMBL4128206_Me



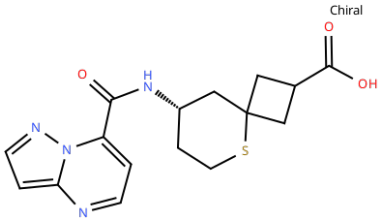
act40

CHEMBL4128206_Et



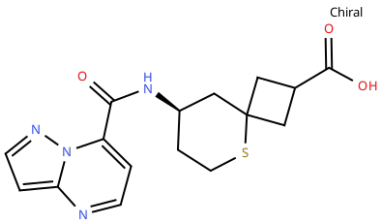
act41

ZINC000095438387



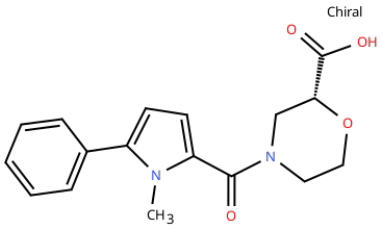
act42/act42_1

ZINC908175388



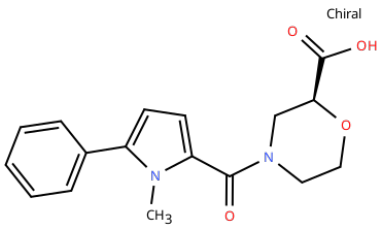
act42_2

ZINC908175389



act43/act43_1

ZINC237713383



act43_2

ZINC237713788

where the ligands act42 and act43 belong to the finetuning set whilst compounds act42_1, act42_2, act43_1 and act43_2 are used in the later docking stage.
Experimental data of selected compounds can be found in the following works: act01-act20[46] , act21-act32[47], act33[48], act34-act36[49], act37-act40[11].

For pretraining, a hyper parameter search was carried out. In particular, 100 random combinations of number of layers, model dimension, dropout, learning rate and batch size were tested. For each combination, training entailed three steps: a forward pass that computes predictions and losses, a backward pass that computes gradients and finally a parameter update step; this whole process is carried out on the training set. Instead, for the validation set, we only have a forward pass, with the computation of the validation loss, so no BPPT or weight update takes place. The model that yields the lowest validation loss, and which will represent our pretrained model, utilizes the following parameter values:

- number of layers: 6
- model dimension: 256
- dropout: 0.25
- vocabulary size: 34
- sequence length: 91
- learning rate: 0.0001
- max number of epochs: 1000
- batch size: 8192

The optimizer used for weight update is the Adam optimizer, which is stochastic gradient-based. Also, as previously mentioned, during training, we use a technique called "Early Stopping", which automatically stops training when the model stops improving on the validation set. For finetuning, model performance was evaluated using repeated random subsampling validation (Monte Carlo cross-validation), this is necessary to lower variance and improve generalization in this low-data regime. The finetuning dataset (which, as previously detailed, is a small set of TRAP1 actives) was randomly divided into training (80%) and validation (20%) subsets, and the splitting procedure was repeated four times. Therefore, finetuning was carried out independently on each split for N epochs. Within each split, after dividing the dataset into training and validation, SMILES were enumerated so each molecule was represented by multiple non-canonical SMILES strings generated via random atom ordering. In particular, for each compound, ten isomeric SMILES strings are generated. This procedure improves robustness for limited datasets as the model, by learning representation-invariant features, is able to generalize across different valid encodings of the same molecular structure.

As described in the earlier Chapter, finetuning, and more in general training, is stopped when the validation loss, computed on the validation dataset, starts increasing.
Therefore to choose the best finetuned model we inspected the validation loss of the last epoch of each split:

- Split 1: 44 epochs - validation loss 0.748
- Split 2: 36 epochs - validation loss 0.689
- Split 3: 42 epochs - validation loss 0.658
- Split 4: 40 epochs - validation loss 0.704

The chosen finetuned model that will be used for further molecular generation is consequently the model trained on Split 3.
Next we generated 100,000 SMILES and retained only strings that were valid, unique and novel which resulted in 62,722 molecules in total. This first result indicates that the model is capable of generating valid and diverse molecules.

3.2 Postprocessing and receptor ensemble preparation

The pipeline of this next stage can be summarized as:

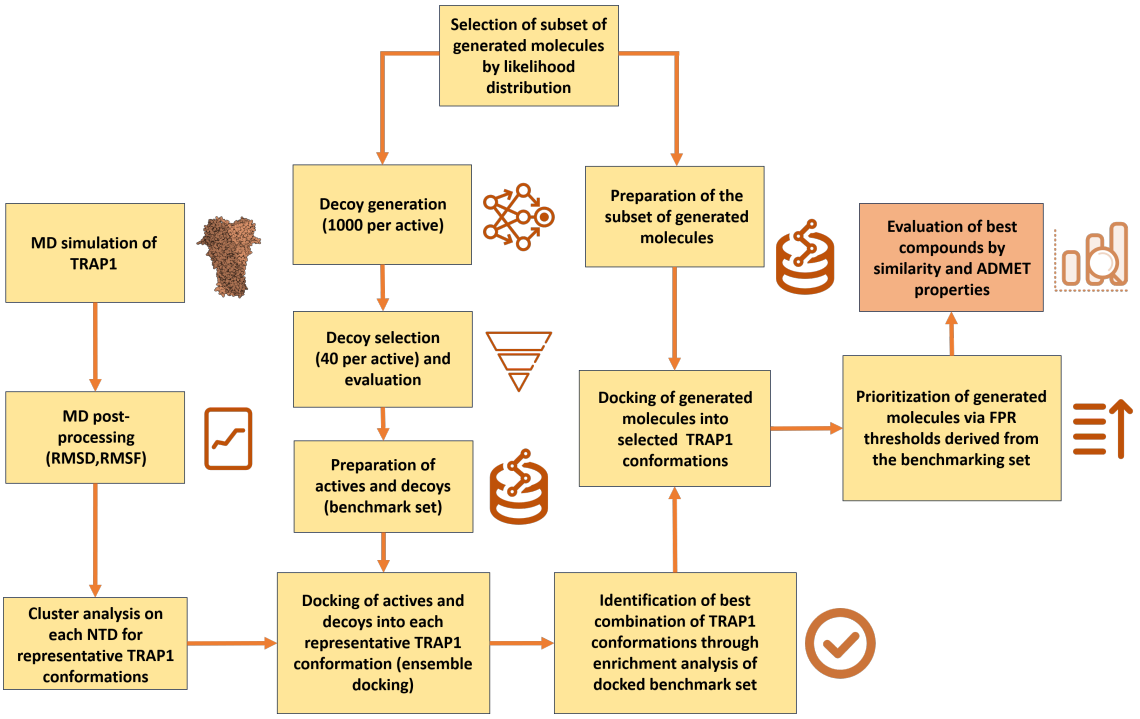


Figure 3.2: MD and ensemble docking workflow

3.2.1 Filtering of generated molecules

We must now proceed in assessing the biological relevance and drug-likeness of the generated compounds. To start with, we do not carry all the 62,722 molecules through to post-processing but we apply a likelihood-based filtering step. The LSTM network assigns to each generated molecule a log-likelihood which measures how probable a sequence is according to the learned model distribution. The distribution of such likelihood was analyzed and the subset corresponding to the upper tail (those exceeding $\mu + 1\sigma$) was selected for subsequent post-processing.

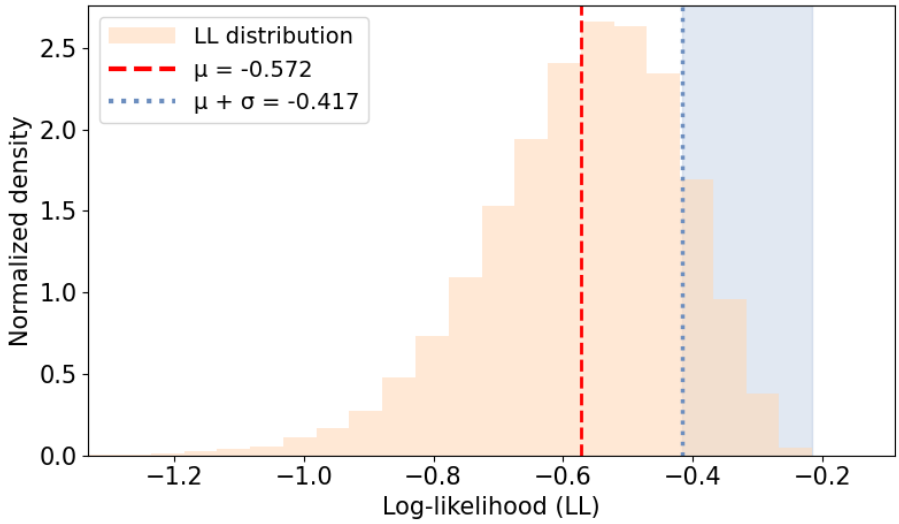


Figure 3.3: Log-likelihood distribution of valid LSTM molecules

First of all, due to high likelihood, the selected compounds can be considered more chemically plausible and structurally coherent. Although all retained SMILES strings from the validity, uniqueness and novelty filter are formally valid, generative models may still produce atypical or less realistic structures due to stochastic sampling. This also means though that these molecules are more consistent with the distribution learned by the fine-tuned model. This likelihood filtering therefore prioritizes high-confidence candidates at the expense of broader chemical novelty.
The final selected LSTM set comprises 9482 SMILES.
To evaluate such molecules, our approach is to carry out ensemble molecular docking on the most representative TRAP1 conformations extracted from molecular dynamic (MD) trajectories. This structure-based strategy is necessary as the likelihood of the generative model is not a measurement of TRAP1 binding but rather indicates how probable a molecule is according to the fine-tuned model. It is essential, therefore, to employ an independent filter based on the target structure to verify if the generated molecules are compatible with the TRAP1 ATP site, pose and interactions. Furthermore, since proteins are intrinsically dynamic, it makes sense to use ensemble docking on representative conformations from MD rather than a single static structure.

3.2.2 Molecular dynamics simulations of ATP-bound TRAP1

Molecular dynamics simulations were performed using Amber pmemd.CUDA with the ff14SB force field under periodic boundary conditions. Before undergoing the simulation, the TRAP1 system was prepared using the Maestro Protein Preparation Wizard. This entails addition of missing hydrogen atoms, assignment of correct protonation states of the protein at physiological pH, optimization of the hydrogen-bonding network and a restrained energy minimization step which is needed to remove steric clashes. Next the system was solvated in a truncated octahedral box of TIP3P water molecules and neutralized by the addition of Na^+. Simulation box dimensions were checked in order to prevent artificial self interactions. The system was then equilibrated through NVT and NPT equilibration steps. Temperature was controlled at 300 K using a Langevin thermostat, while pressure was maintained at 1 bar using a Berendsen barostat. Equilibration and minimization steps are further detailed in the Materials and Methods Chapter. The MD simulation was carried out in four independent replicas of 1000 ns (1 μs) each. It is important to highlight that TRAP1 was simulated in its ATP-bound state, with one ATP molecule and one Mg^{2+} ion bound to each N-terminal domain.
The first MD post-processing step was an RMSD calculation on the backbone of the protein to examine structural stabilization and absence of evident drifts.

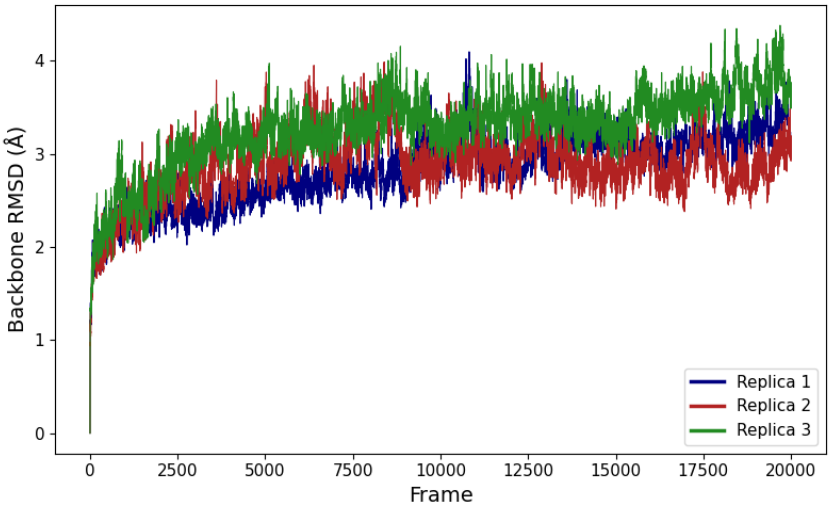


Figure 3.4: RMSD of first three replicas on the backbone of the protein

This was performed on the meta-trajectory and on each replica, first on the whole protein and then separately for each NTD.

The same procedure is carried out for RMSF computation:

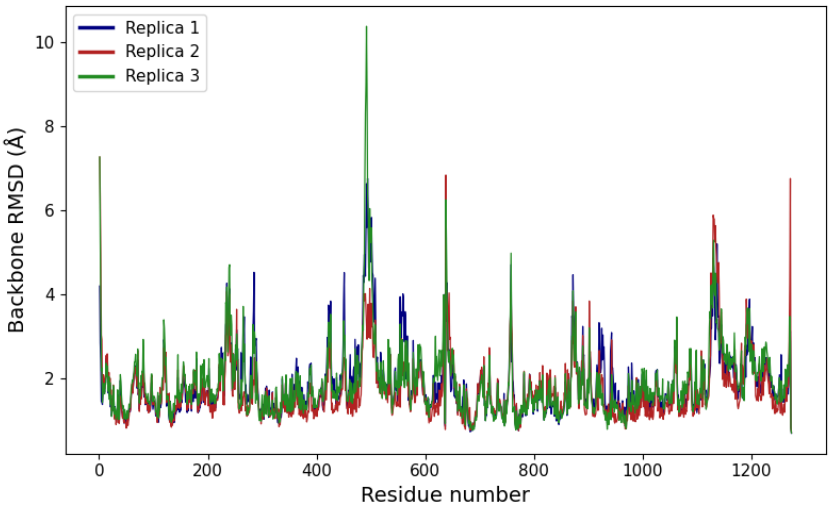


Figure 3.5: RMSF of first three replicas on the backbone of the protein

Here we noticed that replica 4 showed significant fluctuations, after visualising this trajectory with the VMD software, and considering its RMSD instability, we decided to remove such replica. For the other three replicas, we observed that the most fluctuating residues were mainly located in solvent-exposed loop regions, consistent with their expected structural flexibility. Overall, therefore, protein trajectories showed stable structural behavior after the initial equilibration phase. Based on the RMSD profiles, the first 7000 frames, corresponding to approximately 0.35 ns, were discarded as equilibration. In summary, for further analysis, we kept three replicas of 13.000 frames each.

Binding-site clustering and receptor ensemble selection

The next step was cluster analysis to determine the most populated protein conformations. Since the representative conformations, obtained from clustering, will be used for docking ligands into the orthosteric binding site of TRAP1, clustering was focused on the local NTD binding-site environment rather than on global protein motions. Here, local rearrangements of side chains influence docking as they can affect both steric complementarity and the formation of key interactions. Trajectory frames were aligned using the selected binding-site residues within 5 Å of ATP in the reference ATP-bound structure, and clustering was performed based on the RMSD of their side-chain heavy atoms. The initial selected amino acids within this distance were 19. An additional asparagine residue, Asn171, located in the ATP-lid region and reported to contribute to TRAP1 paralog selectivity through hydrogen

bonding[48], was also included. Consequently, the final 20 selected residues for RMSD calculation and clustering were:

119Asn, 123Ala, 126Lys, 158Asp, 161Ile, 163Met, 171Asn, 172Leu, 178Ser, 179Gly, 180Ser, 199Gly, 200Gln, 201Phe, 202Gly, 203Val, 204Gly, 205Phe, 251Thr, 402Arg

Following alignment, hierarchical agglomerative clustering was performed; this groups conformations based on pairwise structural similarity (RMSD). Although this method does not require fixing the number of clusters a priori, the final cluster partition depends on the selected distance cutoff. Alignment and clustering were performed separately for each replica to reduce the influence of inter-replica differences and focus on local binding-site rearrangements. Alignment and clustering were performed separately on the two N-terminal domains, labeled Site 1 and Site 2, each on the same subset of twenty residues. For each replica and site, different RMSD distance cutoffs were tested to identify a suitable separation of binding-site conformations. Then, for each replica and binding site, representative structures were extracted from the most populated clusters. These representatives were then superimposed and compared across the two sites, with particular attention to the orientation of the binding-site side chains. Redundant conformations showing highly similar side-chain arrangements were discarded, whereas representatives displaying distinct side-chain orientations were retained for docking. This selection aimed at building a non-redundant ensemble of TRAP1 binding-site conformations, capturing alternative local geometries relevant for ligand recognition. The final ensemble consisted of twelve conformations, eight from Site 1 and four from Site 2.

Site	Replica	Cluster ID	Cluster population
Site 1	Replica 1	site1_rep1_eps1.1.c0	0.402
Site 1	Replica 1	site1_rep1_eps1.1.c1	0.253
Site 1	Replica 1	site1_rep1_eps1.1.c2	0.215
Site 1	Replica 2	site1_rep2_eps1.15.c0	0.421
Site 1	Replica 2	site1_rep2_eps1.15.c1	0.315
Site 1	Replica 2	site1_rep2_eps1.15.c2	0.211
Site 1	Replica 3	site1_rep3_eps1.1.c0	0.393
Site 1	Replica 3	site1_rep3_eps1.1.c1	0.232
Site 2	Replica 1	site2_rep1_eps1.1.c0	0.356
Site 2	Replica 1	site2_rep1_eps1.1.c1	0.261
Site 2	Replica 1	site2_rep1_eps1.1.c2	0.151
Site 2	Replica 2	site2_rep2_eps1.15.c2	0.500

Figure 3.6: Summary of clustering analysis

3.3 Ensemble docking-based prioritization of LSTM-generated TRAP1 Ligands

Before applying docking to the LSTM-generated molecules, we first evaluated whether all twelve MD-derived TRAP1 conformations should be used in the ensemble docking. Although using all available conformations is possible, this is not necessarily the most effective strategy. Indeed, some receptor structures may represent binding-site arrangements that are less suitable for ligand recognition or that introduce noise into the docking results. Including such conformations could increase the computational cost and reduce the ability of the protocol to prioritize true TRAP1 binders. For this reason, the twelve receptor conformations were first tested in a virtual screening experiment using known TRAP1 active ligands and the corresponding decoys. The aim was to identify which conformations, or combinations of conformations, were most able to rank known actives above decoys. In this way, the final docking ensemble was selected not only on the basis of structural diversity, but also on its ability to reproduce the expected enrichment of known TRAP1 ligands. To do this, conformations were evaluated based on their ability to discriminate between known active ligands and chemically plausible presumed inactive compounds, known as decoys. Decoys are molecules that closely match the physicochemical properties of active ligands (e.g., molecular weight, logP, hydrogen bond donors/acceptors...), while being topologically distinct. The strategy is therefore to proceed with the docking of actives and decoys in the twelve receptor structures and subsequently choose the combination of receptors that yields the best enrichment factor EF1%. The reasoning behind this choice lies in the fact that optimal receptor conformations should preferentially rank true active ligands whilst penalizing decoys.

3.3.1 Generation, filtering and quality assessment of the decoy set

The first step is to generate decoys. This was done through DeepCoy[50], a deep learning-based approach which generates property-matched, structurally dissimilar decoys conditioned on each active ligand. This is done by encoding the active ligand into a continuous latent representation, using a graph neural network (GNN), which captures essential features of the molecule, including its size, functional groups, and overall connectivity. Subsequently, this representation is perturbed and decoded to generate a new molecular graph, which is constructed atom-by-atom and bond-by-bond while respecting chemical valence rules. With DeepCoy, by generating on a

per-active basis, each active has its own tailored set of decoys. This is an advantage compared to standard database-derived decoys sets, such as DUD-E. Although these represent standard benchmark sets, they have several limitations. Although decoys are selected to match key physicochemical properties of active ligands, residual biases often remain and can be exploited by virtual screening methods, such as docking. In addition, by relying on pre-existing molecular databases, chemical diversity and representativeness is restricted; the pool of available decoys is inherently limited and may over-represent certain scaffolds. DeepCoy thus enables more precise property matching and reduces dataset bias; its output does however depend on training data and can produce chemically implausible structures that require post-generation filtering. Furthermore, DeepCoy decoys are not experimentally validated as inactive, meaning that some generated molecules could potentially bind the target.
We decided to generate 1000 decoys per active and then filter these to select 40 final decoys per active resulting in a final decoy set of 1720 compounds. This ratio is consistent with established benchmarking datasets[51;52]; it is important to highlight that decoy quality and physicochemical matching matter more than hitting an exact count. In the filtering stage, firstly, exact charge is enforced, so only candidate decoys with the same charge as the relative active are retained. Remaining candidates were then ranked according to their Mahalanobis distance from the active ligand in the selected physicochemical descriptor space:

$$D_M(x_d, a) = \sqrt{(x_d - a)^T S^{-1} (x_d - a)} \tag{3.1}$$

where x_d is the descriptor vector of a candidate decoy, a is the descriptor vector of the corresponding active ligand, and S is the covariance matrix of the descriptor space. In this work, each vector included five molecular descriptors: molecular weight, logP, hydrogen-bond donors, hydrogen-bond acceptors and TPSA. Before calculating Mahalanobis distances, the molecular descriptors were standardized using StandardScaler. This preprocessing step subtracts the mean and divides by the standard deviation for each descriptor, bringing MW, logP, HBD, HBA and TPSA onto a comparable numerical scale Mahalanobis distance was used because it measures active–decoy similarity while accounting for the variance of each descriptor and the correlations between descriptors. Thus, a low Mahalanobis distance indicates that a candidate decoy closely matches the corresponding active in the selected physicochemical property space. Conversely, higher values indicate poorer property matching.

Decoy candidates are therefore ranked by increasing Mahalanobis distance and a similarity filter is applied: only decoys with a Tanimoto coefficient T<0.5 to the respective active are considered. Also a similarity of T=0.7 is enforced between decoys. After this filter, for each active, we selected the 40 decoys with lowest Mahalanobis distance.

For all decoys we verified that these do not share any scaffold with the actives via a Bemis-Murcko scaffold analysis thus confirming structural diversity, compared to actives, of the decoy dataset. Also we wanted to check decoy bias with a property-only Random Forest (RF) classifier. The idea is to assess if the classifier is able to distinguish between actives (label=1) and decoys (label=0) using as input only the five properties: MW, logP, HBD, HBA and TPSA. This is a critical test because artificially biased decoy sets can lead to an overestimated virtual screening performance. Subsequent docking, therefore, could reflect property-based discrimination rather then true binding interactions. We must remember in fact that docking scoring functions incorporate energy terms that are strongly correlated with global physicochemical properties. For example, contributions such as van der Waals interactions, hydrophobic contacts, and desolvation effects often scale with molecular size, lipophilicity, and surface area. During RF property-only training, a grouped cross-validation strategy (StratifiedGroupKFold) was employed to prevent data leakage between actives and their corresponding decoys. The model achieved a ROC-AUC of 0.74 indicating moderate separability between actives and decoys based on physicochemical descriptors, suggesting that some property bias remains in the dataset. Whilst an ideal decoy set would display minimal separability (ROC-AUC $\approx$ 0.5), achieving perfect property matching is challenging due to the simultaneous enforcement of structural dissimilarity to active, diversity between decoys and no decoy reuse among actives. This result, in any case, remains within ranges reported for other benchmarking datasets, such as DUD-E[50].

3.3.2 Docking on MD-derived TRAP1 conformations

The next step consisted of docking the actives and decoys benchmark set in the twelve representative TRAP1 conformations previously obtained via binding-ste cluster analysis. As previously noted, two actives (ZINC908175388 and ZINC237713788) each have two stereoisomers that experimentally show TRAP1 selectivity. Although stereochemical information was not used during LSTM fine-tuning and molecule generation, it was explicitly considered at the docking stage because stereochemistry affects ligand three-dimensional geometry and, consequently, protein–ligand interactions. Thus, the total number of actives now amounts to 45 compounds. In order to carry out docking, all ligands, actives and decoys, were prepared with Maestro LigPrep. This converts the SMILES strings to 3D conformers and performs the following steps: hydrogen addition, ionization state adjustment, tautomer generation, stereoisomer expansion and energy minimization. For actives, during conformer generation, specified chirality of SMILES was retained, whilst for decoys, all possible stereoisomers were generated. Ligprep generated 89 prepared active ligand structures. For decoys, 10214 structures were generated by LigPrep. These were then filtered as to not exceed 100 decoy isomers per active whilst maintaining the same active/decoy isomer ratio for all actives. The decoy-to-active ratio is critical in benchmarking, as enrichment metrics such as EF1% are sensitive to data composition.

First, we looked at the "State Penalty" value for all structures, this reflects the relative energetic cost associated with a given protonation or tautomeric state, with higher values indicating less favourable and less probable states under physiological conditions. For each decoy, we retained only the isomers with the lowest state penalty. After this step, we still had multiple isomeric forms with identical or very similar (low) state penalty values for a given molecule. Therefore, we subsequently inspected the "Energy" value and kept, for each decoy, the lowest-energy isomer as well as isomers within an energy window of two Kcal/mol, up to a maximum of four isomers. This ensures energetically favorable conformations while limiting redundancy from multiple similar structures. The final decoy set consisted of 3324 structures, corresponding to 60-80 decoy-structures per active ligand.

For each receptor, a docking grid was generated around the ATP binding site. This was done by centering the grid on the ATP molecule of the relevant binding site (some receptors are of site 1 whilst others site 2), the ligand box was expanded from 0.10 Å to 0.12 Å in all directions. Also, after visually inspecting the two binding sites, we allowed, for some of the closer residues, the rotation of rotatable groups.

All prepared actives and decoys were then docked, independently, in all twelve receptors. During docking, Glide Standard Precision (SP) was employed and a maximum of five poses was retained for each ligand.

3.3.3 Enrichment-based selection of the final receptor ensemble

As previously mentioned, to select the receptor conformations that most effectively ranked known TRAP1 actives above decoys, enrichment factor analysis was performed on the active/decoys docking results. Because docking scores are not directly comparable across different receptor structures, ligand performance was evaluated using relative rankings within each receptor rather than raw docking scores across receptors. For each receptor conformation, all poses and prepared states corresponding to the same parent ligand were grouped and only the best-scoring pose (by docking score) per ligand was retained. Ligands were then ranked within each receptor according to their docking score, with more favorable scores (coinciding with a more negative value) corresponding to better ranking positions. Subsequently, all possible combinations of the 12 receptor conformations were generated, ranging from single structures to the full ensemble. For each combination, ligand rankings were aggregated across receptors, and the enrichment factor EF1% was calculated.
The enrichment factor quantifies the ability of a screening method to recover active compounds in the top-ranked fraction of a screened library compared with random selection.
EF1% is defined as:

$$EF_{1\%} = \frac{N_{top\ 1\%\ actives}}{N_{total\ actives}} \cdot 100 \tag{3.2}$$

To evaluate performance two ranking strategies were employed: EF1% min and EF1% avg. In EF1% min, for each ligand, the best ranking position across all receptors is retained whilst for EF1% avg, the average rank across receptors is considered. EF1% avg was selected as the criterion for identifying the optimal combination. This approach favors ligands that consistently achieve favorable rankings across multiple receptor conformations, hence providing a more robust and generalizable measure of binding performance. In contrast, EF1% min may overestimate performance by rewarding ligands that rank highly in only a single receptor conformation, potentially leading to increased false positives. Although the highest EF1% avg value was obtained with a two-receptor combination, this ensemble was considered too limited to represent the conformational variability of the TRAP1 ATP-binding site for prospective screening of the LSTM-generated molecules.

We therefore examined the next high-performing EF1% avg plateau, corresponding to an EF1% avg of 18.93. This value was achieved by 21 different receptor combinations, suggesting that several receptor subsets provided comparable enrichment performance. To select a non-redundant and structurally diverse ensemble from this group, we analysed the recurrence of each receptor conformation across the 21 high-performing combinations.

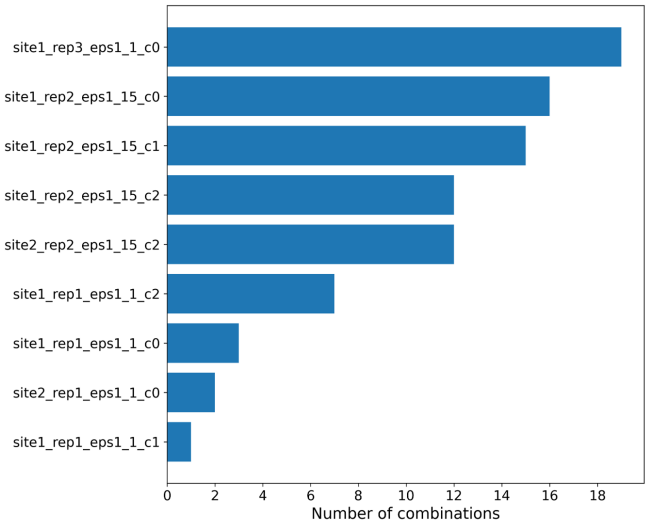


Figure 3.7: Frequency of receptors across the 21 high-performing combinations

Conformations appearing most frequently were retained, resulting in a final five-receptor ensemble comprising four Site 1 conformations and one Site 2 conformation. This selection strategy balances enrichment performance with sufficient representation of binding-site conformational variability.

The same ranking, aggregation and EF1% calculation procedure was repeated using ligand efficiency,defined as docking score normalized by the number of heavy atoms, instead of docking score. This analysis yielded a lower EF1% value of 16.56. Therefore, raw docking score was retained as the primary ranking metric, while ligand efficiency was considered as a control for potential size-dependent scoring effects. It is important to note that during the aggregation of ligand rankings across receptor conformations, only ligands present in all receptors of a given combination were considered. This intersection-based approach ensures that rankings are computed on a consistent set of compounds, avoiding biases arising from missing docking results in individual receptor structures. By inspecting all poses in all receptors of the chosen combination, we noticed that the final number of docked decoys are 1552 (out of 1720 input decoys). This is expected due to the intersection constraint: not all decoys survive docking in all of the receptors, therefore, they were excluded from the consensus ranking and did not contribute to the EF1% calculation.

For the selected ensemble (EF1% avg = 18.93), the top 1% of the ranked list corresponds to 16 compounds, given the dataset size of 1597 ligands (actives and decoys). Within this subset, approximately 8.5 active compounds on average were recovered out of a total of 45 actives. By comparison, random selection would be expected to yield approximately 0.45 actives, indicating substantial enrichment. Assuming a theoretical maximum EF1% value of 35.49, corresponding to the scenario in which all top-ranked compounds are active, the observed performance represents approximately 53% of the maximum achievable enrichment. This indicates substantial retrospective enrichment of known TRAP1 actives over decoys and supports the use of the selected receptor ensemble for docking the LSTM-generated molecules.

3.3.4 Prioritization of LSTM-generated molecules

Following identification of the optimal receptor ensemble, the next step was to dock the LSTM-generated molecules (9482 compounds) into each of these five conformations. The generated ligands were prepared with LigPrep, during which possible stereoisomers were generated according to the same preparation settings used for the benchmark set, resulting in 22425 prepared ligand structures. The same post-filtering of the benchmarking set, based on "State Penalty" and "Energy", is applied to the generated molecules yielding 10525 final prepared structures for docking. The docking of these into the five best TRAP1 receptors follows the same protocol as for the actives and decoys, using Glide Standard Precision (SP) and retaining up to five poses per ligand structure. The docking results were then aggregated using the same procedure used for the benchmarking dataset: ranking of the best pose per ligand within receptor (by docking score) and then aggregation across receptors through intersection. This ensured methodological consistency between model evaluation and receptor selection, allowing the generated molecules to be interpreted within the same scoring framework used for known actives and decoys. From the benchmarking set, False Positive Rate thresholds were derived by analyzing the decoy score distribution. False Positive Rate, or FPR, measures the fraction of non-binders (decoys) that are incorrectly classified as "actives" by a scoring threshold:

$$FPR = \frac{FP}{FP + TN} \tag{3.3}$$

Specifically, score cutoffs corresponding to the 1st and 5th percentiles of the decoys were used to define FPR1% and FPR5% thresholds. First of all, decoys were sorted according to their ensemble docking score, with lower values indicating stronger predicted binding.

The FPR1% cutoff is then defined as the score value at the 1st percentile of this decoy distribution, meaning that only the top 1% lowest-scoring decoys achieve scores equal to or better than this threshold. Likewise, the FPR5% cutoff corresponds to the 5th percentile of the decoy score distribution. In practical terms, FPR1% and FPR5% define confidence regions within the docking score space. The FPR1% threshold corresponds to an extremely tight region in which only 1% of decoys are incorrectly classified as high-affinity binders, and thus represents a stringent active-like docking score region with minimal expected false positives. The FPR5% threshold defines a broader region where up to 5% of decoys fall below the cutoff, corresponding to a less stringent active-like score region. FPR1% corresponded to a docking score of -9.26 kcal/mol, allowing approximately 17 decoys to fall within the active-like region. On the other hand, FPR5% equals a score of -7.29 kcal/mol, including approximately 81 decoys within the same region.

The behavior of the selected receptor ensemble was further characterized by evaluating the recovery of known actives at these FPR thresholds, by calculation of the True Positive Rate or TPR:

$$TPR = \frac{TP}{TP + FN} \tag{3.4}$$

The ensemble achieved a TPR of 22.2% at the FPR1% threshold and 31.1% at the FPR5% threshold, indicating that a significant fraction of experimentally reported TRAP1 ligands is concentrated within the low-score region defined by non-binder behavior, showing meaningful early enrichment of actives by the ensemble docking model.

The same score-threshold framework was then applied to evaluate the selected molecules generated by the LSTM network (9167 ligands). The same FPR thresholds were applied to the generated molecules so for each generated compound we assessed whether its ensemble docking score was equal or lower than the FPR1% (-9.26 kcal/mol) and FPR5% (-7.29 kcal/mol) cutoff. This resulted in 549 generated molecules (5.99% of total) achieving scores equal to or better than the FPR1% cutoff, while 1074 molecules (11.72%) passed the FPR5% threshold. The presence of LSTM-generated molecules within these regions suggests that the generative model preferentially explores areas of chemical space associated with active-like binding behavior.

Overall, this post-docking prioritization workflow involved three main steps. First, the selected receptor ensemble was characterized using EF1% and TPR calculations to assess how well known TRAP1 actives were recovered relative to decoys. Second, the decoy docking-score distribution was used to define stringent and less stringent active-like score thresholds, corresponding to FPR1% and FPR5%, respectively. Third, these thresholds were applied to the LSTM-generated molecules to identify compounds whose docking scores fell within the benchmark-defined active-like regions.

3.3.5 Chemical diversity and ADMET analysis of prioritizedgenerated candidates

The FPR1% subset, comprising 549 generated compounds, was further analyzed to characterize its chemical diversity and physicochemical properties To further characterize the chemical space retained after docking-based prioritization, the structural similarity between the selected FPR1% molecules and the known TRAP1 active compounds was evaluated using the Tanimoto coefficient calculated from molecular fingerprints. Two metrics were considered: (i) the maximum Tanimoto similarity of each generated molecule to its nearest known active compound and (ii) the mean similarity calculated against the complete set of 45 known TRAP1 active ligands. Compared with the complete generated dataset, the FPR1% subset showed higher similarity to known TRAP1 actives in both similarity metrics. The mean maximum Tanimoto similarity increased from 0.3728 for the full set of generated molecules to 0.5640 for the selected subset. Likewise, the mean similarity calculated over all known active compounds increased from 0.1638 to 0.2005. These results indicate that the docking-prioritized subset is shifted toward molecules closer to the known TRAP1 active chemical space, reflecting finetuning distribution, while still maintaining an overall intermediate level of similarity. Consistent with this observation, 439 out of the 549 selected molecules exhibited a maximum Tanimoto coefficient greater than 0.5 with at least one known active compound, indicating that most selected compounds retained moderate similarity to the known TRAP1 ligand set.

The distribution of maximum Tanimoto similarities within the selected subset is shown in Figure 3.8:

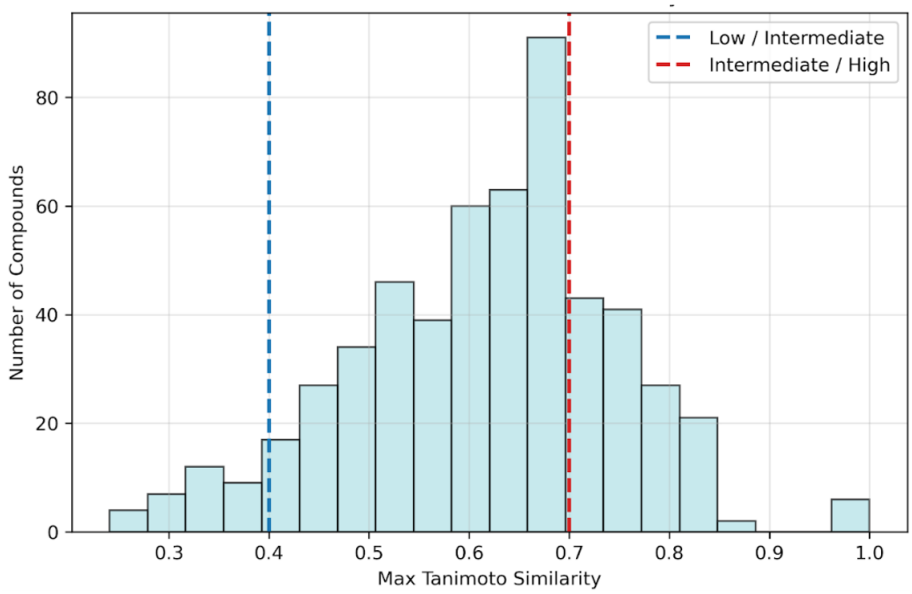


Figure 3.8: Distribution of maximum Tanimoto similarity between FPR1%-selected generated molecules and their nearest known TRAP1 active.

The histogram displays the frequency of the 549 molecules as a function of their maximum similarity to the known active set. Based on the maximum Tanimoto coefficient, using operational thresholds, molecules were divided into three similarity classes: low similarity (T=0.0–0.4), intermediate similarity (T=0.4–0.7), and high similarity (T=0.7–1.0). Most molecules belonged to the intermediate similarity class (379 compounds), followed by the high similarity class (136 compounds), whereas only 34 molecules exhibited a maximum Tanimoto coefficient below 0.4. Compounds with Tanimoto values close to 1.0 were inspected to exclude exact duplicates of known actives This distribution shows that the selected subset is predominantly composed of compounds sharing moderate to high structural similarity with at least one known TRAP1 ligand. To assess whether docking ranking was associated with structural similarity to known actives, the maximum Tanimoto similarity of each selected molecule was plotted against its docking rank (Figure 3.9).

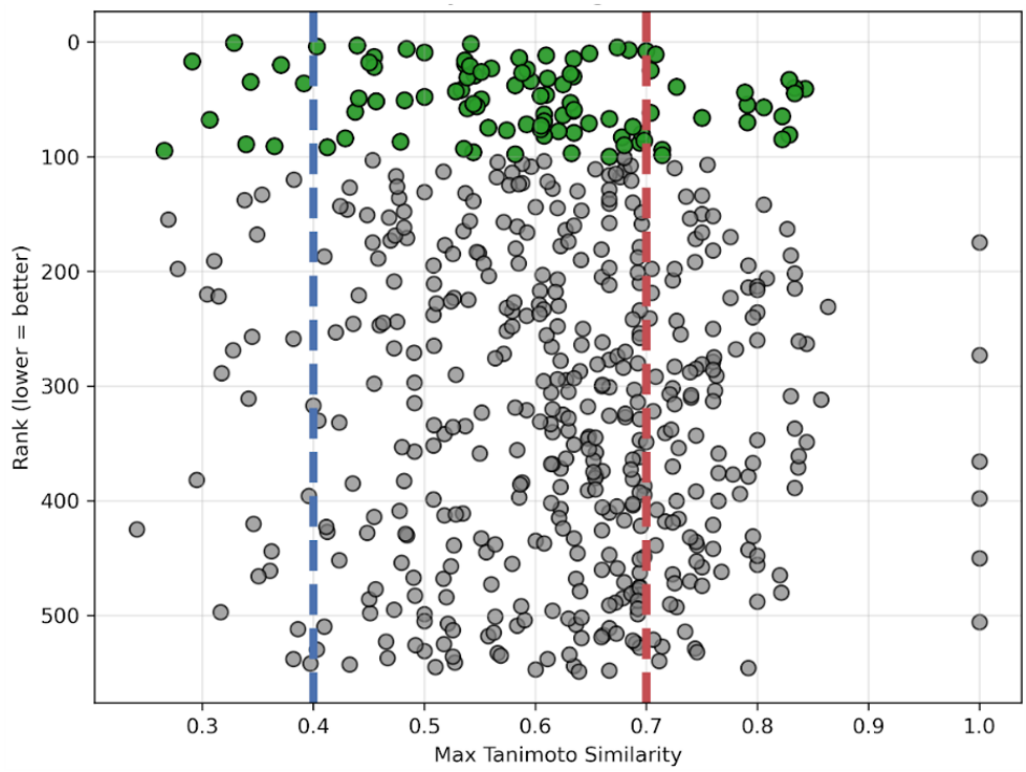


Figure 3.9: Maximum Tanimoto similarity versus docking rank for FPR1%-selected generated molecules. Green points indicate top-ranked compounds (docking score $<$ -10.2076 kcal/mol); grey points indicate the remaining selected molecules.

No clear relationship between molecular similarity and docking rank was observed. Molecules belonging to the high-, intermediate-, and low-similarity classes are distributed throughout the entire ranking, indicating that favorable docking scores are not restricted to compounds most closely resembling the known active ligands. These results suggest that the docking-based prioritization is not driven solely by ligand similarity but may also reflect predicted structural compatibility with the ATP-binding site, allowing chemically diverse candidates to be retained among the highest-ranked molecules.

To provide an initial assessment of the suitability of the selected compounds as potential drug candidates, physicochemical properties were calculated using the built-in QikProp tool in Maestro. These predictions provide information relevant to the evaluation of physicochemical properties and descriptors of the generated molecules that may influence absorption, distribution, metabolism and excretion (ADME).

For the main physicochemical properties, we indicate the mean value, range, recommended range and % of molecules within recommended range in the following Table 3.10:

Property	Mean	Range	Recommended range	% of molecules within recommended range
Molecular Weight (MW)	359.02	245.28 – 576.69	≤ 500	96.5
Octanol-water partition coefficients (logP)	2.13	-2.10 – 4.81	-2 – 6.5	99.8
Topological Polar Surface Area (TPSA), Å^2	110.61	61.04 – 236.21	≤ 140	88.5
Hydrogen-bonding donors (HBD)	3	0 – 8	≤ 5	99.5
Hydrogen-bonding acceptors (HBA)	6	4 – 14	≤ 10	95.4
Rotable bonds	4	1 – 22	≤ 10	99.1
Water solubility (QplogS)	-4.21	-7.61 – -1.03	-6.5 – 0.5	97.3
Predicted apparent permeability across Caco-2 cells (QPPCaco), nm/s	170.43	0.01 – 1945.81	≥ 500	2.2

Figure 3.10: Main physicochemical properties calculated via Qikprop

where logP is a measurement of the lipophilicity of the neutral molecule, rotable bonds one of molecular flexibility, and, TPSA and QPPCaco associated with permeability and absorption.

The generated molecules exhibited favorable physicochemical properties overall, with mean molecular weight, logP, hydrogen-bonding characteristics, and solubility predictions falling largely within the recommended ranges. Compounds show moderate polarity with 88.5% of these remaining within the preferred range. Predicted Caco-2 permeability was comparatively lower, indicating that absorption may be limited for a substantial fraction of the generated molecules; this is consistent with the relatively high polar surface area.

Another property which was calculated via Qikprop is molecular charge. Of the 549 compounds, 509 were predicted to be negatively charged, 37 neutral and only 3 positively charged. The negative charge of molecules may contribute to favorable electrostatic interactions within the TRAP1 ATP-binding site and can also improve aqueous solubility. However, negatively charged molecules represent a limitation for membrane permeability and thus mitochondrial accumulation. This aspect must therefore be evaluated when deciding on compounds to be synthesized. For example, considering that 220 out of 549 compounds are carboxylic acids, ester prodrugs could be explored as a possible strategy to transiently mask the negative charge

and improve cellular uptake. Once inside the cell, these esters may be hydrolyzed to regenerate the corresponding active acid, although the efficiency of intracellular hydrolysis and mitochondrial delivery would need to be experimentally assessed.
To further assess the drug-likeness of the generated compounds, compliance with Lipinski's rule of five was evaluated in Fig 3.11 below:

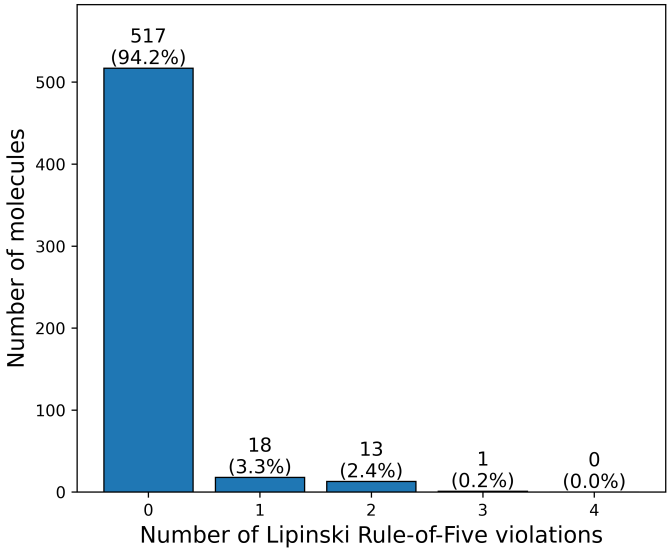
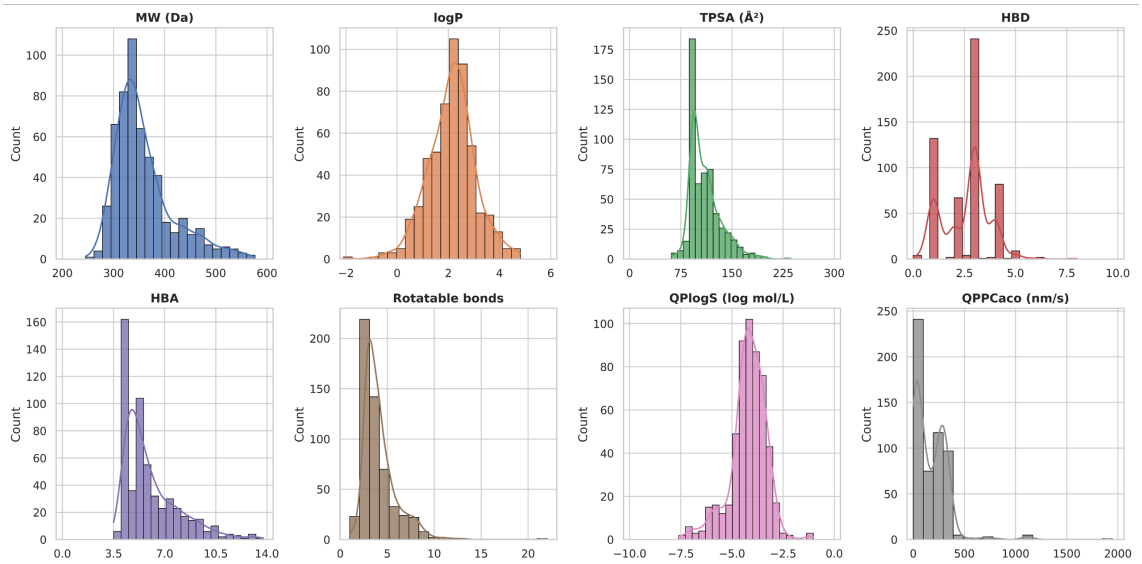


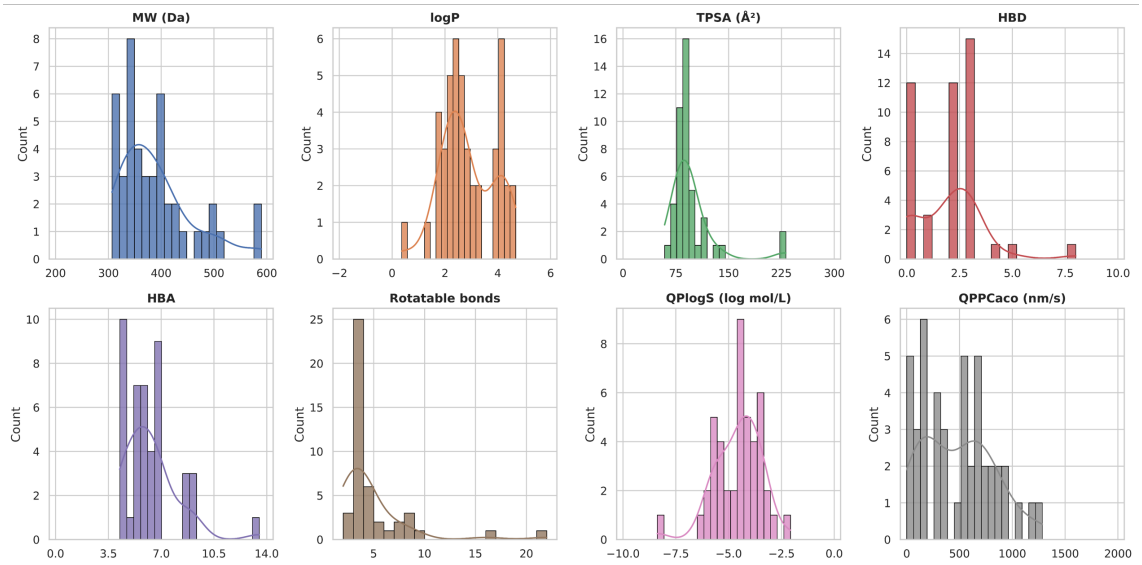
Figure 3.11: Lipinski's rule of five violation of generated compounds

The distribution of Lipinski's rule-of-five violations indicates that the generated library predominantly consists of drug-like molecules. Among the 549 generated compounds, 517 (94.2%) fully complied with all four Lipinski criteria, while only 18 (3.3%) and 13 (2.4%) violated one and two rules, respectively. A single compound violated three rules, and no molecules violated all four criteria. These findings demonstrate that the LSTM model effectively learned the physicochemical constraints of the training compounds and generated molecules that largely reside within the conventional oral drug-like chemical space.

Subsequently, the distributions of key physicochemical descriptors predicted by QikProp were compared with those of the known TRAP1 active compounds in Fig 3.12a and Fig 3.12b below:



(a) Property distributions of generated molecules calculated with QikProp.



(b) Property distributions of active compounds calculated with QikProp.

Figure 3.12: Comparison of the main property distributions for generated molecules and active compounds.

Overall, the generated library exhibited a physicochemical profile largely consistent with that of the known TRAP1 inhibitors, with substantial overlap observed for molecular weight, lipophilicity, and solubility-related descriptors. Notably, predicted Caco-2 permeability (QPPCaco) showed a clear shift, with generated compounds predominantly spanning the 0–500 nm/s range, whereas the actives extended to higher permeability values (up to 1300 nm/s). Such shift is consistent with the slightly increased polarity observed in the generated compounds, as reflected by TPSA and hydrogen-bonding descriptor, and suggests that generated compounds are less membrane-permeable than known TRAP1 binders. This must be taken into account when selecting compounds for synthesis and biological evaluation.

To qualitatively assess the docking results and verify that the generated molecules adopt biologically plausible binding modes, a visual inspection of representative compounds from the selected library was performed. In particular, the analysis of compound gen5634, which achieved a docking score ranking of 30 and presents a maximum Tanimoto coefficient of 0.55 (Intermediate class), is detailed as follows. The predicted binding pose of gen5634 was analyzed in each of the five selected TRAP1 receptor conformations and compared with the corresponding docking pose of ATP, the natural ligand of the ATP-binding site.

In Table 3.13 the comparison of key-interactions for gen5634 and ATP in the TRAP1 conformations is further detailed:

KEY INTERACTIONS	RESIDUES	ATP					GEN5634				
		site1_rep3_eps1_1_c0	site1_rep2_eps1_15_c0	site1_rep2_eps1_15_c1	site1_rep2_eps1_15_c2	site2_rep2_eps1_15_c2	site1_rep3_eps1_1_c0	site1_rep2_eps1_15_c0	site1_rep2_eps1_15_c1	site1_rep2_eps1_15_c2	site2_rep2_eps1_15_c2
H BOND	Asn50					✓	✓				
	Asp53						✓				
	Asp89	✓	✓	✓	✓	✓	✓		✓	✓	
	Asn102		✓	✓	✓			✓			✓
	Ser109	✓	✓	✓	✓	✓			✓		✓
	Gly110		✓	✓	✓	✓			✓		
	Ser111		✓	✓	✓	✓			✓	✓	
	Gln131		✓	✓	✓						
	Phe132	✓	✓	✓	✓	✓					
	Gly133					✓	✓	✓			
	Val134						✓	✓			
	Gly135	✓	✓	✓	✓	✓	✓				
	Phe136	✓									
	Thr182	✓	✓	✓	✓						
	Arg333	✓	✓	✓	✓	✓					
SALT BRIDGE	Gly93										✓
	Arg333										✓
HALOGEN BOND	Gly93						✓	✓			
	Thr182						✓	✓			
π - CATION	Lys57							✓		✓	✓

Figure 3.13: Key-interactions for gen5634 and ATP in the five TRAP1 conformations

Across all receptor conformations, gen5634 adopted a binding orientation highly consistent with that of ATP. Comparison of the protein-ligand interactions revealed substantial overlap between the generated compound and ATP. Similar to ATP, gen5634 established hydrogen-bond interactions with several residues that are recurrently involved in nucleotide recognition, including Asp89, Ser109, Ser111, Gly133 and Val134, while interactions with Thr182 were also observed in multiple receptor conformations. This suggests that gen5634 maintains the interaction network characteristic of ATP binding. Particularly significant is the interaction with Asn102, observed in multiple docking poses of gen5634; such residue, as mentioned earlier in this Chapter, has been reported to contribute to TRAP1 selectivity over the othere HSP90 paralogs. In addition to the interactions shared with ATP, gen5634 formed supplementary contacts, including π-cation interactions with Lys57, halogen bonding in some receptor conformations, and a salt bridge with Arg333. These interactions may contribute to stabilizing the ligand within the binding pocket and show how the generated molecule can exploit alternative interaction patterns while preserving the essential recognition features of ATP.
The binding mode of gen5634 within the TRAP1 ATP-binding site (site1_rep2_eps1_15_c0) is illustrated in Figure 3.14 , highlighting the key protein–ligand interactions described above.

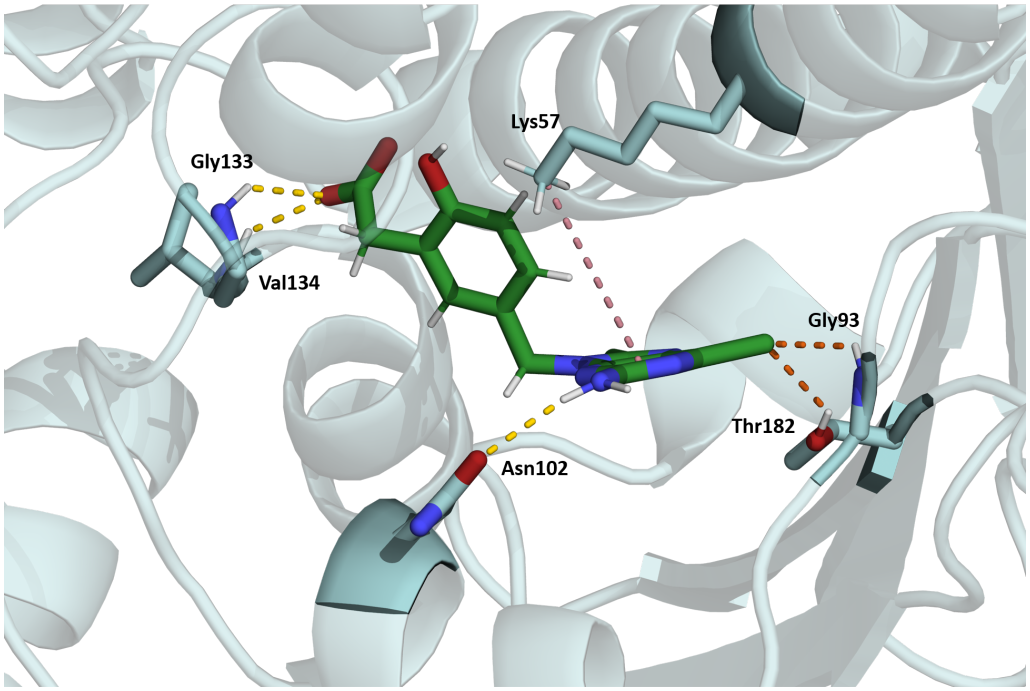


Figure 3.14: Docked pose of gen5634 in the TRAP1 receptor conformation site1_rep2_eps1_15_c0. Hydrogen bonds are shown in yellow, π-cation interactions in pink, and halogen bonds in orange

Additional generated molecules were visually inspected, including compounds displaying different levels of structural similarity to the known TRAP1 actives. In the examined cases, the ligands occupied the ATP-binding pocket with binding orientation comparable to that of ATP and established interactions with residues involved in nucleotide recognition. For example, compound gen4947, bearing a maximum Tanimoto coefficient of 0.29, displayed hydrogen-bond interactions with following residues: Lys57, Val134, Asn102. Once again, we observe an interaction with key residue Asn102, this time with a generated compound belonging to the low similarity class.
This analysis suggests that the generative model was able to produce molecules bearing chemical features compatible with ATP-site recognition. The subsequent docking-based prioritization selected compounds that preserved key interaction patterns within the TRAP1 ATP-binding pocket while also introducing alternative contacts and structurally diverse chemotypes.

3.3.6 Conclusions and perspectives

The objective of this thesis was to develop a target-specific Chemical Language Model (CLM) capable of generating novel compounds with potential affinity toward TRAP1 by combining deep generative modeling with structure-based virtual screening. Transfer learning successfully biased the learned molecular distribution from a broad drug-like chemical space toward one enriched in TRAP1-related chemical features, suggesting that biologically relevant chemical patterns can be learned even from relatively small target-specific datasets. The selected molecules generated from the finetuned model were evaluated using an ensemble docking strategy based on representative TRAP1 conformations extracted from molecular dynamics simulations; this provided a structure-based assessment of their predicted compatibility with the TRAP1 ATP-binding site while accounting for receptor flexibility. The active/decoy screening was used to build a data-driven receptor selection framework. By comparing different MD-derived TRAP1 receptor combinations, we identified the ensemble that best reproduced the separation between experimentally reported TRAP1 actives and property-matched decoys. This selected ensemble was then used as a structure-based prioritization tool to rank the LSTM-generated molecules according to their predicted compatibility with the TRAP1 ATP-binding site. Analysis of molecular similarity suggests that the workflow selected compounds closer to known TRAP1 ligands than the full generated set, while still retaining an intermediate level of structural diversity. Molecules displaying high, intermediate, and even low similarity to known actives were distributed throughout the docking ranking,

indicating that favorable docking scores were not restricted to compounds closely resembling known ligands. This observation highlights the complementary roles of the two components of the workflow. While the generative model and likelihood-based filtering guide molecular generation toward chemically relevant regions of TRAP1 ligand space, the docking procedure performs an independent structure-based selection based on predicted compatibility with the ATP-binding site. Consequently, the workflow is capable of identifying not only analogues of known ligands but also structurally less obvious candidates that may represent novel candidate chemotypes. Preliminary physicochemical and ADME-related profiling provided additional information for candidate prioritization.

Overall, the results demonstrate that the proposed workflow effectively integrates ligand-based deep learning with structure-based virtual screening to generate and prioritize novel TRAP1-targeted compounds. Although the workflow provides a rational and efficient strategy for selecting promising candidates TRAP1-tergeted candidates, the most important next step will be experimental validation through chemical synthesis followed by biochemical and cellular assays. Such analyses will be essential to confirm TRAP1 binding, evaluate inhibitory activity and selectivity, and assess cell permeability, mitochondrial accumulation and cellular activity. In the present work, only a subset of high-likelihood molecules generated by the fine-tuned LSTM model was subjected to the complete structure-based evaluation pipeline. Extending this workflow to a larger fraction of the generated library could reveal additional promising candidates, in particular lower-likelihood compounds could comprise novel promising scaffolds.

The presented computational strategy is generalizable across other therapeutic targets, provided that sufficient ligand data are available for fine-tuning and structural information for docking. Consequently this pipeline represents a versatile and scalable framework for target-specific molecular generation in drug discovery.

A future direction would be the integration of this pipeline into an active learning framework. In this approach, the active/decoy benchmark developed in this thesis could provide the training set for a Graph Neural Network (GNN). Experimentally reported TRAP1 actives would be used as positive examples, while property-matched decoys would provide computational negative examples. For both classes, the model could exploit molecular information together with structure-based descriptors extracted from the selected receptor ensemble, including ensemble docking scores and protein–ligand interaction fingerprints. The LSTM model would then generate new candidate molecules, which would be evaluated through the same ensemble docking and interaction analysis workflow. The trained GNN could be

used to prioritize compounds predicted to show active-like molecular and interaction profiles, based on the information learned from the active/decoy benchmark. Candidate selection could then be further refined by applying additional criteria based on chemical diversity and preliminary ADME-related properties, calculated independently from the GNN model. The most promising compounds would then be synthesized and experimentally tested for TRAP1 binding, inhibitory activity and selectivity The experimental results obtained from these assays would provide new data for model refinement, progressively improving the ability of the generative and predictive models to explore TRAP1-relevant chemical space. In this way, the workflow would evolve from docking-based prioritization toward an iterative molecular optimization strategy driven by experimental feedback, while maintaining diversity- and developability-based criteria for candidate selection

Bibliography

[1] Marco Sciacovelli, Giulia Guzzo, Virginia Morello, Christian Frezza, Liang Zheng, Nazarena Nannini, Fiorella Calabrese, Gabriella Laudiero, Franca Esposito, Matteo Landriscina, et al. The mitochondrial chaperone trap1 promotes neoplastic growth by inhibiting succinate dehydrogenase. *Cell metabolism*, 17(6):988–999, 2013.

[2] Otto Warburg. The metabolism of carcinoma cells. *The Journal of Cancer Research*, 9(1):148–163, 1925.

[3] Soichiro Yoshida, Shinji Tsutsumi, Guillaume Muhlebach, Carole Sourbier, Min-Jung Lee, Sunmin Lee, Evangelia Vartholomaiou, Manabu Tatokoro, Kristin Beebe, Naoto Miyajima, et al. Molecular chaperone trap1 regulates a metabolic switch between mitochondrial respiration and aerobic glycolysis. *Proceedings of the National Academy of Sciences*, 110(17):E1604–E1612, 2013.

[4] Bo Zhang, Jing Wang, Zhen Huang, Peng Wei, Ying Liu, Junfeng Hao, Lijing Zhao, Fenglin Zhang, Yaping Tu, and Taotao Wei. Aberrantly upregulated trap1 is required for tumorigenesis of breast cancer. *Oncotarget*, 6(42):44495, 2015.

[5] Giulia Guzzo, Marco Sciacovelli, Paolo Bernardi, and Andrea Rasola. Inhibition of succinate dehydrogenase by the mitochondrial chaperone trap1 has anti-oxidant and anti-apoptotic effects on tumor cells. *Oncotarget*, 5(23):11897, 2014.

[6] Marta Anna Kowalik, Giulia Guzzo, Andrea Morandi, Andrea Perra, Silvia Menegon, Ionica Masgras, Elena Trevisan, Maria Maddalena Angioni, Francesca Fornari, Luca Quagliata, et al. Metabolic reprogramming identifies the most aggressive lesions at early phases of hepatic carcinogenesis. *Oncotarget*, 7(22):32375, 2016.

[7] Soichiro Yoshida, Shinji Tsutsumi, Guillaume Muhlebach, Carole Sourbier, Min-Jung Lee, Sunmin Lee, Evangelia Vartholomaiou, Manabu Tatokoro,

Kristin Beebe, Naoto Miyajima, et al. Molecular chaperone trap1 regulates a metabolic switch between mitochondrial respiration and aerobic glycolysis. *Proceedings of the National Academy of Sciences*, 110(17):E1604–E1612, 2013.

[8] Lorenza Sisinni, Francesca Maddalena, Valentina Condelli, Giuseppe Pannone, Vittorio Simeon, Valeria Li Bergolis, Elvira Lopes, Annamaria Piscazzi, Danilo Swann Matassa, Carmela Mazzoccoli, et al. Trap1 controls cell cycle g2–m transition through the regulation of cdk1 and mad2 expression/ubiquitination. *The Journal of Pathology*, 243(1):123–134, 2017.

[9] Changwook Lee, Hye-Kyung Park, Hanbin Jeong, Jaehwa Lim, An-Jung Lee, Keun Young Cheon, Chul-Su Kim, Ajesh P Thomas, Boram Bae, Nam Doo Kim, et al. Development of a mitochondria-targeted hsp90 inhibitor based on the crystal structures of human trap1. *Journal of the American Chemical Society*, 137(13):4358–4367, 2015.

[10] Sung Hu, Mariarosaria Ferraro, Ajesh P Thomas, Jeong Min Chung, Nam Gu Yoon, Ji-Hoon Seol, Sangpil Kim, Han-ul Kim, Mi Young An, Haewon Ok, et al. Dual binding to orthosteric and allosteric sites enhances the anticancer activity of a trap1-targeting drug. *Journal of Medicinal Chemistry*, 63(6):2930–2940, 2020.

[11] Hye-Kyung Park, Hanbin Jeong, Eunhwa Ko, Geumwoo Lee, Ji-Eun Lee, Sang Kwang Lee, An-Jung Lee, Jin Young Im, Sung Hu, Seong Heon Kim, et al. Paralog specificity determines subcellular distribution, action mechanism, and anticancer activity of trap1 inhibitors. *Journal of medicinal chemistry*, 60(17):7569–7578, 2017.

[12] Clelia Mathieu, Quentin Chamayou, Thi Thanh Hyen Luong, Delphine Naud, Florence Mahuteau-Betzer, Mouad Alami, Elias Fattal, Samir Messaoudi, and Juliette Vergnaud-Gauduchon. Synthesis and antiproliferative activity of 6brcaq-tpp conjugates for targeting the mitochondrial heat shock protein trap1. *European Journal of Medicinal Chemistry*, 229:114052, 2022.

[13] Carlos Sanchez-Martin, Daniela Menon, Elisabetta Moroni, Mariarosaria Ferraro, Ionica Masgras, Justin Elsey, Jack L Arbiser, Giorgio Colombo, and Andrea Rasola. Honokiol bis-dichloroacetate is a selective allosteric inhibitor of the mitochondrial chaperone trap1. *Antioxidants & Redox Signaling*, 34(7):505–516, 2021.

[14] Evelyn Fix. *Discriminatory analysis: nonparametric discrimination, consistency properties*, volume 1. USAF school of Aviation Medicine, 1985.

[15] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152, 1992.

[16] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[17] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65:386–408, 11 1958.

[18] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[19] Peter Ertl, Richard Lewis, Eric Martin, and Valery Polyakov. In silico generation of novel, drug-like chemical matter using the lstm neural network. *arXiv preprint arXiv:1712.07449*, 2017.

[20] Marcus Olivecrona, Thomas Blaschke, Ola Engkvist, and Hongming Chen. Molecular de-novo design through deep reinforcement learning. *Journal of cheminformatics*, 9(1):48, 2017.

[21] Marwin HS Segler, Thierry Kogej, Christian Tyrchan, and Mark P Waller. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS central science*, 4(1):120–131, 2018.

[22] Anvita Gupta, Alex T Müller, Berend JH Huisman, Jens A Fuchs, Petra Schneider, and Gisbert Schneider. Generative recurrent networks for de novo drug design. *Molecular informatics*, 37(1-2):1700111, 2018.

[23] Yangyang Chen, Zixu Wang, Xiangxiang Zeng, Yayang Li, Pengyong Li, Xiucai Ye, and Tetsuya Sakurai. Molecular language models: Rnns or transformer? *Briefings in Functional Genomics*, 22(4):392–400, 2023.

[24] Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.

[25] George Bird and Maxim E Polivoda. Backpropagation through time for networks with long-term dependencies. *arXiv preprint arXiv:2103.15589*, 2021.

[26] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[27] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[28] Josep Arús-Pous, Simon Viet Johansson, Oleksii Prykhodko, Esben Jannik Bjerrum, Christian Tyrchan, Jean-Louis Reymond, Hongming Chen, and Ola Engkvist. Randomized smiles strings improve the quality of molecular generative models. *Journal of cheminformatics*, 11(1):71, 2019.

[29] Esben Jannik Bjerrum. Smiles enumeration as data augmentation for neural network modeling of molecules. *arXiv preprint arXiv:1703.07076*, 2017.

[30] Daniil Polykovskiy, Alexander Zhebrak, Benjamin Sanchez-Lengeling, Sergey Golovanov, Oktai Tatanov, Stanislav Belyaev, Rauf Kurbanov, Aleksey Artamonov, Vladimir Aladinskiy, Mark Veselov, et al. Molecular sets (moses): a benchmarking platform for molecular generation models. *Frontiers in pharmacology*, 11:565644, 2020.

[31] Rıza Özçelik and Francesca Grisoni. How evaluation choices distort the outcome of generative drug discovery. *Journal of Cheminformatics*, 17(1):169, 2025.

[32] Nathan Brown, Marco Fiscato, Marwin HS Segler, and Alain C Vaucher. Guacamol: benchmarking models for de novo molecular design. *Journal of chemical information and modeling*, 59(3):1096–1108, 2019.

[33] Josep Arús-Pous, Thomas Blaschke, Silas Ulander, Jean-Louis Reymond, Hongming Chen, and Ola Engkvist. Exploring the gdb-13 chemical space using deep generative models. *Journal of cheminformatics*, 11(1):20, 2019.

[34] Oleksii Prykhodko, Simon Viet Johansson, Panagiotis-Christos Kotsias, Josep Arús-Pous, Esben Jannik Bjerrum, Ola Engkvist, and Hongming Chen. A de novo molecular generation method using latent vector based generative adversarial network. *Journal of cheminformatics*, 11(1):74, 2019.

[35] Maxime Langevin, Hervé Minoux, Maximilien Levesque, and Marc Bianciotto. Scaffold-constrained molecular generation. *Journal of chemical information and modeling*, 60(12):5637–5646, 2020.

[36] Todd JA Ewing, Shingo Makino, A Geoffrey Skillman, and Irwin D Kuntz. Dock 4.0: search strategies for automated molecular docking of flexible molecule databases. *Journal of computer-aided molecular design*, 15(5):411–428, 2001.

[37] Matthias Rarey, Bernd Kramer, Thomas Lengauer, and Gerhard Klebe. A fast flexible docking method using an incremental construction algorithm. *Journal of molecular biology*, 261(3):470–489, 1996.

[38] Kristin L Meagher, Luke T Redman, and Heather A Carlson. Development of polyphosphate parameters for use with the amber force field. *Journal of computational chemistry*, 24(9):1016–1025, 2003.

[39] Olof Allnér, Lennart Nilsson, and Alessandra Villa. Magnesium ion–water coordination and exchange in biomolecular simulations. *Journal of chemical theory and computation*, 8(4):1493–1502, 2012.

[40] In Suk Joung and Thomas E Cheatham III. Determination of alkali and halide monovalent ion parameters for use in explicitly solvated biomolecular simulations. *The journal of physical chemistry B*, 112(30):9020–9041, 2008.

[41] Tom Darden, Darrin York, Lee Pedersen, et al. Particle mesh ewald: An n log (n) method for ewald sums in large systems. *Journal of chemical physics*, 98(12):10089–10092, 1993.

[42] Herman JC Berendsen, JPM van Postma, Wilfred F Van Gunsteren, ARHJ DiNola, and Jan R Haak. Molecular dynamics with coupling to an external bath. *The Journal of chemical physics*, 81(8):3684–3690, 1984.

[43] Shuichi Miyamoto and Peter A Kollman. Settle: An analytical version of the shake and rattle algorithm for rigid water models. *Journal of computational chemistry*, 13(8):952–962, 1992.

[44] Anna Gaulton, Louisa J Bellis, A Patricia Bento, Jon Chambers, Mark Davies, Anne Hersey, Yvonne Light, Shaun McGlinchey, David Michalovich, Bissan Al-Lazikani, et al. Chembl: a large-scale bioactivity database for drug discovery. *Nucleic acids research*, 40(D1):D1100–D1107, 2012.

[45] Xi Chen, Yuhmei Lin, and Michael K Gilson. The binding database: overview and user's guide. *Biopolymers: Original Research on Biomolecules*, 61(2):127–141, 2001.

[46] R Rondanin, Giacomo Lettini, P Oliva, R Baruchello, C Costantini, C Trapella, D Simoni, T Bernardi, L Sisinni, M Pietrafesa, et al. New trap1 and hsp90 chaperone inhibitors with cationic components: Preliminary studies on mitochondrial targeting. *Bioorganic & medicinal chemistry letters*, 28(13):2289–2293, 2018.

[47] Sung Hu, Mariarosaria Ferraro, Ajesh P Thomas, Jeong Min Chung, Nam Gu Yoon, Ji-Hoon Seol, Sangpil Kim, Han-ul Kim, Mi Young An, Haewon Ok, et al. Dual binding to orthosteric and allosteric sites enhances the anticancer activity of a trap1-targeting drug. *Journal of Medicinal Chemistry*, 63(6):2930–2940, 2020.

[48] Sujae Yang, Nam Gu Yoon, Dongyoung Kim, Eunsun Park, So-Yeon Kim, Ji Hoon Lee, Changwook Lee, Byoung Heon Kang, and Soosung Kang. Design and synthesis of trap1 selective inhibitors: H-bonding with asn171 residue in trap1 increases paralog selectivity. *ACS Medicinal Chemistry Letters*, 12(7):1173–1180, 2021.

[49] Taylor Merfeld, Shuxia Peng, Bradley M Keegan, Vincent M Crowley, Christopher M Brackett, Andrew Gutierrez, Nathan R McCann, Tyelor S Reynolds, Matthew C Rhodes, Katherine M Byrd, et al. Elucidation of novel trap1-selective inhibitors that regulate mitochondrial processes. *European journal of medicinal chemistry*, 258:115531, 2023.

[50] Fergus Imrie, Anthony R Bradley, and Charlotte M Deane. Generating property-matched decoy molecules using deep learning. *Bioinformatics*, 37(15):2134–2141, 2021.

[51] Michael M Mysinger, Michael Carchia, John J Irwin, and Brian K Shoichet. Directory of useful decoys, enhanced (dud-e): better ligands and decoys for better benchmarking. *Journal of medicinal chemistry*, 55(14):6582–6594, 2012.

[52] Jie Xia, Hongwei Jin, Zhenming Liu, Liangren Zhang, and Xiang Simon Wang. An unbiased method to build benchmarking sets for ligand-based virtual screening and its application to gpcrs. *Journal of chemical information and modeling*, 54(5):1433–1450, 2014.

[53] Young Chan Chae, M Cecilia Caino, Sofia Lisanti, Jagadish C Ghosh, Takehiko Dohi, Nika N Danial, Jessie Villanueva, Stefano Ferrero, Valentina Vaira, Luigi Santambrogio, et al. Control of tumor bioenergetics and survival stress signaling by mitochondrial hsp90s. *Cancer cell*, 22(3):331–344, 2012.

[54] Danilo Swann Matassa, Maria Rosaria Amoroso, H Lu, Rosario Avolio, D Arzeni, Claudio Procaccini, Deriggio Faicchia, Francesca Maddalena, V Simeon, IJCD Agliarulo, et al. Oxidative metabolism drives inflammation-induced platinum resistance in human ovarian cancer. *Cell Death & Differentiation*, 23(9):1542–1554, 2016.

[55] Andrew Gutierrez, Jason Archdeacon, and Brian SJ Blagg. Trap1 and its therapeutic potential. *Bioorganic & Medicinal Chemistry Letters*, page 130395, 2025.

[56] Lorenza Sisinni, Francesca Maddalena, Valentina Condelli, Giuseppe Pannone, Vittorio Simeon, Valeria Li Bergolis, Elvira Lopes, Annamaria Piscazzi, Danilo Swann Matassa, Carmela Mazzoccoli, et al. Trap1 controls cell cycle g2–m transition through the regulation of cdk1 and mad2 expression/ubiquitination. *The Journal of Pathology*, 243(1):123–134, 2017.

[57] Hector Flores-Hernandez and Emmanuel Martinez-Ledesma. A systematic review of deep learning chemical language models in recent era. *Journal of Cheminformatics*, 16(1):129, 2024.

[58] Francesca Grisoni. Chemical language models for de novo drug design: Challenges and opportunities. *Current Opinion in Structural Biology*, 79:102527, 2023.

[59] Mingyang Wang, Zhe Wang, Huiyong Sun, Jike Wang, Chao Shen, Gaoqi Weng, Xin Chai, Honglin Li, Dongsheng Cao, and Tingjun Hou. Deep learning approaches for de novo drug design: An overview. *Current opinion in structural biology*, 72:135–144, 2022.

[60] Joshua Noble. What is long short-term memory (lstm)? https://www.ibm.com/think/topics/lstm.

[61] Understanding lstm networks, 2015. https://colah.github.io/posts/2015-08-Understanding-LSTMs/.

[62] Carlos Sanchez-Martin, Elisabetta Moroni, Mariarosaria Ferraro, Claudio Laquatra, Giuseppe Cannino, Ionica Masgras, Alessandro Negro, Paolo Quadrelli, Andrea Rasola, and Giorgio Colombo. Rational design of allosteric and selective inhibitors of the molecular chaperone trap1. *Cell reports*, 31(3), 2020.

[63] Alice Triveri, Carlos Sanchez-Martin, Luca Torielli, Stefano A Serapian, Filippo Marchetti, Giovanni D'Acerno, Valentina Pirota, Matteo Castelli, Elisabetta Moroni, Mariarosaria Ferraro, et al. Protein allostery and ligand design: computational design meets experiments to discover novel chemical probes. *Journal of molecular biology*, 434(17):167468, 2022.

[64] Richard J Loncharich, Bernard R Brooks, and Richard W Pastor. Langevin dynamics of peptides: The frictional dependence of isomerization rates of n-acetylalanyl-n-methylamide. *Biopolymers: Original Research on Biomolecules*, 32(5):523–535, 1992.