

TECHNISCHE HOCHSCHULE MITTELHESSEN

MASTER THESIS

---

# Development of AI tools for the design of therapeutic agents against Covid-19

---

*Author:*

Maximilian Robert Baum

*Supervisor:*

Prof. Dr. Andreas Dominik

*Second Examiner:*

Eric Hartmann

*A thesis submitted in fulfillment of the requirements  
for the degree of Master of Science (M. Sc.)*

*in the*

MNI - Mathematik, Naturwissenschaften und Informatik  
Life-Science Informatics

29. Oktober 2024

# **Eidesstattliche Erklärung**

Hiermit versichere ich, die vorgelegte Thesis selbstständig und ohne unerlaubte fremde Hilfe und nur mit den Hilfen angefertigt zu haben, die ich in der Thesis angegeben habe. Alle Textstellen, die wörtlich oder sinngemäß aus veröffentlichten Schriften entnommen sind, und alle Angaben die auf mündlichen Auskünften beruhen, sind als solche kenntlich gemacht. Bei den von mir durchgeführten und in der Thesis erwähnten Untersuchungen habe ich die Grundsätze guter wissenschaftlicher Praxis, wie sie in der ‚Satzung der Justus-Liebig-Universität zur Sicherung guter wissenschaftlicher Praxis‘ niedergelegt sind, eingehalten. Entsprechend § 22 Abs. 2 der Allgemeinen Bestimmungen für modularisierte Studiengänge dulde ich eine Überprüfung der Thesis mittels Anti-Plagiatssoftware.

Unterschrift:

Datum:

TECHNISCHE HOCHSCHULE MITTELHESSEN

## *Zusammenfassung*

MNI - Mathematik, Naturwissenschaften und Informatik  
Life-Science Informatics

Master of Science (M. Sc.)

**Development of AI tools for the design of therapeutic agents against Covid-19**  
von Maximilian Robert Baum

In dieser Arbeit wurde ein in silico Framework entwickelt, um potenzielle therapeutische Wirkstoffe zu identifizieren die auf die AXL-Rezeptor-Tyrosinkinase abziehen. Im Zentrum stehen der DeepSMILESGenerator und der DeepSMILESPredictor, zwei maschinelle Lernmodelle, die speziell zur Generierung und Vorhersage der Aktivität von Molekülen entwickelt wurden. Durch den Einsatz der speziell für Deep Learning entwickelten DeepSMILES-Notation in der Kombination mit LSTM-Netzwerken wurde das Modell in mehreren Phasen trainiert, einschließlich Transfer Learning und einem speziell entwickelten Reinforcement Learning Ansatz. Dies ermöglicht dem System, valide und funktional relevante Moleküle zu erzeugen. Die erzeugten Moleküle wurden anschließend auf ihre potenzielle biologische Aktivität untersucht und lieferten vielversprechende Ergebnisse für eine experimentelle Validierung. Die Resultate dieser Arbeit unterstreichen die zentrale Rolle von maschinellem Lernen für die Entwicklung von therapeutischen Wirkstoffen in der modernen Arzneimittelforschung, insbesondere in Zeiten globaler Gesundheitskrisen wie der COVID-19-Pandemie. Der entwickelte in silico Workflow demonstriert eindrucksvoll das Potenzial von computergestützten Ansätzen, um die Effizienz und Präzision bei der Entwicklung neuer therapeutischer Lösungen zu steigern.

## *Danksagung*

Ich möchte meinem Betreuer, Prof. Dr. Andreas Dominik, für seine unermüdliche Unterstützung und seine aufschlussreiche Kritik während meiner Forschungsarbeit danken. Sein großes Engagement für akademische Exzellenz und seine akribische Aufmerksamkeit für Details haben diese Masterarbeit maßgeblich geprägt. Ebenso dankbar bin ich für meinen Zweitkorrektor, Eric Hartmann für sein konstruktives Feedback und sein technisches Wissen, durch das die Qualität meiner Arbeit enorm verbessert werden konnte.

Mein Dank gilt auch den Dozenten und Mitarbeitern des Fachbereichs MNI an der Technischen Hochschule Mittelhessen und des Fachbereichs Biologie an der Jus-tus Liebig Universität in Giessen, deren Ressourcen und Unterstützung von unschätzbarem Wert waren. Ich möchte mich auch bei meinen Kollegen für ihre Kame-radschaft und die anregenden Diskussionen bedanken, die mich auf meinem akade-mischen Weg inspiriert haben. Ihre kollektive Weisheit und Ermutigung waren ein Eckpfeiler meiner Forschungserfahrung.

Schließlich gilt mein aufrichtiger Dank meiner Familie und meinen Freunden, die mir mein Studium durch ihre Unterstützung ermöglicht haben und stets ein offenes Ohr für mich hatten und mich beim Korrekturlesen tatkräftig unterstützt haben.

# Inhaltsverzeichnis

<b>Zusammenfassung</b>	<b>ii</b>
<b>Danksagung</b>	<b>iii</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 AXL-Rezeptor-Tyrosin-Kinase . . . . .	2
1.3 In silico AXL-Kinase-Inhibitor Workflow . . . . .	2
<b>2 Theoretischer Hintergrund</b>	<b>5</b>
2.1 DeepSMILES Notation . . . . .	5
2.2 Long Short-Term Memory Networks . . . . .	8
2.3 Transfer Learning . . . . .	10
2.4 Fine-Tuning Reinforcement Learning . . . . .	11
2.5 Multilayer Perceptron Klassifikator . . . . .	12
2.6 Molekulare Deskriptoren . . . . .	13
2.7 Extended-Connectivity Morgan-Fingerprints . . . . .	17
<b>3 Materialien und Methoden</b>	<b>19</b>
3.1 PyTorch & CUDA . . . . .	19
3.2 ChEMBL & PubChem Data . . . . .	20
3.2.1 Compound-Datensatz . . . . .	20
3.2.2 Vokabular . . . . .	20
3.2.3 AXL-Kinase-Inhibitor-Datensatz . . . . .	20
3.3 Trainingsworkflow . . . . .	22
3.3.1 Complete Classification Training (Phase 0) . . . . .	22
3.3.2 Basic Molecular Structures Training (Phase 1) . . . . .	23
3.3.3 Transfer Learning AXL-Kinase-Inhibitor (Phase 2) . . . . .	26
3.3.4 Fine-Tuning Reinforcement Learning Ansatz (Phase 3) . . . . .	26
<b>4 Ergebnisse</b>	<b>30</b>
4.1 DeepSMILESPredictor Training . . . . .	30
4.1.1 Feature Importance . . . . .	32
4.1.2 T-SNE Plot . . . . .	33
4.2 DeepSMILESGenerator Training . . . . .	35
4.2.1 Base-Training Phase 1 . . . . .	35
4.2.2 Transfer Learning Phase 2 . . . . .	37
4.2.3 Fine-Tuning Phase 3 . . . . .	40
4.3 Erweiterte Trainingsansätze . . . . .	42
4.3.1 Fine-Tuning ohne Transfer Learning . . . . .	42
4.3.2 Verlängertes Transfer Learning ohne Fine-Tuning . . . . .	44
4.4 Vergleich der einzelnen Trainingsphasen . . . . .	45

<b>5 Diskussion</b>	<b>47</b>
<b>6 Schlussfolgerung</b>	<b>49</b>
<b>A Häufigkeit der vorkommenden Token im Compound-Datensatz</b>	<b>50</b>

# Abbildungsverzeichnis

2.1	Visualisierung von Ethanol in verschiedenen SMILES-Darstellungen . . . . .	6
2.2	RNN Architektur zur Verarbeitung von sequenziellen Daten . . . . .	9
2.3	Schematische Darstellung eines MLPs . . . . .	12
2.4	Visualisierung des ECFP von Aspirin. . . . .	18
3.1	Darstellung des Vokabulars. . . . .	21
3.2	Visualisierung der Data Augmentation am Beispiel von Aspirin. . . . .	24
3.3	Schematische Darstellung des gesamten Workflows des entwickelten de novo Design Frameworks. . . . .	29
4.1	Berechnung des Class Weights für die positive Klasse (AXL-Kinase-Inhibitoren). . . . .	31
4.2	Verlauf des Trainings- und Testverlusts für den DeepSMILESPredictor.	32
4.3	Konfusionsmatrizen für Trainings- und Testdatensätze des Deep-SMILESPredictor. . . . .	33
4.4	Analyse der Feature Importance . . . . .	34
4.5	T-SNE Visualisierung der Klassifikationsergebnisse des DeepSMILES-Predictor. . . . .	35
4.6	Verlauf des Trainingsverlustes DeepSMILESGenerator (Phase 1). . . . .	36
4.7	T-SNE Plot nach dem Base-Training (Phase 1) des DeepSMILES-Generator. . . . .	37
4.8	Verlauf des Trainingsverlustes DeepSMILESGenerator (Phase 2). . . . .	38
4.9	T-SNE Plot nach dem Transfer Learning (Phase 2) des DeepSMILES-Generator. . . . .	39
4.10	Verlauf der Gesamtgenauigkeit der generierten Moleküle Deep-SMILESGenerator (Phase3). . . . .	40
4.11	Verlauf der Anzahl der generierten Moleküle DeepSMILESGenerator (Phase3). . . . .	41
4.12	T-SNE Plot nach dem Fine-Tuning (Phase 3) des DeepSMILESGenerator.	42
4.13	Verlauf der Gesamtgenauigkeit der generierten Moleküle Fine-Tuning ohne Transfer Learning. . . . .	43
4.14	Verlauf der Anzahl der generierten Moleküle Fine-Tuning ohne Transfer Learning. . . . .	43
4.15	T-SNE Plots nach dem verlängerten Transfer Learning ohne Fine-Tuning. . . . .	44
5.1	Generierte Moleküle des DeepSMILESGenerator. . . . .	48

# Tabellenverzeichnis

4.1 Anzahl invalider Moleküle, Compounds und AXL-Kinase-Inhibitoren in den verschiedenen Trainingsphasen. . . . .	46
--	----

# Abkürzungsverzeichnis

<b>COVID-19</b>	Coronavirus Disease 2019
<b>KI</b>	Künstliche Intelligenz
<b>ML</b>	Machine Learning
<b>AXL</b>	AXL receptor tyrosine kinase
<b>GAS6</b>	Growth-arrest-specific gene-6
<b>QSAR</b>	Quantitative structure–activity relationship
<b>LSTM</b>	Long Short-Term Memory
<b>RNN</b>	Recurrent Neural Networks
<b>MLP</b>	Multilayer perceptron
<b>SMILES</b>	Simplified Molecular Input Line Entry System
<b>ReLU</b>	Rectified Linear Unit
<b>RegExp</b>	Regular Expression
<b>t-SNE</b>	t-Distributed Stochastic Neighbor Embedding

# Kapitel 1

## Einleitung

### 1.1 Motivation

Die rasche Verbreitung der neuartigen Coronavirus-Erkrankung 2019 hat weltweit beispiellose Herausforderungen für die öffentlichen Gesundheitssysteme mit sich gebracht [1]. Die Bemühungen zur Bekämpfung dieser Pandemie haben die dringende Notwendigkeit innovativer Ansätze in der Arzneimittelentwicklung verdeutlicht [2]. Zu diesen Ansätzen haben sich *in silico* Technologien, insbesondere künstliche Intelligenz und maschinelles Lernen, als leistungsfähige Werkzeuge heraustraktallisiert, die neue Strategien zur Gestaltung therapeutischer Wirkstoffe gegen COVID-19 bieten [3,4].

In der modernen Arzneimittelentwicklung stehen traditionelle Methoden zunehmend vor Herausforderungen, insbesondere angesichts der Komplexität biologischer Systeme und der schieren Größe des molekularen Raums, der für potenzielle Wirkstoffe zur Verfügung steht. Die herkömmlichen Ansätze, wie das Hochdurchsatz-Screening, stoßen oft an ihre Grenzen, da sie zwar große Mengen an Molekülen testen können, jedoch bei weitem nicht den gesamten Raum der arzneimittelähnlichen Moleküle abdecken [3]. In Anbetracht der Tatsache, dass schätzungsweise  $10^{60}$  synthetisch zugängliche arzneimittelähnliche Moleküle existieren, wird deutlich, dass neue effizientere Methoden erforderlich sind [5]. Hier bieten *in silico* Ansätze, insbesondere durch den Einsatz von künstlicher Intelligenz und maschinellem Lernen, revolutionäre Möglichkeiten. Diese Technologien ermöglichen es, den riesigen Suchraum gezielt einzuzgrenzen und sogar neue Moleküle *de novo* zu entwerfen, die spezifisch auf biologische Zielstrukturen abgestimmt sind [6]. Damit gehen diese modernen Ansätze weit über die Fähigkeiten traditioneller Methoden hinaus und eröffnen neue Horizonte in der Wirkstoffforschung, die sowohl die Effizienz als auch die Erfolgsraten in der Medikamentenentwicklung signifikant steigern können [7].

Die Motivation hinter dieser Studie bezieht sich auf die Entwicklung eines *in silico* Workflows zur Generierung potenzieller therapeutischer Wirkstoffe gegen COVID-19. Insbesondere wird eine Möglichkeit gesucht, um neue potenzielle Inhibitoren zu entwerfen, die auf die AXL-Rezeptor-Tyrosin-Kinase abzielen – ein vielversprechendes Ziel, das in der Pathogenese von COVID-19 eine Rolle spielt.

## 1.2 AXL-Rezeptor-Tyrosin-Kinase

Die AXL-Rezeptor-Tyrosin-Kinase ist ein Mitglied der TAM (Tyro3, AXL, MER) Rezeptorfamilie, die eine Schlüsselrolle bei der Regulierung des Immunsystems spielt [8]. AXL ist vor allem für seine Fähigkeit bekannt, die Immunantwort zu modulieren, indem es entzündungshemmende Signale verstärkt und die Aktivität von Immunzellen wie Makrophagen und dendritischen Zellen reguliert [8, 9]. AXL wird durch verschiedene Liganden, wie dem GAS6 Protein aktiviert, dies wird durch das Wachstumsarrest-spezifische Gen 6 kodiert und zählt zu der Familie der Vitamin-K abhängigen Proteinen [9]. Durch die Aktivierung wird eine Reihe von zellulären Prozessen ausgelöst, darunter Zellmigration und Überleben. Diese Mechanismen sind auch für die Pathogenese verschiedener Krebsarten relevant, darunter akute myeloische Leukämie, Brust-, Prostata- und nicht-kleinzelligem Lungenkrebs [8, 9]. In Bezug auf COVID-19 haben Studien gezeigt, dass die AXL-Rezeptor-Tyrosin-Kinase nicht nur eine Rolle bei der viralen Replikation spielt, sondern auch den zellulären Eintritt des Virus in die Zellen erleichtert [10, 11]. Dies geschieht durch die Interaktion von AXL mit dem SARS-CoV-2-Spike-Protein, wodurch das Virus leichter in die Wirtszellen eindringen kann. Durch die Blockierung oder Hemmung der AXL-Aktivität hoffen Forscher, die Virusvermehrung und den Krankheitsverlauf bei COVID-19-Patienten zu beeinflussen [11]. Dies könnte insbesondere in Fällen schwerer Verläufe von Bedeutung sein, in der die Kontrolle über die Viruslast entscheidend ist.

## 1.3 In silico AXL-Kinase-Inhibitor Workflow

In dieser Studie wird daher ein umfassender *in silico* Workflow entwickelt, der darauf abzielt, neuartige AXL-Kinase-Inhibitoren zu identifizieren, die potenziell dazu beitragen können, die Virusinfektion zu bekämpfen und gleichzeitig das Fortschreiten der Krankheit zu mildern. Der Workflow wird systematisch in mehrere Schritte unterteilt, um einen zielgerichteten und effizienten Prozess zu gewährleisten.

- Vorhersage der Wirksamkeit von Molekülen
- Generierung spezifischer Moleküle (Zielmoleküle)

**Vorhersage der Wirksamkeit von Molekülen.** Der erste Schritt im Workflow besteht darin, die Wirksamkeit von Molekülen als AXL-Kinase-Inhibitoren zu bestimmen. Maschine Learning Methoden basieren dabei auf verschiedenen Vorgehensweisen zum Beurteilen von Molekülen in aktiv und inaktiv. In den meisten Vorhersagen werden Algorithmen wie *Random Forest* oder *Support Vector Machine* benutzt, um Zusammenhänge aus Cluster abzuleiten, dabei wird aber nicht direkt auf die Eigenschaften der Moleküle zur Bewertung eingegangen [12, 13]. Ein anderer weiterer Ansatz bieten die quantitativen Struktur-Wirkungs-Beziehungen [14]. Hierbei werden quantitative Beziehungen zwischen biologischen, pharmakologischen, chemischen und physikalischen Wirkungen mit der chemischen Struktur beschrieben und als Deskriptoren für die anschließende Klassifizierung der Moleküle benutzt [15].

**Generierung spezifischer Zielmoleküle.** Der zweite Schritt im Workflow besteht darin, spezifische Moleküle zu generieren, die potenziell als AXL-Kinase-Inhibitoren fungieren können. In der Regel besteht der Ansatz darin, Moleküle aus vordefinierten Gruppen von Atomen oder Fragmenten aufzubauen oder virtuelle chemische Reaktion auf der Grundlage von durch Experten kodierten Regeln durchzuführen [16, 17]. In der Praxis erweisen sich beide Strategien oft als ineffektiv, da sie entweder zu schwer synthetisierbare Moleküle hervorbringen oder im Kontext der durch Experten definierten Regeln Moleküle generieren, welche für die meisten Anwendungen zu generalisiert sind - also zu wenig auf die speziellen Eigenschaften der Zielinhibitoren abgestimmt wurden [18–20].

Nachfolgend wird ein sich ergänzender, vollständig datengesteuerter Ansatz für das *de novo* Design von Arzneimitteln präsentiert. Dieses Framework besteht aus zwei Komponenten. Dazu wurde zunächst der DeepSMILESGenerator entwickelt, ein fortschrittliches Deep Learning Modell, das auf *Long Short-Term Memory*-Netzwerken basiert. Einer verbesserten *Recurrent Neural Network*-Variante, die es dem Modell ermöglicht Speicherkapazitäten über einen längeren Zeitraum hinweg zu berücksichtigen [21].

Der DeepSMILESGenerator wird in einem dreistufigen Trainingsprozess entwickelt. In Phase 1 erfolgt das Training auf einen großen allgemeinen Moleküldatensatz, um grundlegende chemische Muster und Strukturen zu erlernen. Das Netz wird hierbei drauf trainiert, Molekülstrukturen als DeepSMILES-Strings (*Deep Simplified Molecular Input Line Entry System*) zu generieren. DeepSMILES ist eine optimierte Version der herkömmlichen SMILES-Notation, die eine effizientere und kompaktere Darstellung ermöglicht [22]. Im Gegensatz zur klassischen SMILES-Notation, die häufig für chemische Darstellung verwendet wird, nutzt der hier vorgestellte Ansatz die DeepSMILES-Notation gezielt, um die Modelleistung zu verbessern und die Effizienz bei der Verarbeitung großer Datenmengen zu steigern.

In Phase 2 wird das Modell im Rahmen eines Transfer Learnings weiter verfeinert. Dies geschieht durch das Trainieren auf einem spezifischen Datensatz von Zielmolekülen, wie beispielsweise AXL-Inhibitoren, wodurch das Modell seine generellen chemischen Kenntnisse nun gezielt auf die speziellen Strukturen dieser Zielmoleküle anwenden kann. Dadurch wird eine gezielte Spezialisierung des Modells erreicht, was zu einer erheblich verbesserten Leistungsfähigkeit bei der Generierung von Molekülen mit speziellen Eigenschaften führt [23].

Generative Modelle erlernen durch Trainingsbeispiele eine Wahrscheinlichkeitsverteilung. Die erlernte Verteilung wird durch *Sampling* reproduziert, wodurch anschließend neue Moleküle *de novo* generiert werden, die in ihrer Struktur und Eigenschaften mit den ursprünglichen Trainingsdaten vergleichbar sind [6]. Ein solches Modell, das speziell auf Moleküle trainiert wurde, entwickelt gewissermaßen ein „Verständnis“ dafür wie valide und sinnvolle Molekülstrukturen aussehen sollten [24]. Diese intuitive Fähigkeit erlaubt es dem Modell, neue chemische Verbindungen zu entwerfen, die sowohl strukturell plausibel als auch potenziell biologisch aktiv sind. Mithilfe der speziellen Architektur eines LSTM-Netzwerks und der dadurch resultierenden verbesserten Memory-Funktion ermöglicht es dem Netz, das relevante Wissen über mehrere Trainingsphasen hinweg zu speichern und wieder abzurufen, wodurch so ein mehrstufiges Feintuning erst möglich wird [21].

In Phase 3 wird schließlich ein Reinforcement Learning Ansatz angewandt, in dem das trainierte Grundmodell mit dem DeepSMILESPredictor kombiniert wird. Dadurch werden die durch die Wahrscheinlichkeitsverteilung generierten Ausgaben mit der Wirksamkeitsvorhersage des DeepSMILESPredictor bewertet. Dieser Mechanismus lenkt den Generator in Richtung der Produktion von spezifischen Molekülen, die den gewünschten Zielstrukturen entsprechen [25]. Dieser Prozess stellt sicher, dass die generierten Moleküle nicht nur strukturell sinnvoll, sondern auch funktional relevant sind.

Der DeepSMILESPredictor basiert auf einer *Multilayer-Perceptron*-Architektur, die darauf ausgelegt ist, Moleküle basierend auf ihrer potenziellen biologischen Aktivität zu klassifizieren. Dies funktioniert nach den Prinzipien der quantitativen Struktur-Wirkungs-Beziehungen. Die molekularen Deskriptoren bilden die Grundlage für die Vorhersage [15]. Es werden spezifische physikalisch-chemische Eigenschaften eines Moleküls quantifiziert. Die wiederum eine Vielzahl von molekularen Merkmalen abdecken, darunter hydrophobe Eigenschaften, elektronische Eigenschaften, geometrische Eigenschaften und topologische Indikatoren. ECFPs hingegen erfassen die lokale Umgebung jedes Atoms innerhalb eines definierten Radius, was zu einer kompakten Darstellung der molekularen Struktur führt [26]. Unter der Verwendung einer Kombination aus molekularen Deskriptoren und *Extended-Connectivity Morgan-Fingerprints* wird das Netz trainiert, um sowohl die strukturellen als auch die chemischen Eigenschaften der Moleküle zu erfassen. Der DeepSMILESPredictor ist nicht nur Teil des DeepSMILESGenerator, sondern dient auch als eigenständiges Werkzeug des Frameworks zum Screening der generierten Molekülbibliotheken auf potenzielle therapeutische Aktivitäten. Wodurch die Identifizierung vielversprechender Leitkandidaten für die experimentelle Validierung beschleunigt und erleichtert werden soll.

Dieser *in silico* Workflow bietet ein vielseitiges und leistungsfähiges Framework zur Entdeckung neuer Moleküle mit gezielten Eigenschaften. Durch die Kombination des DeepSMILESGenerator mit dem DeepSMILESPredictor werden nicht nur strukturell plausible, sondern auch funktional relevante Moleküle erzeugt, die gezielt auf spezifische biologische Zielstrukturen, wie die AXL-Rezeptor-Tyrosinkinase, abgestimmt sind. Dieser Ansatz demonstriert, wie die Integration moderner Deep Learning Techniken in den Prozess des Wirkstoffdesigns neue Wege eröffnet, um effizientere und zielgerichtete Therapeutika zu entwickeln. Die Ergebnisse dieser Arbeit verdeutlichen das Potenzial von computergestützten Methoden in der Arzneimittelentwicklung und unterstreichen ihre Bedeutung für die schnelle Reaktion auf globale Gesundheitskrisen wie COVID-19. Die vorgestellten Methoden könnten zukünftig nicht nur die Effizienz und Präzision in der Arzneimittelentwicklung steigern, sondern auch eine wichtige Rolle in der Entwicklung neuer, wirksamer Therapeutika spielen.

## Kapitel 2

# Theoretischer Hintergrund

Die Grundlage dieser Arbeit bildet ein fundierter theoretischer Rahmen, der den Einsatz moderner Machine Learning Methoden und Techniken erläutert. Die eingesetzten Verfahren werden detailliert betrachtet, um ihre Relevanz und Wirksamkeit im Kontext der nachfolgenden Forschung aufzuzeigen.

### 2.1 DeepSMILES Notation

Die präzise und verständliche Darstellung chemischer Strukturen ist in der Arzneistoffforschung von zentraler Bedeutung [27]. Für maschinenlesbare Repräsentationen, die in computergestützten Anwendungen wie dem maschinellen Lernen eingesetzt werden können, wurden spezielle Notationssysteme entwickelt. Eines der am häufigsten verwendete und am weitesten verbreiteten Systeme ist das *Simplified Molecular Input Line Entry System* [28]. Diese Notation, die in den 1980er Jahren von David Weininger entwickelt wurde, hat sich mittlerweile als *de facto* Standard für die einfache, lineare Textdarstellung chemischer Strukturen etabliert.

SMILES beschreibt die Struktur eines Moleküls durch eine Abfolge von Zeichen, die Atome, Bindungen, Verzweigungen, Ringschlüsse und chirale Zentren repräsentieren [28]. Dank dieser kompakten Darstellung können Moleküle leicht von Computern verarbeitet und in verschiedenen chemoinformatischen Anwendungen genutzt werden, was den Einsatz in der *in silico* Arzneistoffforschung enorm erleichtert.

**Atome:** – Ein Atom wird durch sein chemisches Symbol dargestellt. Zum Beispiel repräsentiert „C“ ein Kohlenstoffatom oder „O“ ein Sauerstoffatom. Atome der „organischen Teilmenge“ (B, C, N, O, P, S, F, Cl, Br, I) [28] können ohne Klammern geschrieben werden, solange die Anzahl der gebundenen Wasserstoffatome der üblichen Valenz entspricht. Bei anderen Elementen oder abweichenden Bindungen werden eckige Klammern verwendet.

**Bindungen:** – Einzelbindungen werden durch einfache Leerzeichen dargestellt, Doppelbindungen durch das Zeichen „=“ und Dreifachbindungen durch „#“.

**Verzweigungen:** – Verzweigungen werden durch Klammern dargestellt. Wenn ein Atom eine Seitenkette trägt, die nicht Teil der Hauptkette ist, wird diese Seitenkette in Klammern gesetzt. Zum Beispiel würde bei Isopropanol (CC(O)C) bedeuten, dass die Hydroxylgruppe „(O)“ als Verzweigung am zweiten Kohlenstoffatom hängt.

**Ringschlüsse:** – Ringschlüsse in einem Molekül werden durch Zahlen dargestellt, die an den Stellen des Rings erscheinen, an denen der Ring schließt. Zum Beispiel würde der Benzolring als „C1CCCCC1“ dargestellt werden.

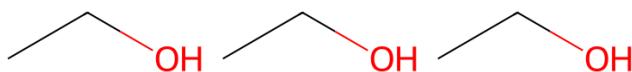
**Chirale Zentren:** – SMILES ermöglicht die Darstellung von Chiralität durch den Einsatz von „@“ Symbolen, um die absolute Konfiguration anzugeben.

Leider gibt es einige Nachteile der SMILES-Notation, welche es nicht zwangsläufig zu dem besten Werkzeug in der modernen chemischen Informatik und der Arzneistoffforschung macht [29]. Insbesondere in Kombination mit Methoden des maschinellen Lernens, bei denen große Datensätze von Molekülen präzise und effizient verarbeitet werden müssen.

Ein zentraler Nachteil der SMILES-Notation ist die Doppeldeutigkeit in der Darstellung von Molekülstrukturen. Aufgrund der verschiedenen möglichen Reihenfolgen und Darstellungsarten für dasselbe Molekül können mehrere SMILES-Repräsentationen existieren, was die Konsistenz und Vergleichbarkeit erschwert. Diese Variabilität kann zu Schwierigkeiten bei der Interpretation im Training von maschinellen Lernmodellen führen, da unterschiedliche SMILES-Repräsentationen für das gleiche Molekül unterschiedliche Ergebnisse erzeugen könnten.

**Beispiel 1:** Vergleich der verschiedenen SMILES-Repräsentationen für Ethanol ( $C_2H_6O$ )

1. SMILES: CCO
2. SMILES: CC(O)
3. SMILES: C(O)C



**ABBILDUNG 2.1: Visualisierung von Ethanol in verschiedenen SMILES-Darstellungen.** Die Abbildung zeigt Ethanol in den SMILES-Varianten CCO (links), CC(O) (mittig) und C(O)C (rechts). Jede dieser Darstellungen beschreibt dieselbe chemische Struktur von Ethanol, bestehend aus zwei Kohlenstoffatomen, fünf Wasserstoffatomen und einer Hydroxylgruppe. Dabei heben die einzelnen abgebildeten SMILES-Darstellungen verschiedene Priorisierungen der Bindungen und funktionellen Gruppen hervor, bleiben jedoch strukturell äquivalent.

In Abbildung 2.1 ist deutlich zu erkennen, dass die alle drei SMILES „CCO“, CC(O) und „C(O)C“, dasselbe Molekül, Ethanol, darstellen. Alle drei Darstellungen repräsentieren die gleiche chemische Verbindung, bestehend aus zwei Kohlenstoffatomen (C) und einer Hydroxylgruppe (-OH), die an ein Kohlenstoffatom gebunden ist. Die SMILES-Sequenzen unterscheiden sich lediglich im Informationsgehalt der Darstellung der Hydroxylgruppe. In der ersten SMILES-Repräsentation wird einfach die lineare Verbindung von zwei Kohlenstoffatomen und der Hydroxylgruppe angegeben, ohne explizite Informationen hervorzuheben. Im Gegensatz dazu zeigt die zweite SMILES-Repräsentation, dass die Hydroxylgruppe am zweiten Kohlenstoffatom als eine Verzweigung gebunden ist. Die dritte SMILES-Repräsentation zeigt, dass die Hydroxylgruppe als eine Verzweigung am ersten Kohlenstoffatom gebunden wird. Da jedoch nur zwei Kohlenstoffatome vorhanden sind, gibt es keine tatsächliche Verzweigung, und die Darstellungen sind äquivalent zueinander.

Ein weiteres Problem ist die Komplexität in der Darstellung von Ringschlüssen und chiralen Zentren. In der SMILES-Notation werden Ringe durch explizite Zahlenpaare gekennzeichnet, was in großen Molekülen oder bei mehreren Ringen leicht zu Fehlern und Verwechslungen führen kann oder zu schwer lesbaren Notationen führt. Besonders für Modelle, die auf maschinellem Lernen basieren, stellt diese Komplexität eine Herausforderung dar, da sie die Analyse und Verarbeitung der Daten erschwert.

Die DeepSMILES-Notation [22] wurde 2019 von Noel M. O’Boyle und Andrew Dalke als eine Alternative zu SMILES entwickelt, um einige Nachteile zu überwinden und die Anwendung der Notation für maschinelles Lernen zu verbessern. Eine der wichtigsten Verbesserungen der DeepSMILES-Notation ist die Reduktion der Doppeldeutigkeit, indem es eine konsistenter Darstellung für die Ring- und Verzweigungs-Strukturen bietet. Es eliminiert die Notwendigkeit für die explizite Nennung von Ziffern zur Markierung von Ringbindungen und vereinfacht so die Lesbarkeit. Dies erleichtert sowohl die Verarbeitung als auch die Analyse der Daten. Darauf hinaus ist die DeepSMILES-Notation robuster gegenüber syntaktischen Fehlern und ermöglicht eine kompaktere und weniger redundante Darstellung von Molekülen. Diese Eigenschaften machen es besonders nützlich für den Einsatz in maschinellen Lernanwendungen, wo eine konsistente und präzise Darstellung entscheidend ist, um die Modellgenauigkeit und Effizienz zu verbessern.

**Vereinfachung der Ringschlussnotation:** – In der SMILES-Notation werden Ringschlüsse durch explizite Zahlenpaare angegeben, die den Anfang und das Ende eines Rings markieren. Am Beispiel Benzol, markiert die „1“ hier den Beginn und das Ende des Rings. Im Gegensatz dazu vereinfacht DeepSMILES die Darstellung von Ringschlüssen erheblich, indem die Anzahl der zurückgelegten Schritte innerhalb der Sequenz gezählt wird, um den Ring zu schließen. Dies führt zu einer klareren und konsistenteren Darstellung, die einfacher zu interpretieren ist. Für das Beispiel Benzol markiert die „6“ am Ende der Sequenz, dass der Ring fünf Atome zurück geschlossen wird.

**Beispiel 2:** Vergleich der SMILES und DeepSMILES-Notation für Benzol ( $C_6H_6$ )

SMILES: C1CCCCC1  
DeepSMILES: CCCCCC6

**Verzweigungen von Ringschlüssen:** – Bei größeren oder komplexeren Molekülen, die aus mehreren verbundenen Ringen bestehen, können in der SMILES-Notation überlappende Ringnotationen nötig sein. Dies führt zu einer verzweigten und komplexen Darstellung. DeepSMILES erlaubt es, durch die vereinfachte Ringschlussnotation solche Probleme zu eliminieren.

**Beispiel 3:** Vergleich der SMILES und DeepSMILES-Notation für Bicyclisches Molekül

SMILES: CC1CCCO[C@]21CCCCO2  
DeepSMILES: CCCCCO[C@@]6CCCCO6

**Vereinfachung der Klammernotation:** – Verzweigungen werden durch Klammern dargestellt. Wenn ein Atom eine Seitenkette trägt, die nicht Teil der Hauptkette ist, wird diese Seitenkette in Klammern gesetzt. Zum Beispiel würde dies bei Isopropanol ( $CC(O)C$ ) bedeuten, dass die Hydroxylgruppe „(O)“ als Verzweigung am zweiten Kohlenstoffatom hängt.

**Beispiel 4:** Vergleich der SMILES und DeepSMILES-Notation für Isopropanol ( $C_3H_8O$ )

SMILES: CC(O)C  
DeepSMILES: CCO)C

Die DeepSMILES-Notation bietet eine überarbeitete Darstellung chemischer Strukturen, die sich insbesondere für maschinelle Lernanwendungen eignet. Aufgrund der zuvor beschriebenen Vorteile, wie der vereinfachten und konsistenteren Darstellung sowie der erhöhten Robustheit gegenüber syntaktischen Fehlern, stellt sie eine vielversprechende Wahl dar. Sie ermöglicht eine präzisere und effizientere Repräsentation chemischer Moleküle, was sie gerade im Kontext moderner maschinelaler Lernverfahren und in der Arzneistoffforschung besonders attraktiv macht. Aus diesen Gründen wird in dieser Arbeit die DeepSMILES-Notation für den vorgestellten *in silico* Workflow verwendet.

## 2.2 Long Short-Term Memory Networks

Nachdem die molekularen Strukturen in der DeepSMILES-Notation repräsentiert werden, stellt sich die Frage, wie diese Informationen in maschinelle Lernmodelle überführt werden können. Genau an dieser Stelle kommen *Recurrent Neural Networks* zum Einsatz.

RNNs bieten eine innovative Möglichkeit, mit sequenziellen Daten umzugehen, die in vielen Bereichen der modernen Informatik und Datenverarbeitung eine zentrale Rolle spielen [30]. Ihre besondere Stärke liegt in ihrer Fähigkeit, eine Abfolge von Daten nicht nur als isolierte Einheiten zu betrachten, sondern die Abhängigkeit und Beziehung zwischen den einzelnen Elementen einer Sequenz zu erfassen [24, 30]. Dies zu berücksichtigen ist essenziell, denn in einer Datenreihe können bestimmte Elemente vorangegangene Informationen beinhalten, die für die Interpretation und das Verständnis des Ganzen entscheidend sind.

RNNs sind speziell darauf ausgelegt, solche Daten zu verarbeiten, indem sie Informationen aus früheren Schritten in den aktuellen Berechnungsprozess einbeziehen. Diese Rückkopplungsschleifen innerhalb des Netzes ermöglichen es RNNs, „Erinnerungen“ an vorherige Datenpunkte aufrechtzuerhalten und so komplexe Abhängigkeiten in den Daten zu erfassen [24, 30]. Dies unterscheidet RNNs grundlegend von traditionellen neuronalen Netzen, bei denen jede Eingabe unabhängig voneinander betrachtet wird (Abbildung 2.2).

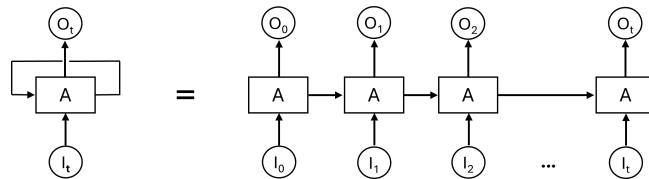


ABBILDUNG 2.2: RNN Architektur zur Verarbeitung von sequenziellen Daten nach [24]. Diese Abbildung illustriert die Funktionsweise eines Recurrent Neural Networks. RNNs sind darauf ausgelegt, Informationen aus vorherigen Schritten in den aktuellen Berechnungsprozess einzubeziehen. Das System betrachtet eine Eingabe  $I_t$  und gibt anschließend den resultierenden Wert  $O_t$  zurück.

Mit dieser Fähigkeit, die kontextuellen Informationen aus der Vergangenheit zu nutzen, ist ein RNN besonders nützlich für die Verarbeitung von Daten, bei denen die Reihenfolge und der Zusammenhang zwischen den Elementen entscheidend sind. Ein klassisches Beispiel ist die natürliche Sprache, wo die Bedeutung eines Wortes stark von den vorhergehenden Wörtern abhängt [24]. Ähnlich verhält es sich mit der DeepSMILES-Notation, bei der die Anordnung und Verknüpfung der Atome in der Sequenz die gesamte chemische Struktur widerspiegelt. Allerdings haben einfache RNNs Schwierigkeiten, Informationen über lange Sequenzen hinweg zu behalten. Dies wird als *Vanishing Gradient Problem* bezeichnet, bei dem die Fähigkeit des Netzes, sich an frühere Informationen zu erinnern, mit zunehmender Sequenzlänge abnimmt [21, 30]. Diese Einschränkung macht RNNs weniger effektiv für komplexe Aufgaben wie die Verarbeitung von langen DeepSMILES Sequenzen über mehrere Epochen hinweg.

Um das *Vanishing Gradient Problem* zu lösen, das bei einfachen RNNs auftritt, wurden *Long Short-Term Memory-Networks* entwickelt. Diese speziellen RNNs wurden 1997 von Hochreiter und Schmidhuber eingeführt und bieten eine effektive Methode, um Langzeitabhängigkeiten in sequenziellen Daten zu modellieren [21].

LSTMs nutzen sogenannte Speicherzellen, die es ihnen ermöglichen, wichtige Informationen über viele Zeitschritte hinweg zu speichern oder zu vergessen, je nachdem, was für die jeweilige Aufgabe relevant ist.

Diese Speicherzellen arbeiten mit drei zentralen Mechanismen, den sogenannten Gates: Ein *Input-Gate* entscheidet, welche Informationen in die Zelle aufgenommen werden, ein *Forget-Gate* bestimmt, welche Informationen gelöscht werden sollen, und ein *Output-Gate* regelt, welche Informationen an die nächste Einheit im Netzwerk weitergegeben werden. Dank dieser Gates können LSTMs frühere Informationen im Entscheidungsprozess berücksichtigen und somit aus der Vergangenheit lernen [21, 24]. Dies ist besonders wertvoll, wenn es um die Verarbeitung von Daten geht, die eine starke sequenzielle Abhängigkeit aufweisen, wie es Beispielsweise bei den chemischen Strukturen in der DeepSMILES-Notation der Fall ist.

Durch ihre spezielle Architektur ermöglichen LSTM-Netzwerke nicht nur eine verbesserte Verarbeitung von sequenziellen Daten, sondern sie können auch komplexe Muster über längere Sequenzen hinweg lernen und analysieren, ohne dass wichtige Informationen verloren gehen [21]. Mit der Analyse der Reihenfolge der Atome und Verbindungen aus der DeepSMILES-Notation kann das LSTM durch diese Fähigkeit intrinsische chemische Strukturen und Zusammenhänge ableiten. Anschließend, und das ist der entscheidende Faktor an LSTM-Netzwerken können nicht nur Vorhersagen getroffen werden, sondern auch neue Sequenzen generiert werden. Dies macht LSTMs zu einem wichtigen Werkzeug in dieser Studie, in der diese Informationen anschließend verwendet werden, um mit dem DeepSMILES-Generator neue valide Molekülstrukturen zu entwerfen, die auf den extrahierten chemischen Eigenschaften basieren.

## 2.3 Transfer Learning

Transfer Learning ist eine leistungsstarke Technik im Bereich des maschinellen Lernens, bei dem ein bereits vortrainiertes Modell für eine Spezialisierung weiter optimiert wird [31]. Im Gegensatz zu dem herkömmlichen Ansatz, bei dem Modelle von Grund auf neu trainiert werden, nutzt Transfer Learning das Wissen, dass ein Modell während seinem vorherigen Training meist auf einer großen, allgemeinen Datengrundlage erlangt hat, und überträgt diese auf spezifischere Aufgaben. Diese Methode erweist sich als besonders vorteilhaft, wenn für eine Aufgabe nur eine begrenzte Menge an Trainingsdaten zur Verfügung stehen.

Im Kontext eines *Recurrent Neural Networks* bedeutet Transfer Learning, dass die ersten Schichten des RNNs, welche die zuvor erlernten allgemeinen Datengrundlagen enthalten unverändert bleiben, während die letzten Schichten, die für die schlussendliche Generierung der Moleküle zuständig sind, neu trainiert werden [23, 32]. RNNs sind besonders gut darin, Muster in zeitlich geordneten Daten zu erkennen. Indem die vortrainierten Schichten beibehalten werden, die bereits grundlegende Merkmale erfasst haben, kann das Modell effizient auf neue, verwandte Aufgaben angepasst werden. Diese Vorgehensweise optimiert gezielt nur die letzten Schichten des Netzwerks, wodurch das Modell auf spezifische Anforderungen abgestimmt wird, ohne dass die bereits erlernten Informationen aus den früheren Schichten verloren gehen. Dies führt in vielen Fällen zu einer Verbesserung der Performance, da

das Modell bereits wichtige zeitliche Zusammenhänge gelernt hat, die für die neue Aufgabe von Relevanz sein können [23].

Transfer Learning ist also besonders nützlich in Bereichen, in denen Daten rar oder kostspielig zu sammeln sind [31]. Es ist deshalb ein vielversprechender Ansatz für die Identifikation neuer Moleküle in der *in silico* Wirkstoffforschung. Dabei wird das LSTM-Modell, statt auf einem zu kleinen Datensatz zu trainieren und Overfitting zu riskieren, zuerst auf einen großen allgemeinen Datensatz vortrainiert und anschließend auf die eigentliche Aufgabe, also die Generierung von speziellen AXL-Kinase Inhibitoren, feinabgestimmt. Diese Vorgehensweise kann die Genauigkeit und Effizienz des LSTM-Modells signifikant verbessern, da das Modell durch das „Vortraining“ bereits über grundlegende Kenntnisse der chemischen Strukturen und Muster verfügt. Aus dieser Hinsicht bietet Transfer Learning die Möglichkeit trotz limitierter Daten verbesserte Ergebnisse zu erzielen und kann dem DeepSMILESGenerator helfen sich auf die spezialisierte Aufgabe der AXL-Kinase-Inhibitoren Generierung anzupassen.

## 2.4 Fine-Tuning Reinforcement Learning

Um den auf DeepSMILES trainierten DeepSMILESGenerator in einem Fine-Tuning Prozess noch besser auf die Generierung spezifischer Moleküle abzustimmen, wird ein Reinforcement Learning Ansatz gewählt.

Reinforcement Learning ist eine Form des maschinellen Lernens, bei dem ein lernendes System durch Interaktion mit seiner Umgebung lernt Entscheidungen zu treffen, die ihn zu einem optimalen Ergebnis führen [25, 33]. Durch das Ausprobieren verschiedener Aktionen erhält das System Rückmeldungen in Form eines Belohnungssystems. Nützliche Aktionen, die ihn seinem Ziel näherbringen, werden mit positiven Belohnungen verstärkt, während fehlerhafte Handlungen, die ihn vom gewünschten Ergebnis ablenken, negative Belohnungen oder Strafen zur Folge haben. Dieser iterative Lernprozess ermöglicht es dem System, mit der Zeit immer bessere Strategien zu entwickeln.

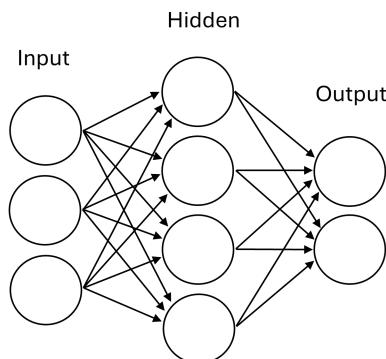
Im Gegensatz zu anderen überwachten Lernverfahren, bei denen das Modell auf der Grundlage eines festen Datensatzes mit vorgegebenen Eingaben und dazugehörigen Ausgaben trainiert wird, arbeitet Reinforcement Learning wesentlich dynamischer. Das System beginnt ohne Wissen über die Umgebung und muss durch Erfolg und Misserfolg lernen, welche Aktionen die besten Ergebnisse liefern. Diese Lernmethode eignet sich besonders für Szenarien, in denen es unmöglich ist, alle potenziellen Ergebnisse im Voraus zu modellieren [25]. Was sie ideal für komplexe Aufgaben wie das Moleküldesign in der Wirkstoffforschung macht. Ein zentraler Bestandteil von Reinforcement Learning ist das Konzept der *Exploration* und *Exploitation* [33]. Das lernende System muss die Balance finden zwischen dem Erkunden neuer Aktionen (*Exploration*), die potenziell bessere Belohnungen bieten könnten, und dem Ausnutzen bereits bekannter Aktionen (*Exploitation*), die bereits als erfolgreich identifiziert wurden. Diese duale Strategie ist entscheidend, um sicherzustellen, dass das System sowohl kurzfristige Erfolge maximiert als auch langfristig optimale Ergebnisse erzielt.

Im Kontext des Fine-Tunings spielt Reinforcement Learning eine entscheidende Rolle. Nachdem das LSTM-Grundmodell auf bekannte DeepSMILES Daten trainiert wurde, kann Reinforcement Learning verwendet werden, um das Modell weiter zu verfeinern und den DeepSMILESGenerator auf die Generierung neuer, potenziell wirksamer Moleküle zu spezialisieren. Hierbei bewertet das Modell die durch das LSTM erzeugten Moleküle basierend auf ihren strukturellen, chemischen und biologischen Eigenschaften, den sogenannten Deskriptoren, und passt dessen Strategie an, um Moleküle zu generieren, die den gewünschten Kriterien besser entsprechen. Die Bewertung dieser Deskriptoren, sowie die Verteilung der Belohnungen während dem Reinforcement Learning, erfolgt mit dem DeepSMILESPredictor.

## 2.5 Multilayer Perceptron Klassifikator

Die Analyse und Bewertung der generierten Moleküle erfolgt durch den Deep-SMILES Predictor, einem speziell entwickelten Modell für Klassifikationsaufgaben. Der Klassifikator besteht dabei aus einem Zweiklassen *Multilayer Perceptron*, das darauf spezialisiert wird, zu entscheiden, ob es sich bei einem generierten Molekül um ein Zielmolekül oder ein beliebiges nicht-zielgerichtetes Molekül handelt.

Ein *Multilayer Perceptron* ist ein grundlegendes Modell des maschinellen Lernens, das in der Lage ist, komplexe nichtlineare Beziehungen in Daten zu erkennen und zu modellieren [34,35]. Inspiriert vom Aufbau des menschlichen Gehirns besteht ein MLP aus mehreren Schichten von künstlichen Neuronen, die miteinander vernetzt sind [34]. Ähnlich wie das menschliche Gehirn, das Informationen über ein Netz von Neuronen verarbeitet und lernt assoziative Muster zu erkennen, nutzt das MLP diese neuronalen Strukturen um Eingabedaten zu analysieren und Muster zu erlernen.



**ABBILDUNG 2.3: Schematische Darstellung eines MLPs nach [36].**  
Das Multilayer Perceptron besteht aus mehreren Schichten: einer Input Layer, einer oder mehreren Hidden Layers und einer Output Layer. Die Neuronen in jeder Schicht sind vollständig miteinander verbunden. Diese Verbindungen und ihre Gewichtungen bestimmen wie die Eingabedaten durch das Netzwerk fließen, und anschließend zur Klassifikation verarbeitet werden.

Das Modell ist in mehrere Schichten unterteilt (Abbildung 2.3) [36]. Die linke Schicht wird als Eingabeschicht (*Input Layer*) bezeichnet, und die Neuronen innerhalb dieser Schicht als Eingabeneuronen. Diese Eingabeschicht nimmt die Rohdaten, also die Merkmale der Moleküle als Eingabeneuronen für das Netz. Die rechte Schicht oder Ausgabeschicht (*Output Layer*) enthält die Ausgabeneuronen, also in diesem Fall die zwei Klassifikationsmöglichkeiten, in die das neuronale Netz die generierten Moleküle klassifiziert. Die mittlere Schicht wird als versteckte Schicht (*Hidden Layer*) bezeichnet. Der Begriff „versteckte Schicht“ klingt auf den ersten Moment etwas geheimnisvoll, aber hat keine tiefgründigere philosophische oder mathematische Bedeutung, es bedeutet nur das die Enthaltenen Neuronen weder zur Ausgabe- oder Eingabeschicht gehören [35]. Ein MLP kann dabei mehrere versteckte Schichten haben, je mehr versteckte Schichten ein MLP hat, desto tiefer wird das Netzwerk, was ihm erlaubt, abstraktere Merkmale und tiefere Zusammenhänge in den Daten zu erfassen [36].

Durch die Anwendung von Aktivierungsfunktionen wie Sigmoid, ReLU oder Tanh in diesen versteckten Schichten können MLPs auch nichtlineare Transformationen durchführen, die entscheidend sind, um komplexe Abhängigkeiten und Muster in den Daten zu verstehen [34]. Dies bedeutet, dass ein MLP in der Lage ist, von einfachen linearen Modellen abzuweichen und eine vielschichtige Analyse durchzuführen, die für Aufgaben wie Bilderkennung, Sprachverarbeitung und auch die Analyse chemischer Strukturen unerlässlich ist [37].

## 2.6 Molekulare Deskriptoren

In der *in silico* Wirkstoffforschung spielen molekulare Deskriptoren eine entscheidende Rolle [15]. Diese Deskriptoren sind numerische Werte, die spezifische Eigenschaften und Merkmale von Molekülen quantifizieren und deren chemische Struktur sowie physikalisch-chemische Eigenschaften beschreiben. Durch ihre Verwendung als Eingabeparameter in maschinellen Lernmodellen wie dem MLP, ermöglichen sie die Analyse chemischer Daten und Vorhersagen über die biologische Aktivität zu treffen.

Für die Klassifikation der generierten Moleküle wird der Ansatz der *Quantitative Structure-Activity Relationship* genutzt. QSAR ist ein wissenschaftlicher Ansatz in der Chemie und Biologie, der darauf abzielt, die Beziehung zwischen der chemischen Struktur eines Moleküls und seiner biologischen Aktivität zu quantifizieren [15] [38]. Diese Methode ist in der Arzneimittelforschung weit verbreitet, da sie es ermöglicht, die Wirksamkeit potenzieller Wirkstoffe vorherzusagen und zu optimieren. Die Annahme hinter QSAR-Analysen ist, dass Moleküle mit ähnlichen Strukturen ähnliche biologische Aktivitäten aufweisen sollten. Daher werden mathematische und statistische Methoden eingesetzt, um Beziehungen zwischen molekularen Deskriptoren und biologischer Aktivität herzustellen.

Die Auswahl und Kombination geeigneter molekularer Deskriptoren ist entscheidend für die Leistungsfähigkeit des MLP-Modells. Deskriptoren bieten eine umfassende Möglichkeit die vielfältigen Eigenschaften von Molekülen abzubilden.

Physikochemische Deskriptoren berücksichtigen wichtige Eigenschaften wie Polarität, Hydrophobizität und Molekulargewicht, die die Reaktivität und Löslichkeit beeinflussen [15,39]. Topologische Deskriptoren erfassen die Struktur und Vernetzung, beispielsweise durch die Analyse von Ringgrößen und Verzweigungen [15, 40]. Geometrische Deskriptoren konzentrieren sich auf die dreidimensionale Form und Flexibilität eines Moleküls [15]. Elektronische Deskriptoren hingegen befassen sich mit der Verteilung von Elektronen und der Polarisierbarkeit, die entscheidend für molekulare Wechselwirkungen sind [15]. Im DeepSMILES Predictor wurden nachfolgende Deskriptoren als Eingabe definiert, um eine umfassende Analyse und präzise Vorhersage der biologischen Aktivität von Molekülen zu ermöglichen.

**AlogP [41]** – Ein Maß für die Lipophilie eines Moleküls, das seine Fähigkeit darstellt, sich in Fetten, Ölen Lipiden und unpolaren Lösungsmitteln zu lösen. AlogP ist wichtig für die Vorhersage der Bioverfügbarkeit und Verteilung in Arzneimitteln.

**Polare Oberfläche (Polar Surface Area, PSA) [42]** – Beschreibt die Summe der Oberflächenbereiche polarer Atome wie Sauerstoff, Stickstoff und der daran gebundenen Wasserstoffatome. Diese Fläche bezieht sich in der Regel auf die van-der-Waals-Oberfläche des Moleküls. PSA beeinflusst die Absorption und den Transport von Medikamenten durch biologische Membranen.

**TPSA (Topologische polare Oberfläche) [42]** – Beschreibt die Summe der Oberflächenbereiche polarer Atome wie Sauerstoff, Stickstoff und der daran gebundenen Wasserstoffatome. Im Gegensatz zur PSA, die oft auf der dreidimensionalen Geometrie eines Moleküls basiert, wird die TPSA basierend auf der zweidimensionalen Struktur berechnet. Dadurch können mit diesem Ansatz große Molekülmengen in vergleichsweise kürzerer Zeit gescreent werden. Ein hoher TPSA-Wert kann auf eine schlechte Membranpermeabilität hinweisen.

**HBA (Hydrogen Bond Acceptor) [43]** – Definiert die Anzahl der Wasserstoffbrücken-akzeptorstellen in einem Molekül, was dessen Löslichkeit und Interaktion mit biologischen Zielstrukturen beeinflusst.

**HBD (Hydrogen Bond Donor) [43]** – Definiert die Anzahl der Wasserstoffbrücken-donorstellen in einem Molekül, was die Löslichkeit und biologischen Interaktionen beeinflusst.

**Bioaktivitäten (pKi-Wert) [44, 45]** – Ein Maß, um die Bindungsaffinität von Molekülen zu einem biologischen Ziel zu beschreiben. Ausgedrückt als negativer Logarithmus der Hemmkonstante (Ki). PKi wird dazu häufig in Verbindung mit anderen molekularen Deskriptoren genutzt, um die Struktur-Aktivitäts-Beziehungen von Molekülen zu quantifizieren. Dabei hilft der pKi-Wert Unterschiede in der Aktivität verschiedener Molekülstrukturen zu erklären, insbesondere in Bezug auf ihre Bindungseigenschaften an spezifische Rezeptoren oder Enzyme.

**Chi0 [46,47]** – Auch bekannt als der erste molekulare Konnektivitätsindex. Er ist ein topologischer Deskriptor, der die Struktur eines Moleküls durch seine Verzweigung und Verbindung beschreibt. Er basiert auf der Anzahl und Art der Bindungen sowie den Atomtypen innerhalb eines Moleküls. Chi0 wird oft verwendet, um die

Konnektivität oder Dichte von Molekülen zu analysieren, was dabei hilft, verschiedene chemische Eigenschaften wie Reaktivität, Löslichkeit oder biologische Aktivität zu bewerten.

**Chi1 [46, 47]** – Erweitert "Chi0", indem er die Anzahl der Verbindungen erster Ordnung im Molekül berücksichtigt, z.B. direkte Bindungen zwischen Kohlenstoffatomen. Er liefert eine detailliertere Analyse der lokalen Verbindungen im Molekül.

**Chi0n [46, 47]** – Erweitert „Chi0“, konzentriert sich jedoch speziell auf Stickstoffatome. Er bewertet die Konnektivität und Struktur eines Moleküls unter Berücksichtigung der Position und Bindungen von Stickstoffatomen.

**Chi0v [46, 47]** – Erweitert „Chi0“, indem er die Valenzelektronen der Atome im Molekül berücksichtigt. Er gibt Einblicke in die Elektronendichte und die chemische Reaktivität eines Moleküls.

**Chi1n [46, 47]** – Erweitert „Chi0“, jedoch konzentriert er sich auf Stickstoffatome in Bindungen erster Ordnung. Er bietet eine detaillierte Analyse der lokalen Struktur und der Bindungsmuster, die Stickstoffatome betreffen.

**Chi1v [46, 47]** – Erweitert „Chi0“, mit einem Fokus auf die Valenzelektronen in Bindung erster Ordnung. Dieser Deskriptor hilft dabei die Elektronendichte in spezifischen Bereichen des Moleküls zu analysieren.

**Kappa1 [46, 47]** – Der erste Kappa-Shape-Index ist ein molekularer Deskriptor, der die Flexibilität eines Moleküls quantifiziert. Er basiert auf der Anzahl der Bindungen und deren Anordnung im Molekül. Ein niedriger Kappa1-Wert deutet auf ein Molekül mit einer linearen oder weniger verzweigten Struktur hin, was auf eine höhere Flexibilität hindeutet. Ein höherer Kappa1-Wert weist auf eine komplexere, stärker verzweigte Struktur hin, die in der Regel weniger flexibel ist. Diese Kennzahl wird häufig verwendet, um die dreidimensionale Gestalt und Dynamik von Molekülen zu bewerten.

**Kappa2 [46, 47]** – Erweitert „Kappa1“ und bewertet die Flexibilität eines Moleküls, indem er die Anzahl der möglichen Konfigurationen für Ringe und Verzweigungen berücksichtigt. Außerdem gibt er Aufschluss darüber, wie das Molekül sich in verschiedenen Umgebungen verhalten könnte.

**Kappa3 [46, 47]** – Erweitert „Kappa1“, bewertet die dreidimensionale Struktur eines Moleküls, geht aber noch einen Schritt weiter, indem er die gesamte Form und mögliche Konformationen des Moleküls in einem dreidimensionalen Raum beschreibt. Dies hilft bei der Beurteilung der räumlichen Komplexität und der möglichen Wechselwirkungen des Moleküls mit anderen Strukturen.

**HallKierAlpha [46, 47]** – Ein Maß für die Verzweigung und Komplexität der Struktur eines Moleküls. Ein hoher „HallKierAlpha“-Wert weist auf ein Molekül mit vielen Verzweigungen und einer komplexen Struktur hin, was auf eine höhere chemische Reaktivität oder spezifische biologische Wechselwirkungen hinweisen könnte.

**MolLogP [48]** – Ein Maß der den Logarithmus des Verteilungskoeffizienten (P) eines Moleküls zwischen Wasser und Octanol berechnet. Beschreibt die Lipophilie eines Moleküls und gibt an, wie stark es sich in einer hydrophoben im Vergleich zu einer hydrophilen Umgebung auflöst. Ein hoher MolLogP-Wert deutet auf eine hohe Lipophilie hin. Dies ist entscheidend für die Fähigkeit eines Moleküls die Zellmembranen zu passieren.

**PEOE\_VSA1 bis PEOE\_VSA14 [46,49]** – Eine Reihe von Deskriptoren unterteilt die van-der-Waals-Oberfläche eines Moleküls in verschiedene Bereiche, die durch unterschiedliche Partialladungen geprägt sind. Diese Differenzierung ermöglicht es, die molekularen Interaktionen genauer zu verstehen, da sie aufzeigt, wie die einzelnen Bereiche der Oberfläche mit anderen Molekülen oder biologischen Zielstrukturen in Wechselwirkung treten können.

**Molecular Weight [43]** – Ein Maß für die gesamte Masse eines Moleküls.

**NumRotatableBonds [50]** – Die Anzahl der rotierbaren Bindungen in einem Molekül. Beschreibt wie viele Einfachbindungen vorhanden sind, die sich um ihre Achse frei drehen können. Diese Eigenschaft beeinflusst die Flexibilität des Moleküls. Moleküle mit mehr rotierbaren Bindungen haben eine höhere Flexibilität.

**NumAromaticRings [51]** – Die Anzahl aromatischer (ringförmiger) Strukturen in einem Molekül. Aromatische Ringe tragen zur Stabilität eines Moleküls bei und spielen eine wichtige Rolle in der Arzneimittelforschung, da sie oft zur Verbesserung der Bindungseigenschaften und der pharmakologischen Aktivität eingesetzt werden.

**FractionCSP3 [52]** – Der Anteil  $sp^3$ -hybridisierter Kohlenstoffatome in einem Molekül beschreibt, wie viele Kohlenstoffatome mit vier Einfachbindungen in einem Molekül vorkommen. Dieser Deskriptor gibt Aufschluss über den Grad der Sättigung und die dreidimensionale Struktur eines Moleküls. Moleküle mit einem höheren Anteil an  $sp^3$ -Kohlenstoffen sind weniger „flach“ und haben oft eine verbesserte Löslichkeit und pharmakokinetische Eigenschaften.

**HeavyAtomCount [53, 54]** – Ein Maß für die Anzahl der Nicht-Wasserstoff-Atome in einem Molekül, wie Kohlenstoff, Stickstoff, Sauerstoff. Diese Zahl gibt Aufschluss über die molekulare Größe und Komplexität des Moleküls.

**NHOHCount [53,54]** – Zählt die Gesamtzahl der Stickstoff-, Sauerstoff- und Hydroxylgruppen (OH) in einem Molekül. Beschreibt die Polarität eines Moleküls und seine Fähigkeit, Wasserstoffbrückenbindungen zu bilden.

**NOCount [53,54]** – Die Anzahl der Stickstoff- und Sauerstoffatome in einem Molekül. Beschreibt die Fähigkeit zur Wasserstoffbrückenbildung sowie die chemische Reaktivität. Diese Atome sind oft an polaren und reaktiven Stellen im Molekül zu finden und spielen daher eine wesentliche Rolle in der Molekül-Bindungsdynamik mit biologischen Rezeptoren oder Enzymen.

**NumHeteroatoms [53, 54]** – Beschreibt die Anzahl der Heteroatome in einem Molekül, also die Atome, die weder Kohlenstoff noch Wasserstoff sind. Diese Heteroatome beeinflussen maßgeblich die Reaktivität eines Moleküls.

**NumRadicalElectrons [53, 54]** – Ein Maß für die Anzahl der ungepaarten Elektronen in einem Molekül, bekannt als Radikale. Moleküle mit ungepaarten Elektronen neigen dazu, schnell zu reagieren und sind oft instabil.

**NumValenceElectrons [53, 54]** – Dieser Deskriptor beschreibt die Anzahl der Valenzelektronen, also der Elektronen in der äußersten Schale der Atome eines Moleküls. Valenzelektronen bestimmen das chemische Bindungsverhalten und die Reaktivität eines Moleküls, da sie an der Bildung von Bindungen zwischen Atomen beteiligt sind.

**RingCount [53, 54]** – Die Anzahl der Ringstrukturen, also zyklischen Verbindungen in einem Molekül. Diese Ringe tragen zur molekularen Stabilität bei und beeinflussen die Wechselwirkungen des Moleküls mit biologischen Zielen.

**BalabanJ [46, 55]** – Beschreibt die Struktur eines Moleküls auf einer graphentheoretischen Ebene. Er berücksichtigt die Gesamtform und Verzweigung eines Moleküls, indem er die Distanzen zwischen den Atomen in einem Molekülgraphen berechnet. Der Balaban-J-Index ist besonders nützlich, um die Komplexität und Verzweigung eines Moleküls zu quantifizieren und wird oft in der chemischen Informatik verwendet, um Struktur-Aktivitäts-Beziehungen und molekulare Ähnlichkeiten zu bewerten.

**BertzCT [56]** – Der Bertz-Complexity-Index ist ein Maß der die strukturelle Komplexität eines Moleküls quantitativ erfasst. Er berücksichtigt verschiedene Aspekte der molekularen Struktur, einschließlich der Anzahl der Atome, Bindungen, Ringe und Verzweigungen. Ein höherer Bertz-Index zeigt eine größere strukturelle Vielfalt und Verzweigung des Moleküls an.

## 2.7 Extended-Connectivity Morgan-Fingerprints

Neben den klassischen molekularen Deskriptoren gibt es spezielle Methoden, um die Struktur und Eigenschaften von Molekülen zu analysieren. Eine Möglichkeit sind molekulare Fingerabdrücke. Genauer gesagt *Extended-Connectivity Morgan-Fingerprints* oder auch ECFPs genannt. Diese sind eine besonders effektive Art von molekularen Fingerabdrücken, die speziell für die Erkennung von Ähnlichkeiten zwischen Molekülen entwickelt wurden [26].

Im Gegensatz zu einfachen Deskriptoren erfassen ECFPs die atomare Umgebung innerhalb eines Moleküls durch die Analyse zirkulärer Substrukturen um jedes Atom (Abbildung 2.4) [26]. Diese Substrukturen werden mit einem variablen Radius betrachtet, der alle chemischen Nachbarschaften eines Atoms einbezieht, und anschließend in einen binären Vektor umgewandelt wird. Ihre Effektivität liegt in der Fähigkeit, sowohl globale als auch lokale strukturelle Informationen zu erfassen. Dieser Prozess, der auch als "extended-connectivity" bezeichnet wird, bietet eine

detaillierte Darstellung der molekularen Topologie und ermöglicht eine hochpräzise Berechnung der strukturellen Ähnlichkeit zwischen Molekülen. Dies macht sie zu einem leistungsstarken Werkzeug in der Wirkstoffforschung, insbesondere für das virtuelle Screening und die Identifizierung von potenziellen Leitstrukturen [26].

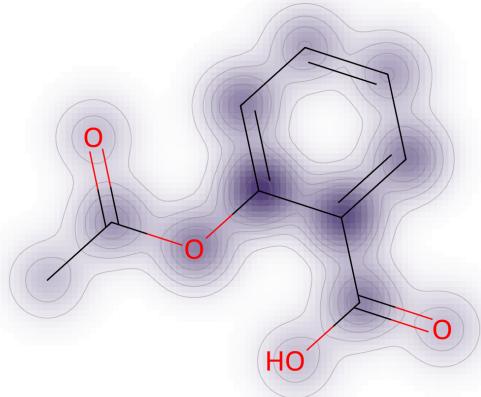


ABBILDUNG 2.4: **Visualisierung des ECFP von Aspirin.** Der ECFP wurde mit einem Radius von 2 und einer Bitlänge von 2048 generiert. Diese zirkulären Fingerabdrücke erfassen strukturelle Merkmale der Molekülumgebung indem sie Informationen über atomare Nachbarschaften innerhalb eines definierten Radius sammeln, um die chemische Struktur von Aspirin effizient darzustellen.

Durch die Vielfalt und Differenziertheit der beschriebenen Deskriptoren soll dem MLP-Modell ermöglicht werden, komplexe, nichtlineare Zusammenhänge zwischen der chemischen Struktur eines Moleküls und dessen biologischer Aktivität erfassen zu können. Mithilfe der einzelnen Schichten und Neuronen des MLP können diese Beziehungen modelliert werden, sodass das Modell nicht nur in der Lage ist die biologische Aktivität eines Moleküls vorherzusagen, sondern auch helfen neue, potenzielle bioaktive Moleküle zu identifizieren.

## Kapitel 3

# Materialien und Methoden

Die Entwicklung des DeepSMILES Predictor und DeepSMILESGenerator folgt einem mehrstufigen Trainingsansatz, der auf moderner Machine Learning Methoden aufbaut. Ein vielschichtiger Workflow wurde etabliert, um präzise Vorhersagen und die Generierung chemischer Strukturen zu ermöglichen. Wesentliche Aspekte sind dabei die Beschaffung und Aufbereitung der Daten, das mehrstufige Training der Modelle sowie die anschließende Feinabstimmung. Diese Grundlagen bilden die Basis für die späteren Experimente und Analysen, die in den folgenden Kapiteln detailliert untersucht werden.

### 3.1 PyTorch & CUDA

Für die Entwicklung des in dieser Studie vorgestellten *in silico* Workflows wurde Python 3.9.8 in Kombination mit der Bibliothek PyTorch 2.2.2+cu121 als zentrales Framework gewählt.

PyTorch stellt eine der führenden Open-Source-Bibliotheken für maschinelles Lernen dar, welche in Python entwickelt wurde. Dabei eignet es sich insbesondere für die Arbeit mit tiefen neuronalen Netzen und zeichnet sich durch seine Flexibilität sowie Benutzerfreundlichkeit aus [57]. Die Entscheidung für PyTorch wurde getroffen, da diese Bibliothek mehrere Vorteile bietet, die für die spezifischen Anforderungen dieses Projektes von besonderer Relevanz sind.

Im Rahmen der Entwicklung des DeepSMILESGenerator und DeepSMILES Predictor wurde PyTorch als leistungsstarkes Werkzeug für den Aufbau und das Training komplexer neuronaler Netzwerke ausgewählt. Die Bibliothek unterstützt eine Vielzahl von Architekturen für neuronale Netze, darunter auch die für unsere *in silico* Workflow relevanten Long Short-Term Memory und Multi-Layer Perceptron Netzwerke. Ein weiteres entscheidendes Merkmal von PyTorch ist die Unterstützung von GPU-Beschleunigung durch CUDA [58]. Dadurch wird das Training großer Modelle erheblich beschleunigt und die Verarbeitung großer Datenmengen ermöglicht, wie sie in der molekularen Wirkstoffforschung benötigt werden. Die Kombination von PyTorch mit der Programmiersprache Python erlaubt die Entwicklung leistungsfähiger und effizienter maschineller Lernmodelle, die sowohl flexibel als auch skalierbar sind, und dadurch den Anforderungen der modernen Wirkstoffforschung gerecht werden [57].

## 3.2 ChEMBL & PubChem Data

Für das Training des DeepSMILESGenerator und des DeepSMILES Predictor wurden generelle Moleküldaten sogenannte „Compounds“ und spezifische Moleküldaten von AXL-Kinase-Inhibitoren als Trainingsdatensatz benötigt. Dazu wurden die zwei großen Moleküldatenbanken ChEMBL und PubChem durchsucht.

### 3.2.1 Compound-Datensatz

In der öffentlich zugänglichen ChEMBL-Datenbank [59] wurden auf der Suche nach generellen Moleküldaten insgesamt 1.874.870 Compounds, gefiltert nach small Molecules, identifiziert. Dieser Datensatz liegt in Form einer Textdatei vor, in welcher die Verbindungen in der SMILES-Notation gespeichert sind. Da im Rahmen dieser Studie jedoch mit der DeepSMILES-Notation gearbeitet wird, ist es notwendig, alle Inputdaten zunächst von der SMILES- in die DeepSMILES-Notation zu konvertieren. Der so umgewandelte Compound-Datensatz kann anschließend verwendet werden, um den DeepSMILESGenerator zu trainieren das Vokabular sowie die Grundstrukturen der Moleküle zu erlernen.

### 3.2.2 Vokabular

Um aus 1.874.870 Compounds nur die relevantesten Tokens für das anschließende DeepSMILESGenerator Vokabular zu filtern, wurde ein Skript geschrieben, dass zuerst alle SMILES-Strukturen in DeepSMILES umwandelt und dann die Häufigkeit der vorkommenden Notationen aufaddiert und der Rangfolge absteigend ausgibt (Anhang A). Es wurde eine Grenze von mindestens 500 Erscheinungen gewählt, um einen im Datensatz enthaltenen Token, in das Vokabular mitaufzunehmen. Diese Grenze wurde gesetzt, um das Vokabular nicht unnötig groß und kompliziert zu gestalten. Das Ergebnis ergab ein 70 Token langes Vokabular, welches außerdem noch ein Start-, End-, Padding-, und Undefined-Token enthielt. Ein Start-Token () und End-Token (\$) für die Markierung des Anfangs und des Endes einer Sequenz. Das Padding-Token (PAD) für das Auffüllen von nicht gleichlangen Batches und ein Undefined-Token (UNK) für die Zeichen der Notation, die nicht im Vokabular enthalten sind und dadurch nicht erkannt werden (Abbildung 3.1).

### 3.2.3 AXL-Kinase-Inhibitor-Datensatz

Für die weiteren Phasen des Trainings werden bereits vorhandene AXL-Kinase-Inhibitoren benötigt. Dazu wurde erneut die ChEMBL-Datenbank durchsucht. Dieses Mal wurde spezifisch nach „AXL-Kinase-Inhibitoren“ gefiltert und dabei 518 Treffer gefunden. Zusätzlich wurde hierbei die PubChem-Datenbank des National Institutes of Health durchsucht [60] und die dabei gefundenen 6764 Treffer nach einem AC50-Wert  $\leq 1$  gefiltert. AC50 steht für die Konzentration eines Inhibitors, die erforderlich ist, um die Aktivität eines bestimmten biologischen Ziels um 50% zu reduzieren [61]. Ein niedriger AC50-Wert deutet darauf hin, dass nur eine sehr geringe Menge des Inhibitors notwendig ist, um eine signifikante hemmende Wirkung

Token	Wert	Token	Wert
'PAD'	0	'S'	40
'UNK'	1	'F'	41
'~'	2	'P'	42
'\$'	3	'T'	43
'3'	4	'B'	44
'4'	5	'Br'	45
'5'	6	'Cl'	46
'6'	7	'[C@]'	47
'7'	8	'[C@H]'	48
'8'	9	'[C@@H]'	49
'9'	10	'[C@@]'	50
'%10'	11	'[nH]'	51
'%11'	12	'[O-]'	52
'%12'	13	'[N+]'	53
'%13'	14	'[n+]'	54
'%14'	15	'[Na+]'	55
'%15'	16	'[S+]'	56
'%16'	17	'[Br-]'	57
'%17'	18	'[Cl-]'	58
'%18'	19	'[I-]'	59
'%19'	20	'[N-]'	60
'%20'	21	'[Si]'	61
'%21'	22	'[2H]'	62
'%22'	23	'[K+]'	63
'%23'	24	'[Se]'	64
'%24'	25	'[P+]'	65
')'	26	'[C-]'	66
'='	27	'[se]'	67
'#'	28	'[Cl+3]'	68
'.'	29	'[Li+]'	69
'_'	30		
'/'	31		
'\'	32		
'n'	33		
'o'	34		
'c'	35		
's'	36		
'N'	37		
'O'	38		
'C'	39		

ABBILDUNG 3.1: **Darstellung des Vokabulars.** Das Vokabular, welches für den DeepSMILESGenerator verwendet wurde. Es umfasst 70 Token, die nach der Häufigkeit ihres Vorkommens in einem Datensatz von über 1,8 Millionen Verbindungen basieren. Zusätzlich sind spezielle Token enthalten, wie ein Start- und End-Token für die Sequenzmarkierung, ein Padding-Token für die Batch-Verarbeitung, sowie ein Undefined-Token für unbekannte Zeichen.

zu erzielen. Die anschließend übriggebliebenen 4046 AXL-Kinase-Inhibitoren wurden zusammen mit den 518 Treffer aus der ChEMBL-Datenbank als AXL-Kinase-Inhibitor-Datensatz gespeichert. Anschließend wurde dieser 4564 Treffer große Datensatz für das Transfer Learning verwendet, um dem DeepSMILESGenerator beizubringen wie AXL-Kinase-Inhibitoren aufgebaut sind.

Für den DeepSMILESPredictor wird der AXL-Kinase-Inhibitor-Datensatz benötigt, um die Klassifizierung zwischen Zielmolekülen und normalen Compound-Molekülen zu erlernen. Für diese Aufgabe wurde eine weitere, kleinere Version des zuvor erstellten Compound-Datensatzes vorbereitet, um kein Klassenungleichgewicht beim Training zu erzeugen. Das Verhältnis wurde auf ungefähr 1:2 AXL-Kinase-Inhibitoren und normalen Compound-Molekülen gesetzt, was bedeutet, dass aus dem Compound-Datensatz die ersten 10812 Zeilen in einen kleinen Compound-Datensatz kopiert wurde. Mit Methoden wie *ClassWeights* und *Oversampling* wurden die unterschiedlichen Datensatzgrößen im Verlauf des Trainings weiter kompensiert.

### 3.3 Trainingsworkflow

Der Ablauf des Trainings umfasst mehrere Phasen, die jeweils spezifische Ziele und Trainingsschritte beinhalten. Der Workflow (Abbildung 3.3) beginnt mit der Vorbereitung der Datensätze, die für das Training und die Validierung verwendet werden. Mit diesen Datensätzen werden die zwei Modelle DeepSMILESPredictor und DeepSMILESGenerator anschließend auf die Generierung und Validierung von AXL-Kinase-Inhibitoren vorbereitet.

#### 3.3.1 Complete Classification Training (Phase 0)

In dieser Phase wird der DeepSMILESPredictor trainiert, mit dem Ziel ein Modell zu entwickeln, das zwischen AXL-Kinase-Inhibitoren und anderen Molekülen unterscheiden kann. Damit soll anschließend die Wirksamkeit der generierten Moleküle validiert werden. Um diese Klassifizierung zu erreichen, wird in einem einstufigen Training das Klassifikations-MLP trainiert. Hierfür wurden die Klassendatensätze „axl\_kinase\_inhibitors“ und „compounds\_small“ benutzt, um dem Modell die Klassifizierung der zu validierenden Moleküle in Zielmoleküle und normale Compound-Moleküle beizubringen. Um ein Ungleichgewicht zwischen den beiden Klassen zu vermeiden, wurden Methoden wie *Class Weights* und *Oversampling* eingesetzt. Diese Techniken sorgen dafür, dass das Modell während des Trainings nicht eine Klasse bevorzugt, sondern eine ausgewogene Leistung in der Klassifikation beider Klassen erreicht. Insbesondere dann, wenn eine Klasse deutlich weniger Trainingsdaten besitzt als die andere.

**Class Weights [62]** – Sorgt dafür, dass das Modell höhere Gewichtungen für die selteneren Klassen zuweist. Dadurch wird das MLP während des Trainings dazu angeregt, Fehler im kleineren „axl\_kinase\_inhibitors“ Datensatz stärker zu gewichten, sodass es nicht dazu neigt, den größeren „compounds\_small“ Datensatz zu bevorzugen.

**Oversampling [63]** – Ist zusätzlich eine Technik, bei der die Beispiele der unterrepräsentierten Klasse künstlich vervielfältigt werden. Durch das Duplizieren der vorhandenen beiden Datensätzen werden die Anzahl der Beispiele beider Klassen angeglichen. Das Modell wird zu einer gleichmäßigen Lernleistung ohne die Fokussierung auf die Mehrheitsklasse gezwungen. Außerdem wird sichergestellt, dass genügend Trainingsdaten vorhanden sind.

Durch die Verbindung dieser beiden Techniken wird gewährleistet, dass das MLP-Modell eine ausgewogene Leistung bei der Klassifikation beider Klassen erreicht.

Während des Trainingsprozesses werden die definierten Deskriptoren gewichtet und das MLP lernt, welche Eigenschaften bei der Klassifizierung wichtig sind. Dieser Lernprozess erfolgt durch *Backpropagation*, wobei die Fehler der Vorhersagen minimiert werden, indem die Gewichte der Neuronen iterativ angepasst werden [35]. Nach Abschluss des Trainings ist das Modell in der Lage, die Wirksamkeit von Molekülen basierend auf ihren molekularen Deskriptoren vorherzusagen. Das trainierte Modell wird gespeichert und dient als Grundlage für das spätere Fine-Tuning mit einem Reinforcement Learning Ansatz. In dem es zur Bewertung der vom DeepSMILESGenerator erzeugten Molekülen verwendet wird, um deren potenzielle biologische Aktivität zu bestimmen.

Mit der Kombination der vorgestellten Methoden und Techniken stellt Phase 0 sicher, dass der DeepSMILESPredictor robust und präzise genug ist, um die Generierung und Validierung von Molekülen im weiteren Verlauf des Workflows sicherzustellen.

### 3.3.2 Basic Molecular Structures Training (Phase 1)

Als nächstes wird der DeepSMILESGenerator trainiert. Für das Hauptziel, der Generierung spezifischer AXL-Kinase-Inhibitoren. In Phase 1 wird dem DeepSMILESGenerator die grundlegende Fähigkeit zum Verständnis von Molekülstrukturen beigebracht. Das Ziel dieser Phase ist es, ein Modell zu schaffen, das in der Lage ist, vielfältige Molekülstrukturen zu erzeugen, die als Ausgangspunkt für weitere Spezialisierungen dienen. Das LSTM-Modell wird auf den Compound-Datensatz trainiert, der aus einer breiten Sammlung von Molekülstrukturen besteht und zusätzlich über Data Augmentation vervielfältigt wird.

**Data Augmentation [64]** – Ist eine Methode die künstlich neue Trainingsdaten erzeugt, um die Vielfalt eines Datensatzes zu erhöhen (Abbildung 3.2). Dies kann die Robustheit und den Lernerfolg des Modells deutlich verbessern. Die implementierte Technik nutzt die Fähigkeit der SMILES-Notationen, mehrere äquivalent gültige Darstellungen desselben Moleküls zu erzeugen, bevor die Moleküle aus dem Compound-Datensatz für das Training des LSTMs in die anschließende DeepSMILES-Notation umgewandelt werden. So wird für jedem der 1.874.870 Compounds mehrere Isomere erstellt und dem Training hinzugefügt. Das Modell kann lernen, Moleküle unabhängig von der verwendeten DeepSMILES-Zeichenfolge zu erkennen.

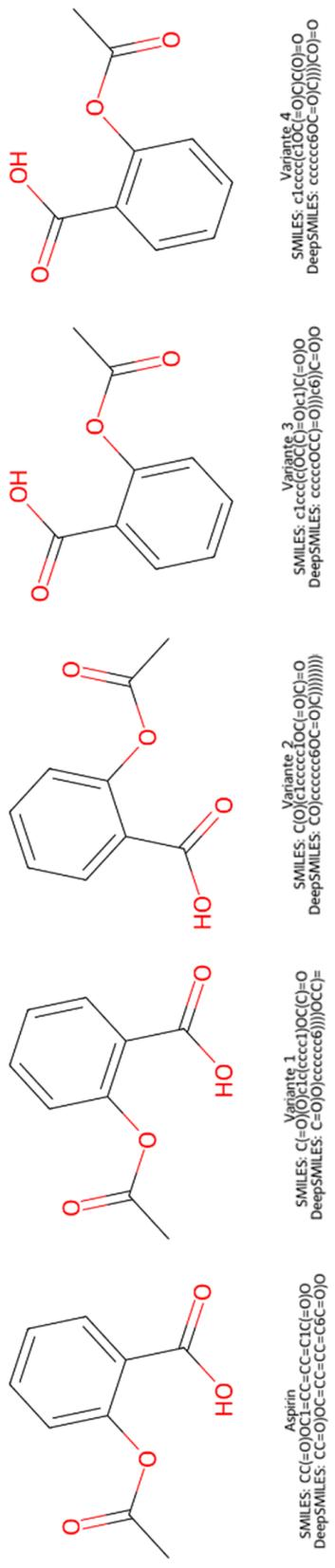


ABBILDUNG 3.2: Visualisierung der Data Augmentation am Beispiel von Aspirin. Gezeigt sind fünf verschiedene äquivalente SMILES- und DeepSMILES-Repräsentationen des Aspirinmoleküls. Durch die synthetische Erzeugung mehrerer Isomere jedes Moleküls wird der Trainingsdatensatz gezielt künstlich erweitert.

Ein zentraler Bestandteil des Trainingsprozesses ist die Definition eines geeigneten Vokabulars und die Tokenisierung der Daten, mit denen das Modell später neue Moleküle generiert. Für den DeepSMILESGenerator wurde dafür ein spezifisches Vokabular definiert, dass die häufigsten und relevantesten Molekülnotationen abdeckt. Die Tokenisierung ist ein weiterer wichtiger Schritt für den Erfolg bei der Datenverarbeitung für das Training des DeepSMILESGenerator. In diesem Kontext dient das Vokabular des Modells der Umwandlung der DeepSMILES-Struktur in eine Folge von Tokens. Tokens sind die vordefinierte kleinste und unteilbare Einheit von Textstrings. Jedes Token repräsentiert einen spezifischen atomaren oder strukturellen Bestandteil der Molekülstruktur. Das bedeutet, dass jedem Token ein eindeutiger Atomtyp oder eine strukturelle Komponente zugeordnet wird. Klammern („“) beispielsweise zeigen Verzweigungen an und geben Informationen über die Konnektivität, und einzelne Formelzeichen („C“, „H“) geben Informationen über die enthaltenen Elemente. Diese Tokenisierung ermöglicht es dem Modell die Sequenzinformationen zu lernen und Muster innerhalb der Moleküle zu erkennen, die für die chemische Validität und die syntaktische Korrektheit entscheidend sind.

Ein entscheidender Faktor der Tokenisierung sind spezielle Regeln für diese Aufteilung, die auf *RegExp* basieren, um Muster innerhalb der DeepSMILES Struktur zu erkennen. Der Tokenizer verwendet eine Reihe von regulären Ausdrücken um spezifische Komponenten und zusammenhängende Zeichen zu identifizieren und in einzelne Tokens umzuwandeln.

**Eckige Klammern:** – Gruppiert Zeichen innerhalb eckiger Klammern zusammen, die geladene Atome („[0-]“) oder bestimmte Konfigurationen in der DeepSMILES Notation darstellen können.

**Ringnummern:** – Stimmt mit Zahlen bis zu zwei Ziffern überein, denen ein Prozentzeichen („%“) vorangestellt ist, das verwendet wird, um Ringschlüsse in Molekülen mit mehr als 9 Ringen zu kennzeichnen.

**Halogenatome:** – Stellt sicher das Halogenatome wie Brom („Br“) und Chlor („Cl“) als zusammenhängende Token in der DeepSMILES Notation erkannt werden.

Die Tokenisierung auf Zeichenebene ermöglicht es dem DeepSMILESGenerator die Struktur und Zusammensetzung von Molekülen auf einer sehr detaillierten Ebene zu verstehen und zu generieren. Durch die präzise Definition und Erkennung einzelner Tokens kann das Modell lernen, valide DeepSMILES-Strukturen zu erzeugen.

Während dem Trainingsprozess nutzt das LSTM-Modell die tokenisierten DeepSMILES-Sequenzen um tiefgehende Abhängigkeiten und Muster in den Daten zu erfassen. Die spezielle Architektur des LSTM ermöglicht es dem Modell, sich an Informationen über längere Sequenzen hinweg zu erinnern [21]. Im Verlauf des Trainings werden die Modellparameter iterativ angepasst. Dadurch verbessert sich die Fähigkeit des DeepSMILESGenerator gültige und realistische Molekülstrukturen zu erzeugen. Mit jedem Trainingsepoch lernt das Modell weiter, wie es die richtigen Tokens in der richtigen Reihenfolge anordnen muss, um realistische molekulare Strukturen zu generieren.

Das Resultat dieser Trainingsphase wird anschließend gespeichert. Dieses vortrainierte Base-Trained Modell bildet die Grundlage für die nachfolgende Phase 2.

Durch die sorgfältige Auswahl des Vokabulars und die effektive Tokenisierung gewährleistet Phase 1, dass der DeepSMILESGenerator über ein umfassendes Repertoire für die Molekülgenerierung verfügt. Diese Grundlage ist entscheidend für die nachfolgenden Phasen des Workflows, insbesondere für die Spezialisierung auf AXL-Kinase-Inhibitoren in Phase 2.

### 3.3.3 Transfer Learning AXL-Kinase-Inhibitor (Phase 2)

In Phase 2 des Workflows wird das zuvor trainierte Modell aus Phase 1 mithilfe von Transfer Learning spezialisiert, um die Generierung von AXL-Kinase-Inhibitoren zu erlernen. Dabei wird das Base-Trained Modell, dass die grundlegenden molekularen Strukturen bereits verstanden hat, mit dem spezifischen Datensatz der bekannten 4564 AXL-Kinase-Inhibitoren nachtrainiert, um es spezifisch auf die Generierung von AXL-Kinase-Inhibitoren auszurichten.

Transfer Learning ermöglicht es, die bereits erlernten Fähigkeiten des Modells effektiv zu nutzen, um gezielt Molekülstrukturen zu generieren, die AXL-Kinase-Inhibitoren ähnlich sind. Anstatt das Modell von Grund auf neu zu trainieren, nutzt es die in Phase 1 erlernten Grundlagen valide Molekülstrukturen zu generieren und verfeinert die Parameter, um auf die speziellen Anforderungen der AXL-Kinase-Inhibitoren einzugehen. Durch diese Spezialisierung kann das Modell lernen, gezielt Molekülstrukturen zu generieren, die potenziell als neue AXL-Kinase-Inhibitoren fungieren können.

Das Ergebnis dieser Trainingsphase wird anschließend als Pre-Trained Modell gespeichert. Durch den gezielten Einsatz von Transfer Learning und der Nutzung des spezialisierten Datensatzes gewährleistet Phase 2, dass der DeepSMILES-Generator ein fundiertes Repertoire besitzt und über das Wissen zur Generierung von AXL-Kinase Inhibitoren verfügt. Dieses spezialisierte Modell ist von entscheidender Bedeutung für die letzte Phase des Workflows, in dem die Optimierung und Feinabstimmung der Molekülgenerierung durchgeführt wird.

### 3.3.4 Fine-Tuning Reinforcement Learning Ansatz (Phase 3)

In der letzten und entscheidenden Phase des Trainings, Phase 3, wird das Pre-Trained Modell mit einem Reinforcement Learning Ansatz feinabgestimmt. Ziel dieser Fine-Tuning Phase ist es die Generierung von spezifischen und validen AXL-Kinase-Inhibitoren weiter zu optimieren. Dafür wird das Modell, das bereits durch Transfer Learning auf AXL-Kinase-Inhibitoren spezialisiert wurde, weiter trainiert, um die Leistung zu verbessern, Moleküle zu erzeugen die den strukturellen und funktionalen Anforderungen entsprechen.

In Phase 3 wird dafür der DeepSMILESGenerator mit dem DeepSMILES-Predictor kombiniert. Diese Integration ermöglicht es, die Generierung von Molekülen durch ein Belohnungssystem zu steuern. Der DeepSMILESGenerator erzeugt eine Vielzahl an Molekülstrukturen, die anschließend vom DeepSMILES-Predictor analysiert und bewertet werden. Die Bewertung und Klassifizierung der generierten Moleküle erfolgt auf der Grundlage des bereits trainierten MLP-Modells, das

spezifisch auf AXL-Kinase-Inhibitoren abgestimmt ist. Durch diese Kombination wird gewährleistet, dass das Modell zunehmend in der Lage ist, Moleküle zu generieren, die den gewünschten Eigenschaften als AXL-Kinase-Inhibitoren entsprechen.

Das LSTM-Modell generiert dabei eine Vielzahl von Molekülen, von denen einige den gewünschten Kriterien entsprechen und andere nicht. Jedes Molekül wird basierend auf seiner Ähnlichkeit zu AXL-Kinase-Inhibitoren mit einer Belohnung bewertet. Moleküle, die den AXL-Kinase-Inhibitoren stark ähneln, erhalten höhere Belohnungen, während ungültige Strukturen mit Strafen versehen werden.

**Hohe Belohnung für Moleküle, die den Zielmolekülen ähneln** – Wenn ein Molekül AXL-Kinase-Inhibitoren strukturell stark ähnelt, erhält es eine besonders hohe Belohnung. Diese Ähnlichkeit wird anhand der vom DeepSMILES Predictor erkannten Schlüsselmerkmalen beurteilt. Enthält ein generiertes Molekül vermehrt solche charakteristischen Strukturen, desto höher ist die Wahrscheinlichkeit, diese Belohnung zu erhalten. Dies fördert gezielt die Generierung von Molekülen, die nicht nur chemisch valide, sondern auch biologisch relevant und potenziell wirksam als AXL-Kinase-Inhibitoren sein können. Durch diesen Verstärkungsprozess lernt das Modell, bevorzugt Moleküle zu erzeugen, die den strukturellen Anforderungen der Zielmoleküle entsprechen.

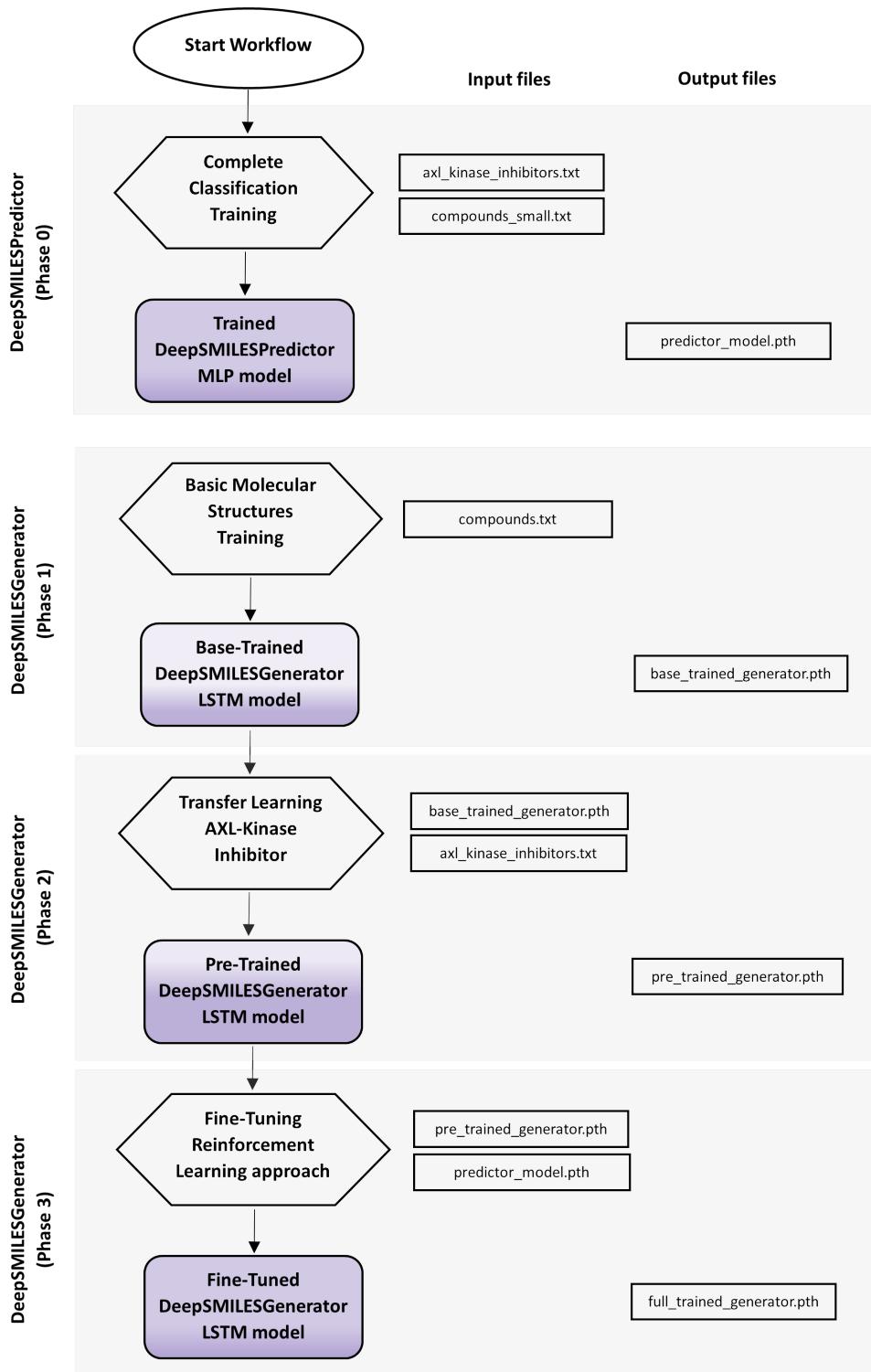
**Niedrige Belohnung für valide, aber nicht zielgerichtete Moleküle** – Moleküle, die den chemischen Regeln entsprechen, indem sie beispielsweise korrekte Bindungen aufweisen und keine strukturellen Inkonsistenzen zeigen, jedoch nicht vollständig die gewünschten Merkmale der AXL-Kinase-Inhibitoren erfüllen, erhalten eine kleine positive Belohnung. Dadurch soll dem Modell signalisiert werden, dass es auf dem richtigen Weg ist, jedoch die generierte Struktur noch nicht optimal den Zielvorgaben entsprechen.

**Strafen für ungültige Moleküle** – Ungültige Moleküle, die chemische Fehler aufweisen, wie etwa zu viele oder zu wenige Bindungen oder inkorrekte Atomanordnungen, werden mit einer Strafe belegt. Diese Maßnahme dient dazu, das Modell davon abzuhalten, zu stark auf Exploration zu setzen und die bereits gelerten validen Strukturen zu vernachlässigen. Zu viel Experimentieren könnte dazu führen, dass das Modell chemisch unsinnige Strukturen generiert, anstatt sich auf bewährte, valide Muster zu konzentrieren. Die Strafen sorgen somit dafür, dass das Modell in seinem Lernprozess stabil bleibt und keine fehlerhaften Moleküle bevorzugt.

Die differenzierte Bewertung der Moleküle spielt eine entscheidende Rolle im Lernprozess des Modells, indem sie dessen Verhalten nachhaltig steuert. Der Feintuning-Prozess erstreckt sich über mehrere Epochen, in denen das Modell fortlaufend neue Moleküle generiert. Dabei werden pro Epoche immer so viele Moleküle generiert bis ein festgelegter Wert an validen Molekülen vorhanden ist, um jede Iteration gleich viele Moleküle bewerten zu können. Durch das fein abgestimmte Belohnungssystem soll das Modell lernen, welche Molekülstrukturen bevorzugt werden sollen. Dabei bewegt sich das Modell auf einem schmalen Grat zwischen der Exploration neuer Strukturen und der Bewahrung bereits erlernter, chemisch valider Muster. Durch diesen kontinuierlichen Feedback-Zyklus optimiert das Modell so seine Fähigkeit, gezielt AXL-Kinase-Inhibitoren zu generieren und steigert damit

kontinuierlich die Erfolgswahrscheinlichkeit, Moleküle zu schaffen, die den therapeutischen Zielen entsprechen.

Der gesamte Workflow ist darauf ausgelegt, Schritt für Schritt spezialisiertere Fähigkeiten zu entwickeln und diese zu optimieren, um schließlich ein hochpräzises und wirksames Modell zu erschaffen. Am Ende von Phase 3 wird das vollständig trainierte Modell als Full-Trained Modell gespeichert, welches nun in der Lage sein sollte, Moleküle mit hoher Relevanz und Wirksamkeit zu generieren.



**ABBILDUNG 3.3: Schematische Darstellung des gesamten Workflows des entwickelten de novo Design Frameworks.** Der Prozess beginnt mit dem Klassifikationstraining für den DeepSMILES-Predictor. Anschließend erfolgt das Grundtraining des DeepSMILES-Generators zur Erfassung molekularer Strukturen, gefolgt von dem Transfer Learning zur Fokussierung auf AXL-Kinase-Inhibitoren. Abschließend wird der DeepSMILESGenerator mithilfe des Reinforcement Learning Ansatzes noch feinabgestimmt.

## Kapitel 4

# Ergebnisse

Die Ergebnisse des Trainings und der Evaluierung der entwickelten Machine Learning Modellen liefern entscheidende Einblicke in die Leistungsfähigkeit des entwickelten *in silico* Workflows.

Zur Generierung neuartiger AXL-Kinase-Inhibitoren wurden zwei spezialisierte Modelle konzipiert, die diesen Prozess gewährleisten. Für die Vorhersage der Wirksamkeit wurde der DeepSMILESPredictor entwickelt, ein Modell, dass auf einem MLP basiert und die Fähigkeit besitzt, Molekülstrukturen anhand ihrer potenziellen biologischen Aktivität zu klassifizieren. Dieser Ansatz ermöglicht es, strukturell plausible und biologisch relevante Moleküle zu identifizieren, die eine Hemmung der AXL-Kinase wahrscheinlich machen. Ergänzend wurde der DeepSMILES-Generator konstruiert, ein generatives Modell basierend auf einem LSTM-Netzwerk. In Kombination mit dem DeepSMILESPredictor ermöglicht dieses Modell die gezielte Generierung biologisch relevanter Zielmoleküle.

### 4.1 DeepSMILESPredictor Training

Der DeepSMILESPredictor wurde als MLP mit insgesamt 6 Schichten entwickelt, die von Batch-Normalisierung und ReLU-Aktivierung unterstützt werden. Die Architektur beginnt mit einer Schicht von 512 Neuronen, gefolgt von weiteren Schichten mit 256, 64, 32 und 8 Neuronen, bevor sie in der finalen Schicht eine Klassifikation der zwei Klassen AXL-Kinase-Inhibitoren (Zielmoleküle = 1) und Compounds (0) durchführt. Zur Regularisierung wurde zu den *Hidden Layers* eine Dropout-Schicht mit einer Rate von 40% eingefügt um das Modell vor Overfitting zu schützen. Das Modell wurde mit einer *Batch Size* von 64 trainiert und der *Adam Optimizer* mit einer Lernrate von 0,00005 angewendet. Der Trainingsprozess erstreckte sich über 150 Epochen, wobei ein Kriterium für *Early Stopping* definiert wurde, wenn eine Genauigkeit von 99,9% für AXL-Kinase Inhibitoren und 97,2% für Compounds erreicht wurde.

Während des Trainingsprozesses wurden die ECFPs und molekularen Deskritoren für die zu klassifizierenden Moleküle berechnet und als Eingaben für das Modell verwendet. Für die ECFPs wurde dabei der Radius auf 2 gesetzt, um Fragmente bis zu zwei Bindungen von jedem Atom ausgehend zu berücksichtigen. Zusätzlich wurde die Anzahl der Bits für den Fingerprint-Vektor auf 2048 festgelegt. Diese hohe Anzahl von Bits ermöglicht es, eine detaillierte und umfassende Darstellung der Molekülstruktur zu erzeugen.

Das resultierende Modell zeigt eine beeindruckende Leistungsfähigkeit bei der Klassifizierung der Zielmoleküle. Insbesondere der Einsatz der *Focal Cross-Entropy Loss Function*, *Class Weights* und *Oversampling* haben sich dabei als effektiv erwiesen um das Ungleichgewicht in den Datensätzen zwischen den Klassen zu bewältigen.

Um dem Problem der Ungleichverteilung entgegenzuwirken, wurde ein *Oversampling* des AXL-Kinase-Datensatzes mit einem Faktor von 6 durchgeführt. Das bedeutet, dass der Datensatz, der die AXL-Kinase-Inhibitoren umfasst, versechsfacht wurde. Nachträglich wurde auch ein *Oversampling* auf den Compound-Datensatz mit einem Faktor von 3 angewandt, da sonst die Differenz an den Trainingsdaten zu groß wurde. Auf diese Weise entstand eine ausgeglichene Verteilung zwischen den beiden Klassen. Für das Training wurden schließlich 27.384 AXL-Kinase-Inhibitor Daten und 32.436 Compound Daten verwendet. Zusätzlich zur Vergrößerung der Datensätze wurde das Gewicht der positiven Klasse, also der AXL-Kinase-Inhibitoren, gezielt angepasst, um deren Einfluss im Trainingsprozess zu verstärken. Dies erfolgte durch eine Gewichtung im inversen Verhältnis der positiven Klasse, was bedeutet, dass das Gewicht basierend auf dem Anteil der AXL-Kinase-Inhibitoren zur Gesamtanzahl der Daten berechnet wurde (Abbildung 4.1). In diesem Fall ergab sich ein *Class Weight* von 2,18 für die positive Klasse, was den Einfluss dieser Datenpunkte im Training erhöhte. Für die negative Klasse, also den Daten aus dem Compound-Datensatz, wurde ein fester Wert von 1,0 festgelegt.

$$\begin{aligned}
 \text{Class Weight (positiv)} &= \frac{\text{Gesamtzahl der Daten}}{\text{Anzahl der positiven Beispiele}} \\
 &= \frac{27384 + 32436}{27384} \\
 &= \frac{59820}{27384} \\
 &\approx 2.18
 \end{aligned} \tag{4.1}$$

**ABBILDUNG 4.1: Berechnung des Class Weights für die positive Klasse (AXL-Kinase-Inhibitoren).** Das Klassen-Gewicht wird berechnet, um das Ungleichgewicht zwischen den positiven und negativen Beispielen im Datensatz zu balancieren. Dabei wird die Gesamtzahl der Daten (59.820) durch die Anzahl der positiven Beispiele (27.384) geteilt. Das gerundete Ergebniss von 2,18 bedeutet, dass jedes positive Beispiel im Modell etwa 2,18-mal stärker gewichtet wird als ein negatives Beispiel.

Im Verlauf des Trainings sank der Verlust kontinuierlich und die Genauigkeit bei der Klassifizierung beider Klassen nahm zu, was zu einem stabil trainierten Modell führte (Abbildung 4.2). Das Modell erreichte seine definierten Ziele in Epoch 106 und beendet das Training mit einer Testgenauigkeit von 99,96% bei der Klassifizierung in den AXL-Kinase-Inhibitor Daten und 97,23% bei der Klassifizierung in den Compound Daten. Die Ergebnisse in Bezug auf die Genauigkeit wurden anschließend in einer Konfusionsmatrix visualisiert (Abbildung 4.3).

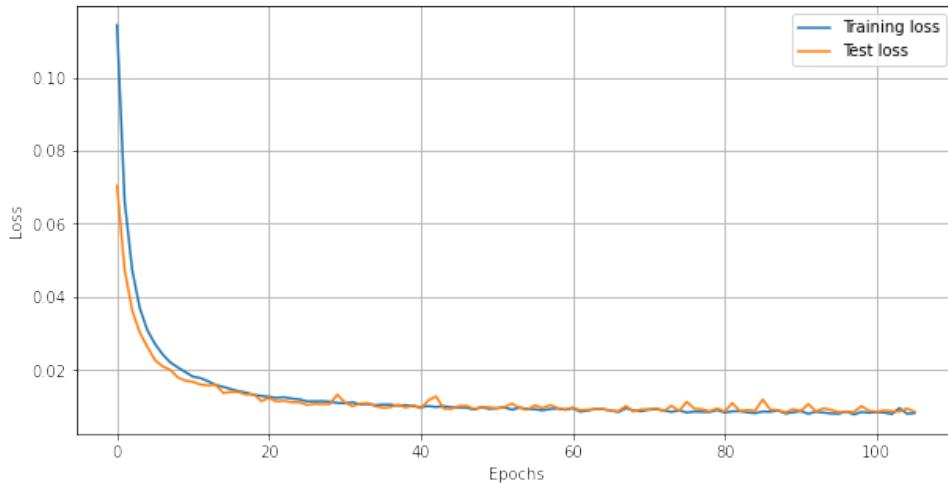


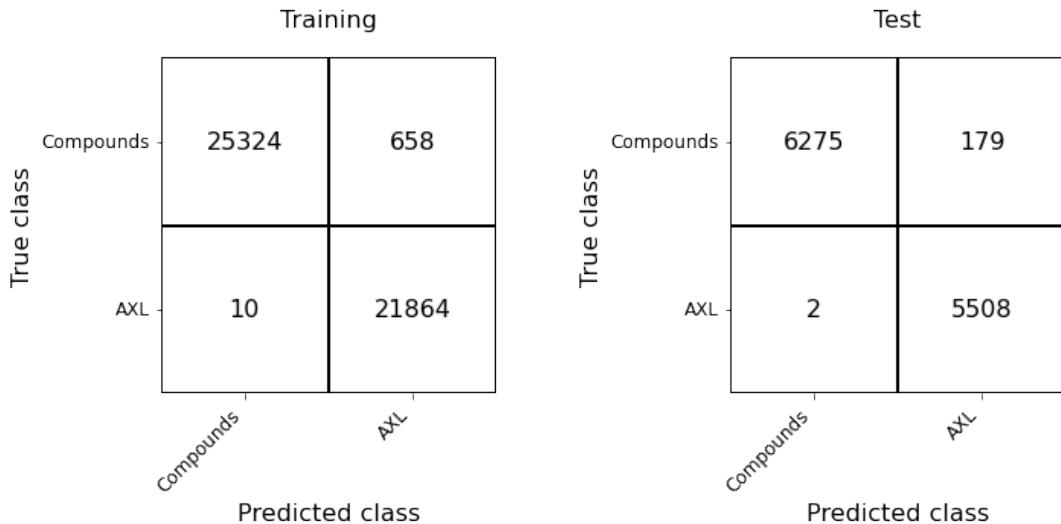
ABBILDUNG 4.2: Verlauf des Trainings- und Testverlusts für den DeepSMILES Predictor. Das Diagramm zeigt einen kontinuierlichen Rückgang des Verlustes sowohl im Trainings- als auch im Testdatensatz. Der Verlust stabilisiert sich nach etwa 40 Epochen und erreicht nach insgesamt 106 Epochen die festgelegte Zielgenauigkeit. Der nahezu parallele Verlauf der Trainings- und Testkurven deutet darauf hin, dass kein Overfitting aufgetreten ist.

#### 4.1.1 Feature Importance

Zu dem Klassifikationsergebnis wird zusätzlich eine *Feature Importance* Analyse durchgeführt, um zu ermitteln, welche Deskriptoren die stärkste Auswirkung auf die getroffenen Klassifikationsentscheidungen hatten (Abbildung 4.4).

Die Ergebnisse dieser Auswertung verdeutlichen, dass insbesondere die ECFPs eine beachtliche Signifikanz für die Klassifikationsentscheidungen haben. Unter den molekularen Deskriptoren hebt sich der BalabanJ-Index als der einflussreichste hervor, was auf die besondere Relevanz dieses Deskriptors für die Struktur- und Topologieinformationen der Moleküle hinweist. Des Weiteren, zeigt auch der Deskriptor für die Anzahl der ungepaarten Elektronen im Molekül eine signifikante Bedeutung.

Diese Ergebnisse deuten darauf hin, dass das Modell eine klare Präferenz für Deskriptoren ausweist, welche die molekularen Strukturen und die damit verbundene strukturelle Flexibilität abbilden. Im Gegensatz dazu scheinen traditionelle Deskriptoren wie das Molekulargewicht oder die Anzahl bestimmter chemischer Elemente im Molekül für die Vorhersage des Modells weniger relevant zu sein. Dabei wird deutlich, dass das Modell eher komplexere strukturelle Merkmale berücksichtigt, die in der Molekül-Topologie und -Struktur verankert sind, anstatt auf einfache physikochemische Eigenschaften zu setzen.

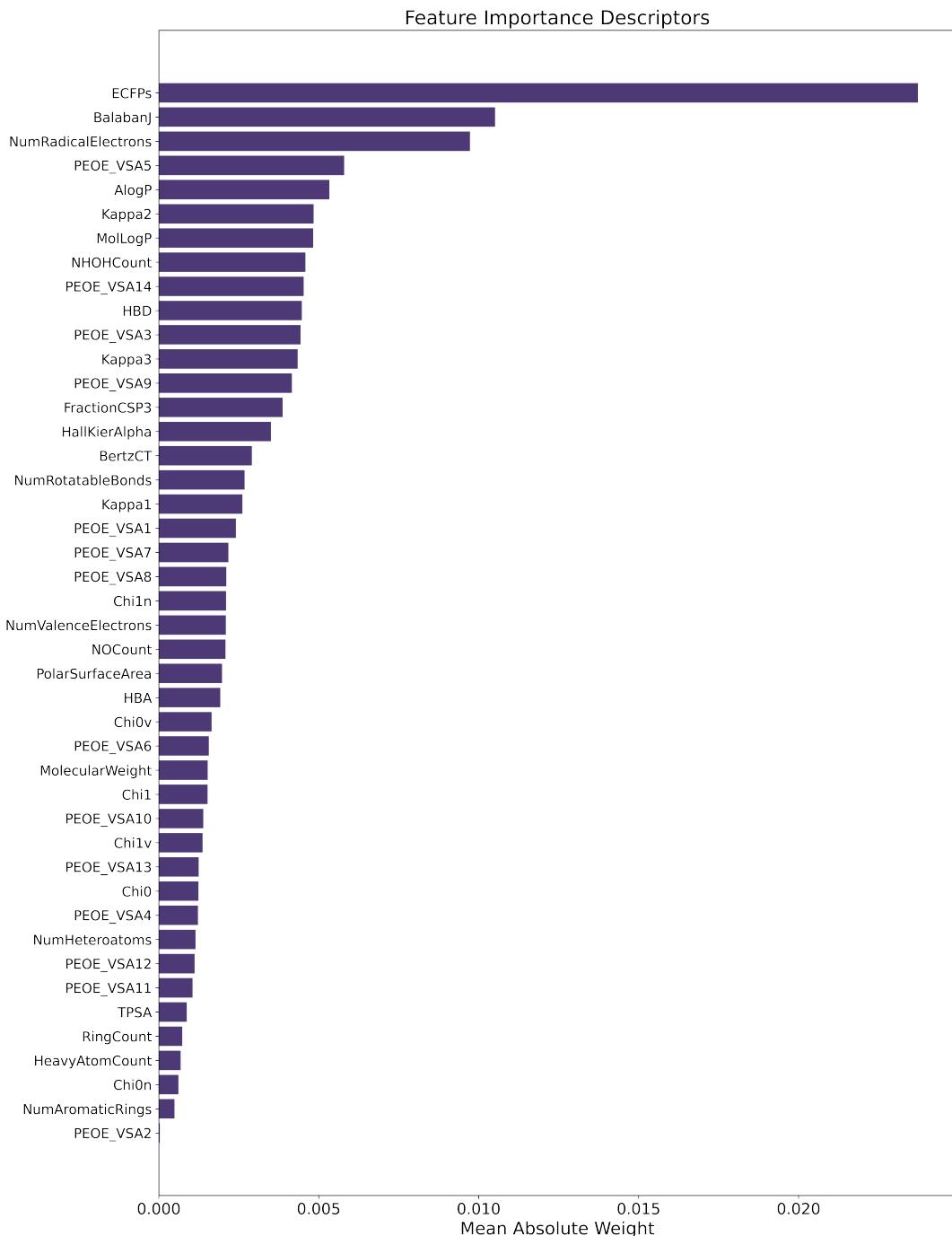


**ABBILDUNG 4.3: Konfusionsmatrizen für Trainings- und Testdatensätze des DeepSMILES Predictor.** Konfusionsmatrix für den Trainingsdatensatz (links) und Konfusionsmatrix für den Testdatensatz (rechts). Die Visualisierung spiegelt die erreichte Klassifikationsgenauigkeit von 99,96% für die AXL-Kinase-Inhibitor-Daten und 97,23% für die Compound-Daten wieder. Die Zuverlässigkeit bei der Erkennung von AXL-Kinase-Inhibitoren wird durch die geringere Anzahl falsch positiver Vorhersagen deutlich.

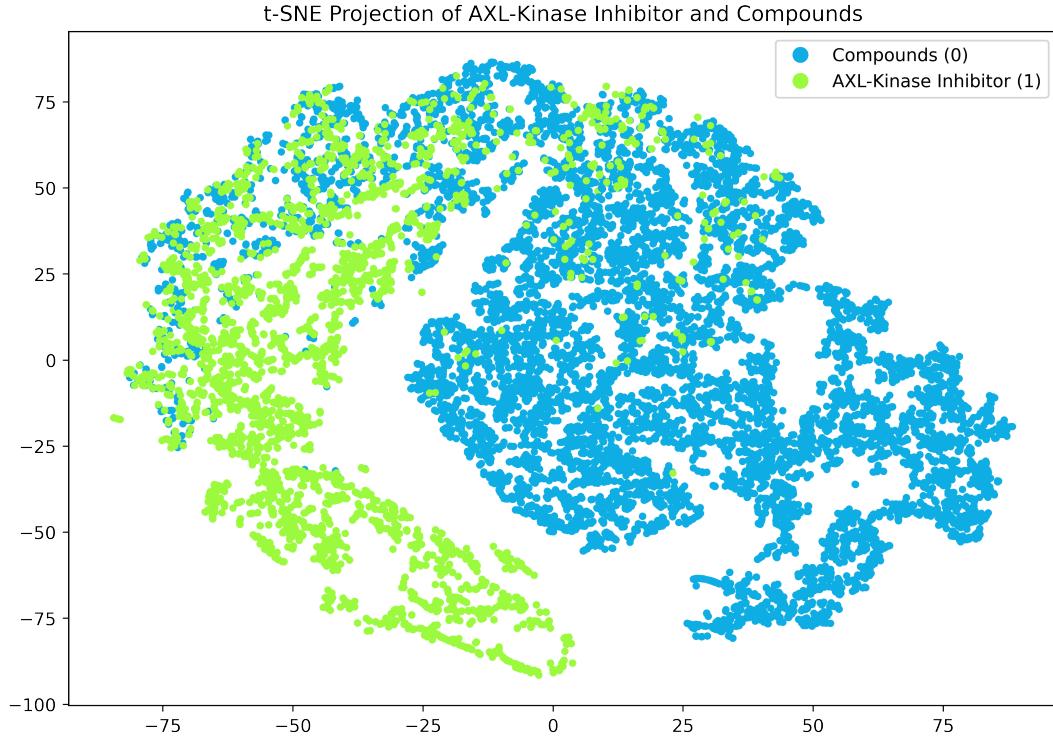
#### 4.1.2 T-SNE Plot

Die Fähigkeit zur Klassifikation der Moleküle wurde in einem t-SNE Plot visualisiert. *T-distributed Stochastic Neighbor Embedding* ist ein Verfahren zur visuellen Darstellung hochdimensionaler Daten in einem niedrig dimensionalen Raum, in diesem Fall in 2D [65]. Da die Molekülklassifikation auf einer Vielzahl von Deskriptoren basiert, handelt es sich bei unseren Daten um hochdimensionale Datensätze.

Das t-SNE Verfahren sorgt dafür, dass ähnlich Datenpunkte, die sich in diesen hochdimensionalen Räumen gleichen auch in einer niedrigdimensionalen Darstellung nahe beieinander liegen. Im Gegensatz dazu werden unähnliche Punkte weiter voneinander entfernt dargestellt, um visuell eine klare Unterscheidung zwischen den Molekülklassen zu ermöglichen. Für die Anwendung wurde der AXL-Kinase-Inhibitor-Datensatz und der Compound-Datensatz in einen t-SNE Plot überführt (Abbildung 4.5). Anschließend wurde mit dem trainierten DeepSMILES Predictor klassifiziert. Die t-SNE Projektion zeigt, dass das Modell in der Lage war, beide Klassen eindeutig zu trennen. Die grünen Punkte repräsentieren die AXL-Kinase-Inhibitoren, während die blauen Punkte allgemeine Verbindungen darstellen, also normale Compounds. Diese klare Trennung zwischen den beiden Gruppen zeigt die gute Klassifikationsleistung des Modells. Die Clusterbildung weist darauf hin, dass sich die AXL-Kinase-Inhibitoren strukturell von den Compounds unterscheiden und das Modell diese Unterschiede erfolgreich genutzt hat, um diese zu trennen. Dieses Ergebnis bestätigt, dass das Modell die molekularen Eigenschaften, die für die AXL-Kinase-Inhibitoren relevant sind, gut erfasst hat, was eine effektive Klassifizierung ermöglicht.



**ABBILDUNG 4.4: Analyse der Feature Importance.** Die Abbildung zeigt die gewichtete Bedeutung der Deskriptoren, die die Klassifikationsentscheidungen des DeepSMILESPredictor maßgeblich beeinflusst haben. Der Mean Absolute Weight auf der x-Achse repräsentiert den absoluten Einfluss jedes Deskriptors auf das Modell, wobei höhere Werte eine stärkere Bedeutung symbolisieren.



**ABBILDUNG 4.5: T-SNE Visualisierung der Klassifikationsergebnisse des DeepSMILES Predictor.** Der T-distributed Stochastic Neighbor Embedding Plot zeigt die zweidimensionale Projektion hochdimensionaler Molekülstrukturen aus dem Compound- und AXL-Kinase-Inhibitor-Datensatz. Ähnliche Moleküle sind im Plot räumlich nah beieinander dargestellt, während unähnliche Moleküle weiter voneinander entfernt sind. Die Punkte repräsentieren allgemeine Verbindungen (blau) und AXL-Kinase-Inhibitoren (grün). Dabei sind zwei deutliche Cluster sichtbar.

## 4.2 DeepSMILESGenerator Training

### 4.2.1 Base-Training Phase 1

Der DeepSMILESGenerator basiert auf einem LSTM-Grundmodell mit drei Schichten, wobei jede Schicht eine Größe von 512 Neuronen besitzt. Zusätzlich verfügt das Netzwerk über eine Embedding-Schicht mit 128 Dimensionen, die dazu dient, die einzelnen Tokens aus dem Vokabular in dichte Vektoren zu transformieren. Statt jedes Symbols als einzelnes isoliertes Objekt zu behandeln, werden sie als Vektoren in einem Raum dargestellt, wo ihre Ähnlichkeit durch die Nähe im Raum ausgedrückt wird. Dies bedeutet, dass ähnliche Tokens ähnliche Vektoren haben, wodurch das Modell lernen kann, wie Tokens in Beziehung zueinander stehen. Das Modell verwendet außerdem eine Dropout-Rate von 0,2 zur Regularisierung von Overfitting. Layer-Normalisierung wird in diesem Setup nicht verwendet.

Während Phase 1 des Trainingsprozesses wurde dieses Grundmodell über einen Zeitraum von 20 Epochen hinweg mit einer Lernrate von 0,001 trainiert. Für die *Data Augmentation* wurde ein Augmentationsfaktor von 5 angewendet, um multiple Variationen der Moleküle für das Base-Training zu generieren. Zusätzlich kam eine

Batch-Size von 256 zum Einsatz, um diese große Datenmenge auch effektiv bearbeiten zu können. Diese Größe unterstützt nicht nur die gleichzeitige Verarbeitung vieler Datenpunkte, sondern trägt auch dazu bei stabile Gradienten während des Trainings zu erhalten.

Während des Trainingsprozesses wird die Generierung der Moleküle kontinuierlich überprüft, um sicherzustellen, dass das Modell erfolgreich lernt und biologisch relevante Strukturen produziert. Das Feedback erfolgt durch den Vergleich der generierten Molekülsequenzen mit den Zielsequenzen, aus dem Eingabedatensatz. Dabei wird der negative *log-Likelihood Loss* verwendet, um die Abweichung zwischen den Vorhersagen des Modells und den tatsächlichen Zielsequenzen zu berechnen. Durch diesen Loss kann das Modell während des Trainings seine Vorhersage iterativ verbessern (Abbildung 4.6). Das Modell zeigte eine deutliche Lernkurve und begann nach einigen Epochen, vermehrt valide Moleküle zu produzieren. Zusätzlich wird das Modell nach jeder Epoche auf einem Validierungsdatensatz getestet, um sicherzustellen, dass es nicht nur auf die Trainingsdaten passt, sondern auch auf neue, ungesehene Daten korrekt reagieren kann. Die Trainings- und Validierungsverluste verlaufen konsistent, das deutet auf ein stabiles und effektives Training hin, bei dem das Modell in der Lage ist, relevante Muster zu extrahieren und keine Anzeichen von Overfitting aufweist.

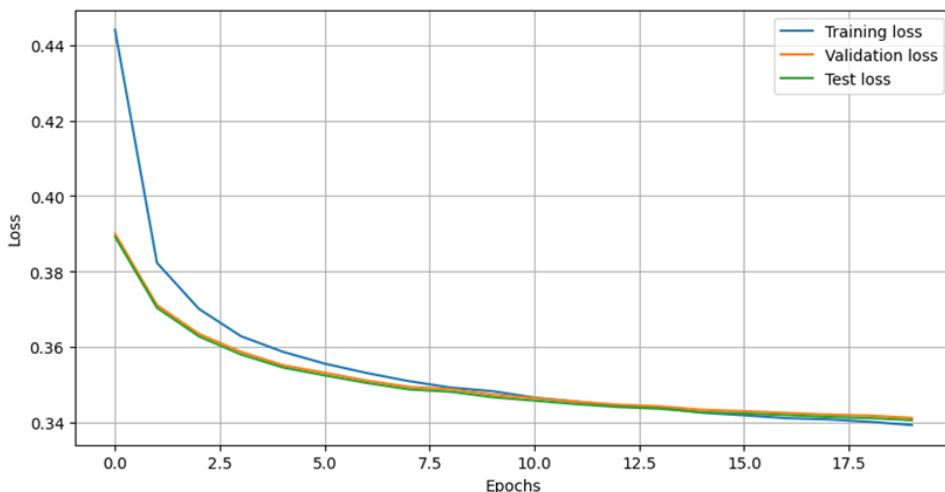
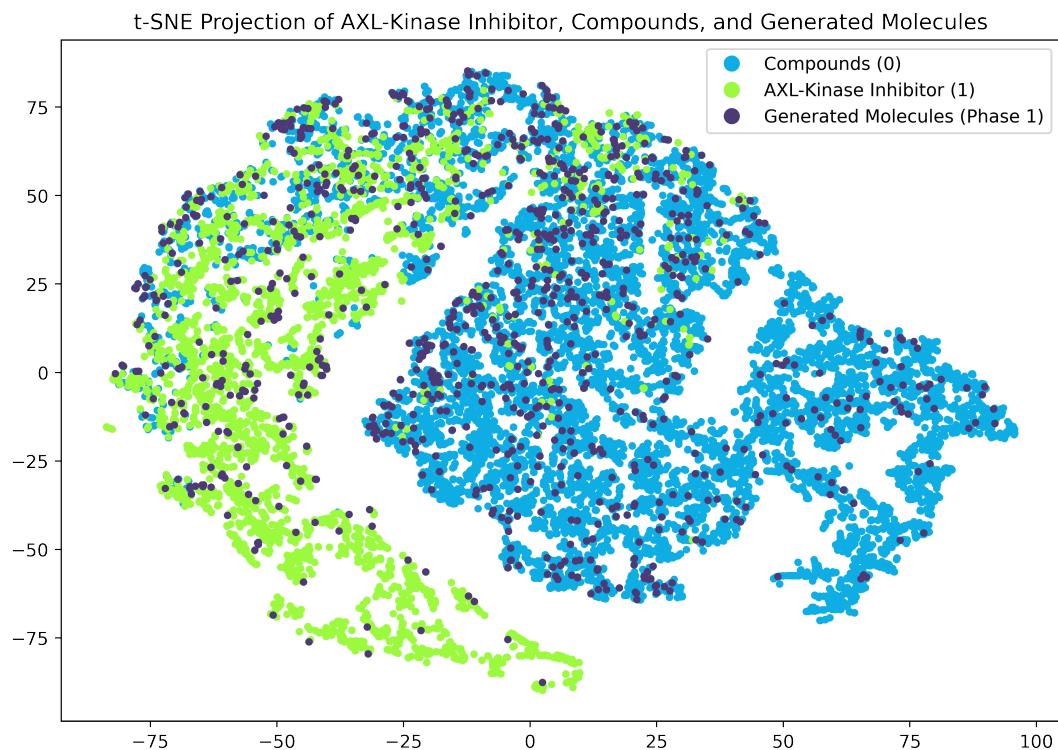


ABBILDUNG 4.6: Verlauf des Trainingsverlustes DeepSMILES-Generator (Phase 1). Die Abbildung zeigt den Trainings-, Validierungs- und Testverlust über 20 Epochen hinweg. Phase 1 des Trainings zielt darauf ab, dem Modell eine grundlegende Fähigkeit zur Generierung allgemeiner Molekülstrukturen zu vermitteln. Während des Trainingsprozesses wird der negative log-Likelihood Loss als Fehlermaß verwendet. Die Verluste verlaufen konsistent und ohne Anzeichen von Overfitting.

Um die Leistungsfähigkeit des Modells nach dem Base-Training zu visualisieren, wurden anschließend 1000 valide Moleküle erzeugt und in den bestehenden t-SNE Plot integriert. Dieser aktualisierte t-SNE Plot zeigt die Verteilung der AXL-Kinase-Inhibitoren (grün), der Compounds (blau) sowie der neu generierten Moleküle aus dem aktuellen Training (lila) (Abbildung 4.7). Auffallend ist, dass die neu generierten Moleküle gleichmäßig über den gesamten Raum verteilt liegen und alle

Molekülklassen abdecken. Dies verdeutlicht, dass das Modell in der Lage ist, eine breite Vielfalt an Molekülstrukturen zu generieren, ohne eine übermäßige Fokussierung auf einzelne Bereiche. Durch die Integration eines divers aufgebauten Trainingsdatensatzes konnte der DeepSMILESGenerator eine große Vielzahl an Molekülen hervorbringen, was die Flexibilität und Generalisierungsfähigkeit des Modells belegt. So wird sichtbar, dass der DeepSMILESGenerator nach dem Base-Training in der Lage ist, Moleküle zu entwerfen, die unterschiedliche Strukturtypen erfassen und somit das gesamte chemische Spektrum der zugrundeliegenden Daten wider- spiegelt.



**ABBILDUNG 4.7: T-SNE Plot nach dem Base-Training (Phase 1) des DeepSMILESGenerator.** Die Abbildung zeigt die Verteilung der generierten Moleküle (lila) im Kontext der als AXL-Kinase-Inhibitoren klassifizierten Molekülen (grün) und den als allgemeine Verbindungen klassifizierten Molekülen (blau). Die neu erzeugten Moleküle sind gleichmäßig über den Raum verteilt und decken alle Molekül- klassen ab.

#### 4.2.2 Transfer Learning Phase 2

In Phase 2 des Trainingsprozesses wurde das Base-Trained Modell weiter verfeinert, um die gezielte Generierung von AXL-Kinase-Inhibitoren zu gewährleisten. Über einen Zeitraum von 10 Epochen wurde das Modell mit einer Lernrate von 0,001 trainiert. Da das Modell nun spezifischere Ergebnisse liefern sollte, wurde die *Data Augmentation* angepasst. Der Augmentationsfaktor wurde auf 2 reguliert, um den AXL-Kinase-Inhibitor-Datensatz nicht zu sehr zu verfälschen. Zudem wurde die *Batch-Size* auf 128 verringert, da die Datenmenge kleiner geworden ist. Diese

Anpassungen im Transfer Learning ermöglichen es dem Modell, präzisere Vorhersagen zu treffen und sich auf die Generierung von AXL-Kinase-Inhibitor relevanten Strukturen zu konzentrieren.

Der Verlauf des Trainingsverlustes zeigt eine stetige und konsistente Abnahme der Trainings-, Validierungs- und Testverluste über die gesamten 10 Epochen (Abbildung 4.8). Der kontinuierliche Rückgang des Trainingsverlustes deutet darauf hin, dass das Modell zunehmend besser in der Lage ist, die Muster aus dem AXL-Kinase-Inhibitor-Datensatz zu erfassen. Die Tatsache, dass der Validierungsverlust stabil und gleichmäßig über dem Trainingsverlust bleibt, ist ein Zeichen dafür, dass kein Overfitting auftritt und deutet auf ein gut reguliertes Lernverhalten hin.

Das Diagramm veranschaulicht außerdem, dass der DeepSMILESGenerator nach Phase 1 des Trainings, in der die grundlegenden molekularen Strukturen erlernt wurden, weiterhin zahlreiche Anpassungen vornimmt, um sich auf die Generierung der AXL-Kinase-Inhibitoren zu spezialisieren. Dies zeigt sich in der großen Steigung der Verlustfunktion, die das Lernen symbolisiert. Obwohl der Trainingsverlust seine vollständige Sättigung noch nicht erreicht hat, wurde das Training bewusst nach 10 Epochen beendet, um zu verhindern, dass das Modell die in Phase 1 gelernten allgemeinen Molekülstrukturen durch ein zu langes Training auf den neuen Datensatz wieder „verlernt“.

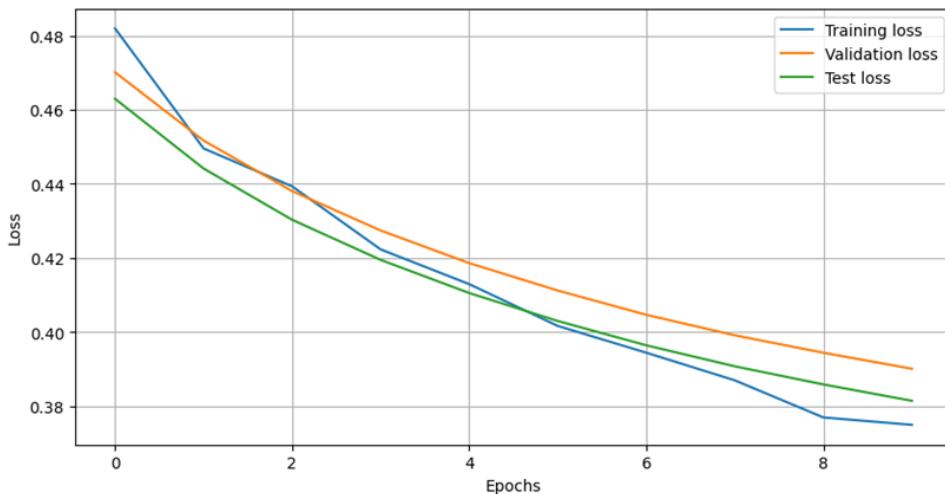
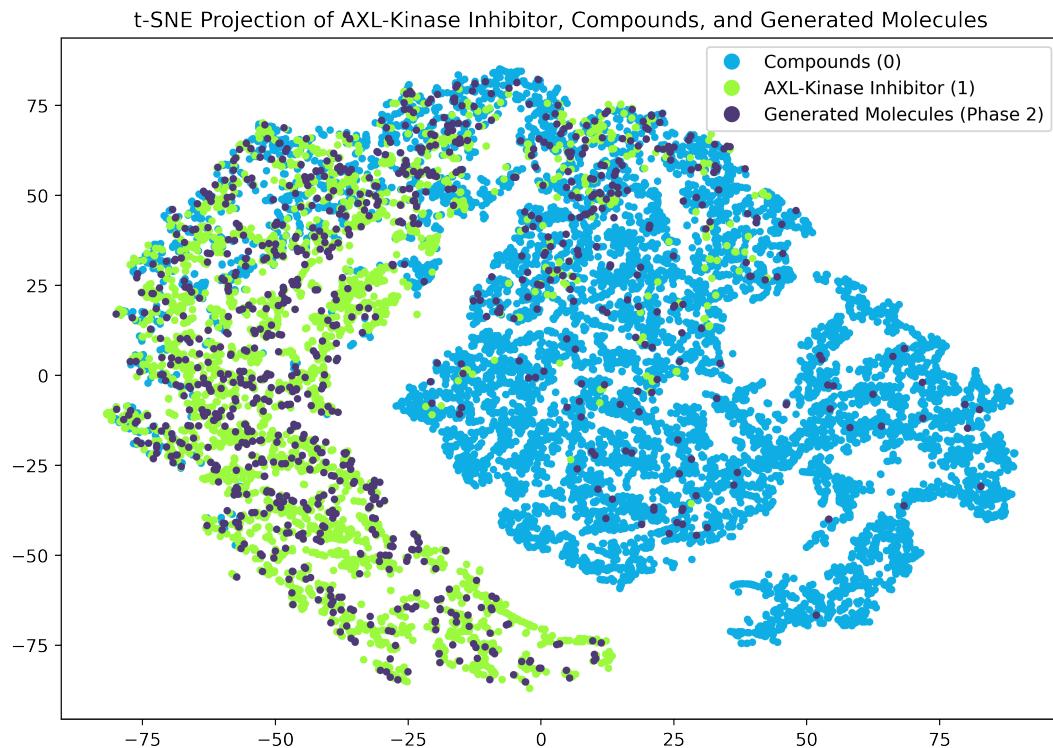


ABBILDUNG 4.8: Verlauf des Trainingsverlustes DeepSMILES-Generator (Phase 2). Die Abbildung zeigt den Trainings-, Validierungs- und Testverlust über 10 Epochen hinweg. Phase 2 des Trainings konzentriert sich darauf, die bereits erlernten generellen Molekülstrukturen auf AXL-Kinase-Inhibitoren zu spezialisieren. Der gleichmäßige Rückgang der Verluste in allen Datensätzen zeigt, dass das Modell kontinuierlich lernt und keine Anzeichen von Overfitting aufweist.

LSTMs können Abhängigkeiten über längere Zeiten hinweg erfassen als einfachere RNNs, jedoch haben sie immer noch ein begrenztes Kontextfenster [66]. Wenn ein LSTM-Modell eine sehr große Menge an neuen Daten verarbeitet, kann es dennoch Schwierigkeiten haben, sich an frühere Informationen zu erinnern, da die neuen Eingaben die älteren überlagern. Dies geschieht dann, wenn das *Forget-Gate* des

LSTMs zu dominant wird. Durch das verkürzte Training in Phase 2 wird dieses Problem ausgeglichen. Das Modell erhält ausreichend Zeit, um spezifische Muster der AXL-Kinase-Inhibitoren zu erlernen, ohne die zuvor erlernten allgemeinen Strukturen aus Phase 1 zu verlieren.



**ABBILDUNG 4.9: T-SNE Plot nach dem Transfer Learning (Phase 2) des DeepSMILESGenerator.** Die Abbildung zeigt die Verteilung der generierten Moleküle (lila) im Kontext der als AXL-Kinase-Inhibitoren klassifizierten Molekülen (grün) und den als allgemeine Verbindungen klassifizierten Molekülen (blau). Die generierten Moleküle verschieben sich zunehmend in Richtung der AXL-Kinase-Inhibitoren.

Die Ergebnisse des Trainings aus Phase 2 wurden in einem weiteren t-SNE Plot visualisiert (Abbildung 4.9). Dabei wurden 1000 valide Moleküle, die nach dem Transfer Learning generiert wurden in den klassifizierenden t-SNE Plot integriert. Im Vergleich zum vorherigen Plot wird eine deutliche Verschiebung der generierten Moleküle in Richtung der AXL-Kinase-Inhibitoren sichtbar. Die Dichte der Moleküle im Bereich der als Compounds klassifizierten Strukturen nimmt merklich ab, dennoch bleiben einige Moleküle über den gesamten Raum verteilt. Dies unterstreicht, dass das Modell nach Phase 2 gezielt auf AXL-Kinase-Inhibitoren fokussiert wurde, ohne jedoch die Fähigkeit zu verlieren, die komplette Bandbreite an chemischen Strukturen für die Generierung von Targets zu verwenden. Die Verlagerung der generierten Moleküle in Richtung der AXL-Kinase-Inhibitoren zeigt, dass der DeepSMILESGenerator in der Lage ist, Strukturen zu erzeugen, die zunehmend den Zielen ähneln und eine biologische Aktivität gegen AXL-Kinase aufweisen könnten. Gleichzeitig behält das Modell seine Flexibilität, die gesamte Bandbreite an zugänglichen chemischen Informationen zu nutzen, um vielfältige und potenziell relevante Moleküle zu generieren.

### 4.2.3 Fine-Tuning Phase 3

Für das Fine-Tuning in Phase 3 wird der bereits vollständig trainierte DeepSMILES-Predictor mit dem sich in Phase 2 befindenden DeepSMILESGenerator kombiniert. Dabei wird das LSTM-Grundmodell des DeepSMILESGenerator über 100 Epochen mittels des Reinforcement Learning Ansatzes angepasst. Um eine feine Abstimmung zu erreichen, wird die Lernrate auf 0,000015 gesenkt und die *Batch-Size* auf 16 verringert. Im Gegensatz zu den vorherigen Phasen sind keine Trainingsdaten und keine *Data Augmentation* mehr erforderlich, da das Modell in dieser Phase die Moleküle selbst generiert. Pro Iteration werden 400 valide Moleküle erzeugt, die anschließend vom DeepSMILES Predictor bewertet werden.

Der Bewertungsprozess basiert auf dem Belohnungssystem, welches generierte Moleküle differenziert bewertet. Zielgerichtete als AXL-Kinase-Inhibitor klassifizierte Moleküle erhalten eine Belohnung von 2,5 Punkten. Valide Moleküle, die nicht als zielgerichtet klassifiziert werden und damit nur Compounds darstellen, erhalten eine Belohnung von 0,5 Punkten. Fehlgeschlagene Konvertierungen oder ungültige Moleküle werden mit einer Strafe von 7,5 Punkten belegt. Diese Belohnungen werden anschließend normalisiert, um das Training stabiler gegenüber Extremwerten zu machen und alle Belohnungen auf eine einheitliche Skala zu bringen.

Diese enormen Strafen auf ungültige Moleküle sorgen dafür, dass während dem Fine-Tuning in Phase 3 des Trainings nicht alle zuvor gelernten Strukturen aus Phase 1 und 2 komplett verlernt oder verworfen werden. Stattdessen soll das Modell nur schrittweise kleine Anpassungen durchführen, um valide Moleküle so zu modifizieren, dass sie AXL-Kinase-Inhibitoren ähneln, ohne dabei die grundlegende Validität der Strukturen so überarbeiten, dass keine validen Moleküle mehr entstehen. Um diesen Prozess zu unterstützen, wird auch eine kleine Belohnung für valide, aber nicht als AXL-Kinase-Inhibitor klassifizierte Moleküle, vergeben.

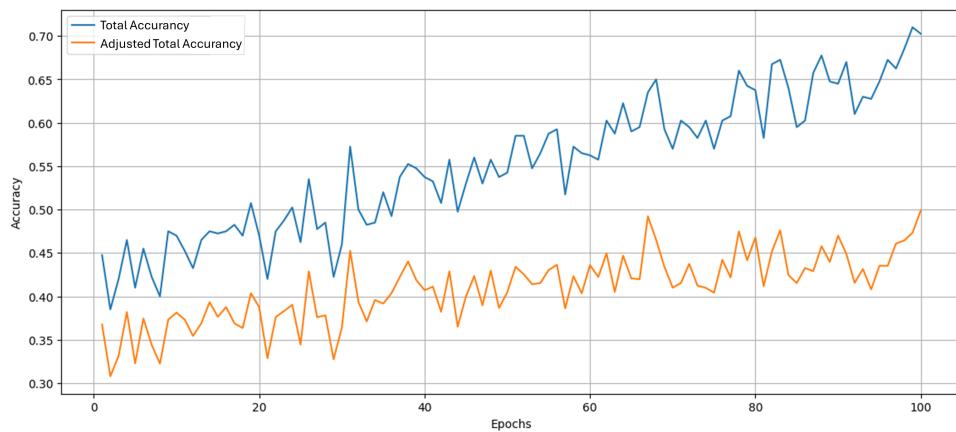
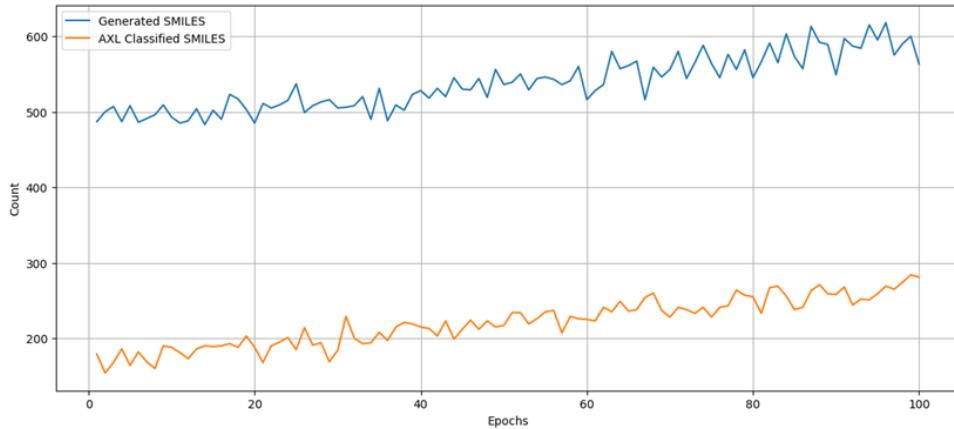


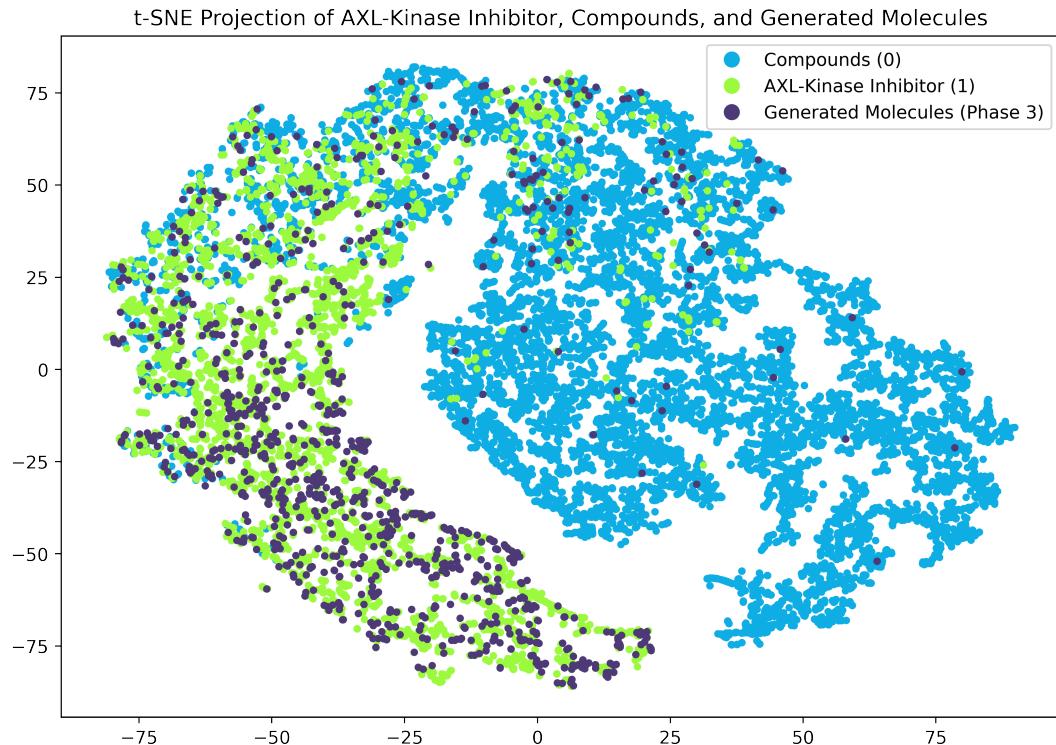
ABBILDUNG 4.10: Verlauf der Gesamtgenauigkeit der generierten Moleküle DeepSMILESGenerator (Phase3). Diese Abbildung zeigt, wie sich die Gesamtgenauigkeit (blau) und die korrigierte Gesamtgenauigkeit (orange), die auch die generierten ungültigen Moleküle berücksichtigt, über den gesamten Trainingszeitraum entwickeln. Gegeben Ende des Trainings wird eine Differenzierung der Kurven sichtbar.



**ABBILDUNG 4.11: Verlauf der Anzahl der generierten Moleküle DeepSMILESGenerator (Phase3).** Diese Abbildung zeigt die Anzahl der pro Epoche generierten Moleküle (blau), die sowohl valide als auch ungültige Moleküle einschließt, sowie die Anzahl der korrekt als AXL-Kinase-Inhibitoren klassifizierten Moleküle (orange). Im Verlauf des Trainings steigt die Anzahl der zielgerichteten Moleküle kontinuierlich an, während die Gesamtzahl der generierten Moleküle weitgehend konstant bleibt.

Der Verlauf des Trainings zeigt eine deutliche Verbesserung in der Fähigkeit des Modells spezifische AXL-Kinase-Inhibitoren zu generieren (Abbildung 4.11). Über die Epochen hinweg ist eine kontinuierliche Steigerung erkennbar, wobei das Modell trotz natürlicher Schwankungen stetig mehr zielgerichtete Moleküle erzeugt. Diese Schwankungen bei der Generierung von Molekülen in den einzelnen Epochen ist auf die inhärente Variabilität des Modells zurückzuführen. Dies bedeutet, dass das Modell nicht bei jedem Versuch genau gleich viele Zielmoleküle generieren kann, durch die fast unendliche Möglichkeit in der Kombinatorik der verschiedenen Molekülbausteine werden einmal mehr valide Zielmoleküle und einmal weniger generiert. Bis zum Ende des Trainings erreicht der DeepSMILESGenerator eine Genauigkeit bei der Generierung von AXL-Kinase Inhibitoren von beeindruckenden 70%. Ungeachtet der hohen Strafen für ungültige Moleküle stieg mit dem Fortschritt auch die Anzahl ungültiger Moleküle, was die korrigierte Gesamtgenauigkeit auf 50% reduziert (Abbildung 4.10). Das bedeutet, dass die Hälfte aller generierten Moleküle AXL-Kinase-Inhibitoren sind.

Diese Moleküle wurden in einem weiteren t-SNE Plot visualisiert (Abbildung 4.12). Dabei wurden 1000 valide Moleküle generiert und in den klassifizierenden t-SNE Plot integriert. Der t-SNE Plot der Phase 3 zeigt eine weitere deutliche Verschiebung der generierten Moleküle im Vergleich zu den früheren Phasen, da die generierten Moleküle sich viel dichter im Zentrum der AXL-Kinase-Inhibitoren gruppieren. Das deutet darauf hin, dass das Modell im Fine-Tuning erfolgreich gelernt hat, die Generierung der AXL-Kinase-Inhibitoren zu verfeinern. Im Vergleich zu dem Plot aus Phase 2 sind deutlich weniger Moleküle im Bereich der allgemeinen Compounds lokalisiert. Dennoch findet man vereinzelte Moleküle über den kompletten Bereich der allgemeinen Compounds verteilt. Das zeigt, dass das Modell weiterhin in der Lage ist Moleküle über das gesamte chemische Spektrum zu generieren und die Vielfalt der molekularen Strukturen nicht verlernt wurde.



**ABBILDUNG 4.12: T-SNE Plot nach dem Fine-Tuning (Phase 3) des DeepSMILESGenerator.** Die Abbildung zeigt die Verteilung der generierten Moleküle (lila) im Kontext der als AXL-Kinase-Inhibitoren klassifizierten Molekülen (grün) und den als allgemeine Verbindungen klassifizierten Molekülen (blau). Die generierten Moleküle haben sich vermehrt im Zentrum der AXL-Kinase-Inhibitoren konzentriert.

### 4.3 Erweiterte Trainingsansätze

Um das Trainingskonzept des DeepSMILESGenerator besser bewerten zu können, wurden zum Vergleich mehrere alternative Trainingsansätze durchgeführt. Dabei wurde als Referenz ein Training ohne Transfer Learning durchgeführt, bei dem das Fine-Tuning direkt auf das Basismodell angewendet wurde. Zusätzlich gab es einen weiteren Ansatz, bei dem umgekehrt komplett auf das Fine-Tuning verzichtet wurde, und stattdessen das Transfer Learning verlängert wurde. Diese Vergleiche helfen dabei, den Einfluss des entwickelten Fine-Tuning Ansatzes im Gesamtkontext zu evaluieren.

#### 4.3.1 Fine-Tuning ohne Transfer Learning

Beim Fine-Tuning ohne vorheriges Transfer Learning wird das Base-Trained Modell genommen und darauf direkt das Fine-Tuning angewandt. Ziel dabei ist es zu testen, ob das Fine-Tuning auch in der Lage ist, vermehrt AXL-Kinase-Inhibitoren zu generieren, ohne ein Modell zu verwenden, dass vorher speziell auf den AXL-Kinase-Inhibitor-Datensatz trainiert worden ist.

Das Training erfolgt über 150 Epochen mit einer leicht erhöhten Lernrate von 0,0000025, um das fehlende Wissen durch das übersprungene Transfer Learning auszugleichen. In jeder Iteration werden wieder 400 Moleküle generiert und anschließend durch den DeepSMILES Predictor bewertet. Dabei bleibt das Bewertungssystem unverändert.

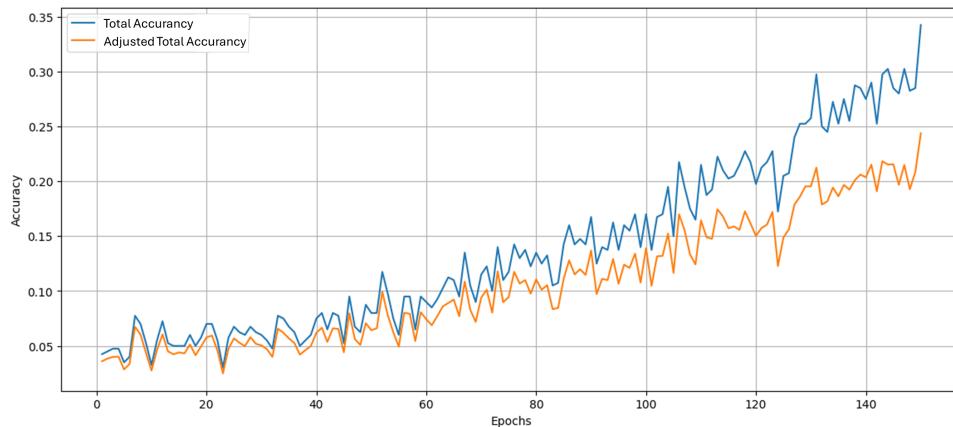


ABBILDUNG 4.13: Verlauf der Gesamtgenauigkeit der generierten Moleküle Fine-Tuning ohne Transfer Learning. Diese Abbildung zeigt, wie sich die Gesamtgenauigkeit (blau) und die korrigierte Gesamtgenauigkeit (orange), die auch die generierten ungültigen Moleküle berücksichtigt, über den gesamten Trainingszeitraum entwickeln. Gegen Ende des Trainings wird eine Differenzierung der Kurven sichtbar. Der Genauigkeitsbereich bleibt insgesamt auf einem niedrigen Niveau.

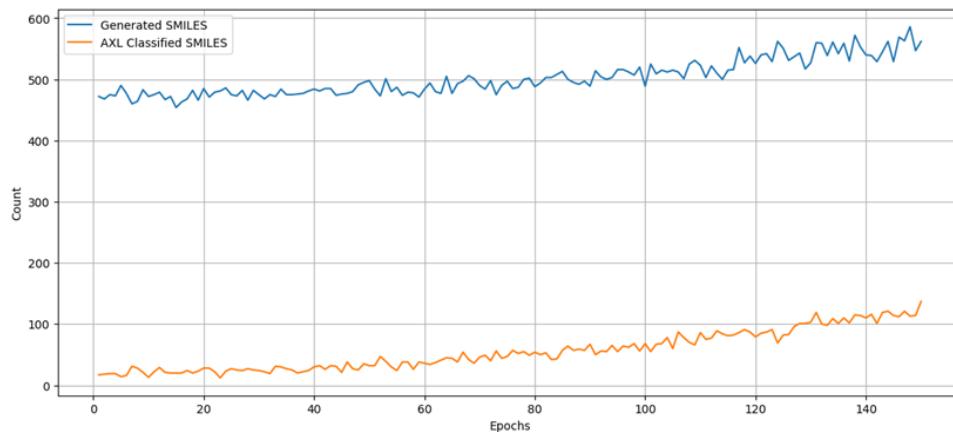
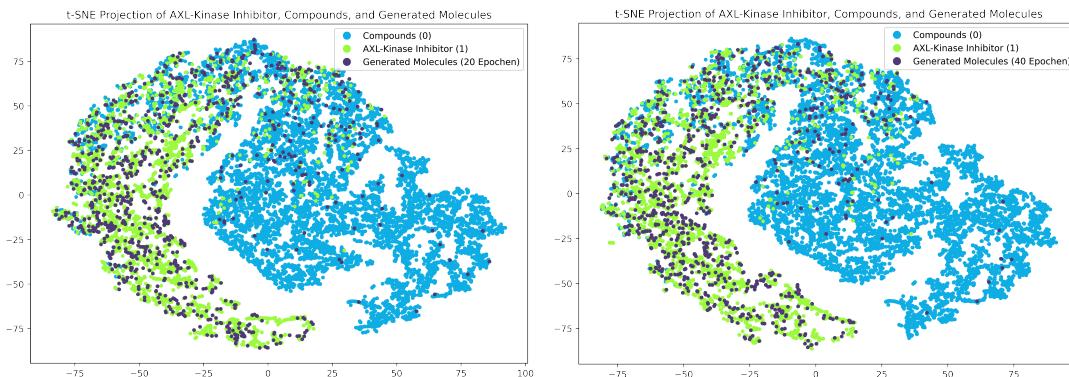


ABBILDUNG 4.14: Verlauf der Anzahl der generierten Moleküle Fine-Tuning ohne Transfer Learning. Diese Abbildung zeigt die Anzahl der pro Epoche generierten Moleküle (blau), die sowohl valide als auch ungültige Moleküle einschließt, sowie die Anzahl der korrekt als AXL-Kinase-Inhibitoren klassifizierten Moleküle (orange). Die Anzahl der zielgerichteten Moleküle steigt im Verlauf des Trainings kontinuierlich an, doch auch die Zahlen bleiben auf einem niedrigen Niveau.

Im Verlauf des Trainings zeigt sich, dass dieser Ansatz zwar Erfolge erzielt und das Modell in der Lage ist, Moleküle zu generieren, die AXL-Kinase-Inhibitoren ähneln (Abbildung 4.14). Jedoch kann es im Fine-Tuning nicht an das Leistungsniveau herankommen, welches mit einem zuvor durchgeföhrten Transfer Learning erreicht wurde (Abbildung 4.13). Der Reinforcement Learning Ansatz im Fine-Tuning verbessert das Training zwar spürbar, es kann jedoch nicht den Vorsprung durch das Transfer Learning aufholen, ohne dass zu viele invalide Moleküle entstehen.



**ABBILDUNG 4.15: T-SNE Plots nach dem verlängerten Transfer Learning ohne Fine-Tuning.** Die Abbildungen zeigen die Verteilung der generierten Moleküle (lila) im Kontext der als AXL-Kinase-Inhibitoren klassifizierten Moleküle (grün) und den als allgemeine Verbindungen klassifizierten Molekülen (blau). Links ist die Variante nach 20 Epochen Transfer Learning und rechts die Variante nach 40 Epochen zu sehen. Trotz längerem Training keine signifikanten Veränderungen in der Verteilung der generierten Moleküle zu erkennen.

### 4.3.2 Verlängertes Transfer Learning ohne Fine-Tuning

In dieser Variante wurde das Base-Trained Modell mit dem Transfer Learning über 20 Epochen hinweg auf den AXL-Kinase-Inhibitor-Datensatz trainiert. Anschließend wurden 1000 Moleküle generiert und zur Visualisierung in den klassifizierenden t-SNE Plot integriert (Abbildung 4.15). Die Visualisierung zeigt, dass mehr generierte Moleküle als AXL-Kinase-Inhibitoren klassifiziert werden, es aber trotz längerer Trainingsphase keine signifikante Veränderung in der Verteilung dieser Moleküle im AXL-Kinase-Inhibitor klassifizierenden Bereich erkennbar ist.

Um die Effekte eines längeren Trainings zu prüfen, wurde ein weiteres Experiment mit 40 Epochen durchgeführt. Auch hier wurden 1000 Moleküle generiert und im t-SNE Plot dargestellt (Abbildung 4.15). Das Ergebnis war jedoch ähnlich. Zwar nahm die Anzahl der als AXL-Kinase-Inhibitoren klassifizierten Moleküle weiter zu, aber die Verteilung im t-SNE Plot änderte sich nicht. Dies deutet darauf hin, dass das Modell zwar durch längeres Transfer Learning mehr relevante Moleküle erzeugen kann, jedoch keine wesentliche Verbesserung in der Fokussierung auf den zentralen Bereich der AXL-Kinase-Inhibitoren erreicht. Im Vergleich zu Phase 3 des Deep-SMILESGenerator, in der das Fine-Tuning angewendet wurde, gruppieren sich die Moleküle weniger dicht um den zentralen Bereich, was zeigt, dass das Fine-Tuning einen entscheidenden Vorteil in der gezielten Anpassung der Strukturen bietet.

## 4.4 Vergleich der einzelnen Trainingsphasen

Die Auswertung der jeweils 1000 generierten Moleküle in den verschiedenen Trainingsphasen verdeutlicht die Unterschiede zwischen den einzelnen Ansätzen (Tabelle 4.1).

In **Phase 1**, dem Base-Training des DeepSMILESGenerator, dass auf die allgemeine Molekülgenerierung abzielte, wurden insgesamt 959 valide Moleküle und 41 AXL-Kinase-Inhibitoren generiert. Die relativ geringe Anzahl an Zielmolekülen zeigt, dass das Modell in dieser Phase, in der es auf das breite Spektrum der Molekülstrukturen trainiert wurde, noch keine Fokussierung auf die spezifischen AXL-Kinase-Inhibitoren stattgefunden hat. Gleichzeitig war die Anzahl der invaliden Moleküle mit 203 relativ moderat.

In **Phase 2**, dem Pre-Training des DeepSMILESGenerator, welches durch das Transfer Learning auf den AXL-Kinase-Inhibitor-Datensatz geprägt ist, stieg die Anzahl der generierten AXL-Kinase-Inhibitoren signifikant auf 418. Dies verdeutlicht, dass das Modell durch das zusätzliche Training auf die zielgerichteten Moleküle fokussierter wurde. Gleichzeitig nahm die Anzahl der invaliden Moleküle leicht zu, was zeigt, dass das Modell begann, spezifischere Strukturen zu erzeugen, was zu Lasten der Validität der Moleküle führte.

In **Phase 3**, dem Fine-Tuning des DeepSMILESGenerator indem der entwickelte Reinforcement Learning Ansatz zum Einsatz kommt, konnte die Anzahl der AXL-Kinase-Inhibitoren weiter auf 660 gesteigert werden. Dabei wird sichtbar, dass die Anpassungen durch das Fine-Tuning, das Modell weiter in Richtung der gewünschten Zielmoleküle optimiert haben. Allerdings ist in dieser Phase auch die Anzahl der invaliden Moleküle auf 468 signifikant angestiegen.

**Das Fine-Tuning mit einem Reinforcement Learning Ansatz ohne vorheriges Transfer Learning** generiert mit 307 deutlich weniger Zielmoleküle als das Fine-Tuning, dass mit einem vorhergegangen Transfer Learning einhergeht. Mit 450 invaliden Moleküle steigt die Anzahl der invaliden Moleküle genauso viel wie die Anzahl der generierten Zielmoleküle, was auf die Notwendigkeit hinweist, das Modell durch vorherige Transfer Learning vorzubereiten, um eine gezieltere Generierung zu erreichen.

**Das verlängerte Transfer Learning über 40 Epochen ohne nachfolgendem Fine-Tuning** generiert mit 650 zwar fast genauso viele AXL-Kinase-Inhibitoren wie das Training mit anschließendem Fine-Tuning durch den entwickelten Reinforcement Learning Ansatz und generiert mit 224 deutlich weniger invalide Moleküle jedoch ist im dazugehörigen t-SNE Plot keine derart präzise Fokussierung in der Verteilung der Zielmoleküle zu beobachten.

TABELLE 4.1: Anzahl invalider Moleküle, Compounds und AXL-Kinase-Inhibitoren in den verschiedenen Trainingsphasen. Diese Tabelle zeigt die Verteilung der generierten Moleküle, unterteilt in valide und invalide Strukturen sowie die als AXL-Kinase-Inhibitoren klassifizierten Moleküle, in den unterschiedlichen Trainingsphasen. Die Ergebnisse verdeutlichen, wie sich die Anzahl der Zielmoleküle im Verlauf der einzelnen Trainingsphasen entwickelt haben.

	Invalid Moleküle	Compounds	AXL-Kinase-Inhibitor
<b>Phase 1</b>	203	959	41
<b>Phase 2</b>	227	582	418
<b>Phase 3</b>	468	340	660
<b>Fine-Tuning ohne Pre-Training</b>	450	693	307
<b>Transfer Learning ohne Fine-Tuning (20 Epochen)</b>	243	430	570
<b>Transfer Learning ohne Fine-Tuning (40 Epochen)</b>	224	350	650

## Kapitel 5

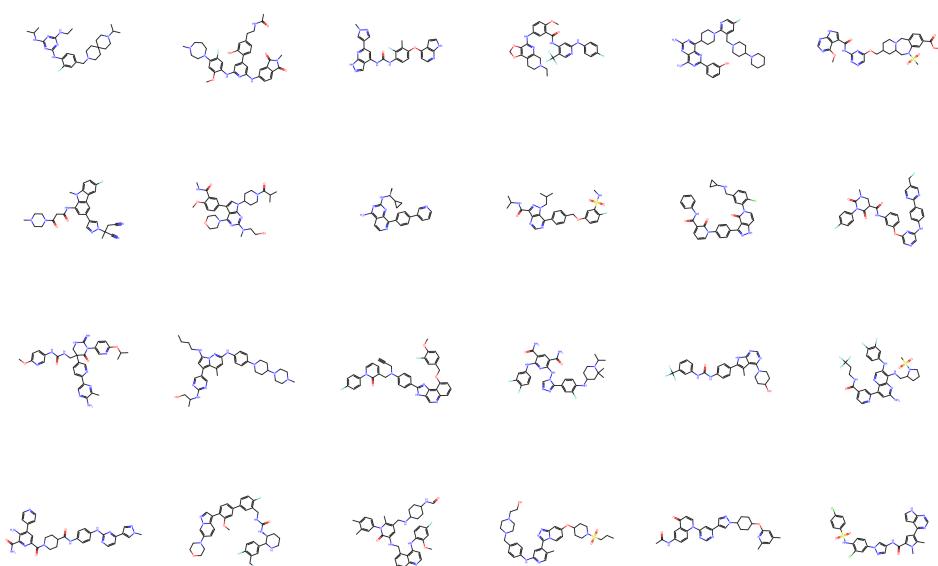
# Diskussion

In dieser Arbeit wurden zwei zentrale Forschungsfragen verfolgt. Erstens, wie gut die Wirksamkeit von Molekülen im Hinblick auf ihre potenzielle Aktivität als AXL-Kinase-Inhibitor vorhergesagt werden kann, und zweitens, wie sich spezifische Moleküle gezielt generieren lassen, die anschließend als AXL-Kinase-Inhibitoren fungieren könnten. Um diese Fragen zu beantworten, wurden der DeepSMILES-Generator und der DeepSMILESPredictor entwickelt.

Die gezeigten t-SNE Darstellungen verdeutlichen, dass der QSAR-Ansatz in Kombination mit einem MLP-Modell eine sehr gute Aufteilung der Moleküle in AXL-Kinase-Inhibitoren und Compounds erreicht. Die Klassifikationsgenauigkeit von 99,96% bei den AXL-Kinase-Inhibitor Daten und 97,23% in den Compound Daten im Training unterstützen diese Beobachtung. Besonders die ECFPs erwiesen sich in der *Feature Importance Analyse* als zentraler Faktor für die präzise Vorhersage biologischer Aktivitäten. Sie zeigen eine weitaus stärkere Relevanz als der Großteil der klassischen molekularen Deskriptoren und unterstreichen somit ihre entscheidende Rolle im Vorhersageprozess.

Die Tabelle der einzelnen Trainingsphasen zeigt, dass der DeepSMILESGenerator nach jeder Phase eine verbesserte Generierung von AXL-Kinase-Inhibitoren aufwies. Die Implementierung aus der für das Maschine Learning angepassten DeepSMILES-Notation in verbindung mit einem LSTM-Netzwerk bot eine solide Grundlage für die Kombination aus einem Transfer Learning und dem entwickelten Reinforcement Learning Ansatz. Die t-SNE-Darstellung des Fine-Tunings zeigte eine Verschiebung der generierten Moleküle in Richtung der AXL-Kinase-Inhibitor Klassifizierung, was verdeutlicht, dass die erzeugten Moleküle nicht nur chemisch valide, sondern auch funktional relevant sind. Zusätzlich nahm die Anzahl der als AXL-Kinase-Inhibitor klassifizierten Moleküle nach jeder Phase zu und erreichte in Phase 3 eine Genauigkeit bei der Generierung von AXL-Kinase-Inhibitoren von beeindruckenden 70% beziehungsweise einer korrigierten Gesamtgenauigkeit von 50%, unter Berücksichtigung der invaliden Moleküle. Allerdings gibt es auch Schwächen im Vergleich zu dem alternativen Training indem ein verlängertes Tansfer Learning ohne nachfolgendem Fine-Tuning durchgeführt wurde, generiert das Fine-Tuning des DeepSMILESGenerator deutlich mehr invalide Moleküle. Möglicherweise haben die Gewichtungen bestimmter Molekülstrukturen oder eine Überanpassung im Training zu dieser Entwicklung beigetragen. Die verstärkte Generierung invalider Moleküle nach dem entwickelten Reinforcement Learning Ansatz stellt ein unerwartetes Ergebnis dar und zeigt, dass das Modell in zukünftigen Anwendungen weiter optimiert werden muss, um die Balance zwischen Spezifität und Validität der Moleküle zu verbessern.

Nach Abschluss des mehrstufigen Trainingsprozesses konnte der DeepSMILES-Generator valide Molekülstrukturen erzeugen, die in ihrer chemischen Beschaffenheit als AXL-Kinase-Inhibitoren fungieren könnten. Einige zufällig ausgewählte generierte Moleküle wurden zur Veranschaulichung der Abbildung 5.1 dargestellt. Diese Ergebnisse unterstreichen das Potenzial des entwickelten *in silico* Workflows gezielt Moleküle zu entwerfen, die als therapeutische Wirkstoffe gegen COVID-19 eingesetzt werden könnten. Das langfristige Ziel besteht darin, die generierten Moleküle im Nachgang zu testen, um ihre Wirksamkeit und Eignung als AXL-Kinase-Inhibitoren zu bestätigen, und die Leistungsfähigkeit des entwickelten *in silico* Workflows zur Entwicklung therapeutischer Wirkstoffe zu bestätigen.



**ABBILDUNG 5.1: Generierte Moleküle des DeepSMILESGenerator.**  
Die gezeigten Moleküle wurden nach dem Trainingsprozess des DeepSMILESGenerators erstellt. Sie umfassen valide chemische Strukturen die potenziell als AXL-Kinase-Inhibitoren fungieren könnten und für eine experimentelle Validierung in einem Labor vorgesehen sind.

## Kapitel 6

# Schlussfolgerung

Der entwickelte *in silico* Workflow hat vielversprechende Ergebnisse geliefert und gezeigt, dass Maschine Learning Methoden in der Wirkstoffforschung bedeutende Beiträge zur Wirkstoffentwicklung leisten können. Insbesondere im Kontext von Krisen, wie der COVID-19-Pandemie, könnten solche Methoden eine beschleunigte Entwicklung therapeutisch relevanter Moleküle ermöglichen.

Trotz dieser vielversprechenden Ergebnisse darf nicht vergessen werden, dass es sich um ein *in silico* Verfahren handelt. Das bedeutet, dass die generierten Moleküle nicht im Labor auf ihre tatsächliche Wirksamkeit getestet wurden. Auch wenn die t-SNE-Visualisierung der generierten Moleküle vielversprechende Hinweise auf die Ähnlichkeit zu AXL-Kinase-Inhibitoren lieferte, fehlt noch die experimentelle Validierung, um die tatsächliche Eignung dieser Moleküle zu bestätigen. In diesem Zusammenhang trifft die Einschätzung von Segler et al. den Kern der Herausforderung. „Deep Learning ist kein Allheilmittel. Es reicht nicht aus, Moleküle zu generieren, die fast richtig sind, denn in der Chemie ist ein Fehlschuss so gut wie eine Meile, und die Arzneimittelforschung ist ein Nadel im Heuhaufen-Problem bei dem auch die Nadel wie Heu aussieht“ [67].

Das Zitat unterstreicht, dass maschinelles Lernen allein nicht ausreicht, um die Komplexität der Arzneimittelforschung vollständig zu erfassen. Die in dieser Arbeit entwickelten Modelle haben gezeigt, dass sie in der Lage sind, potenziell vielversprechende AXL-Kinase-Inhibitoren zu generieren. Allerdings bleibt der nächste Schritt, diese Moleküle experimentell zu testen, und ihre Wirksamkeit im Labor zu validieren.

## Anhang A

# Häufigkeit der vorkommenden Token im Compound-Datensatz

Total unique tokens: 281

Tokens and their frequencies (sorted by frequency, descending):

) : 30284579  
c : 22310613  
C : 15243365  
O : 5481005  
6 : 4006327  
=: 3584780  
N : 3181905  
n : 2211868  
5 : 1149275  
F : 787817  
- : 590471  
[C@H] : 585296  
[C@@H] : 549433  
S : 483141  
Cl : 470388  
9 : 456217  
/ : 429009  
%10 : 404917  
[nH] : 224025  
s : 221189  
o : 214808  
# : 123433  
. : 112679  
[C@] : 112447  
[O-] : 109969  
[N+] : 97778  
\ : 97199  
Br : 91249  
[C@@] : 91070  
3 : 81170  
7 : 57199  
%14 : 56311  
%13 : 51924  
8 : 48243

4: 38347  
%11: 33405  
P: 31930  
[n+]: 31009  
%12: 21455  
%17: 17442  
%15: 12643  
I: 11867  
[Na+]: 8902  
%18: 8876  
%16: 6610  
[S+]: 6347  
[Br-]: 5841  
[Cl-]: 4612  
[I-]: 3998  
[N-]: 3685  
%22: 3291  
B: 3071  
[Si]: 2787  
%20: 2683  
%19: 2138  
%21: 1974  
[2H]: 1580  
[K+]: 1278  
[Se]: 1272  
[P+]: 1081  
[C-]: 854  
[se]: 733  
%24: 683  
[Cl+3]: 597  
[Li+]: 514  
[18F]: 486  
%26: 473  
[B-]: 448  
[3H]: 442  
%25: 422  
[O]: 381  
%23: 353  
[P@C]: 347  
[S-]: 343  
[P@]: 335  
[s+]: 334  
[n-]: 280  
%28: 256  
[125I]: 256  
[PH]: 239  
[11CH3]: 239  
[As]: 218  
[S@+]: 211  
[S@@+]: 193  
[o+]: 169

[P-] : 166  
%30: 165  
%27: 146  
[N<sub>0</sub>+]: 132  
%29: 129  
[Ca+2]: 121  
[C+]: 112  
[N@O+]: 98  
%33: 91  
[Zn+2]: 83  
[I+]: 81  
[Mg+2]: 79  
%32: 79  
[OH-]: 79  
[123I]: 78  
[14C]: 75  
%36: 73  
%31: 71  
[BH<sub>2</sub>-]: 69  
[11C]: 69  
[Te]: 60  
%34: 52  
[N@O]: 50  
[se+]: 47  
[c+]: 46  
[131I]: 43  
[NH+]: 42  
[NH<sub>2</sub>+] : 41  
[te+]: 41  
[14cH]: 38  
p: 37  
[NaH]: 33  
[SeH]: 33  
[14c]: 28  
[O+]: 28  
[NH-]: 28  
%35: 27  
[14CH<sub>2</sub>]: 26  
[N<sub>0</sub>]: 26  
[Zn]: 25  
[te]: 24  
[A1]: 24  
[Al+3]: 21  
[S@O]: 21  
[SiH<sub>2</sub>]: 21  
[S<sub>0</sub>]: 20  
[14CH<sub>3</sub>]: 19  
[BH<sub>3</sub>-]: 18  
[SH]: 15  
[Ag+]: 12  
[124I]: 12

%38: 12  
[Ba+2]: 11  
[Se+]: 11  
[P@+]: 11  
[cH-]: 11  
[SiH4]: 10  
[Mg]: 10  
[19F]: 10  
[C1+]: 10  
[Sr+2]: 9  
[As+]: 8  
%40: 8  
[Ag]: 8  
[14CH]: 7  
[BH-]: 7  
[H+]: 7  
[P@O+]: 7  
[35S]: 6  
[c-]: 6  
[CH-]: 6  
[32P]: 6  
[13C]: 6  
[13CH]: 6  
[13c]: 6  
[13cH]: 6  
[SiH]: 5  
%43: 5  
[K]: 5  
[Cs+]: 5  
[17F]: 5  
%37: 5  
[SH+]: 5  
[TeH2]: 4  
[Bi+3]: 4  
[MgH2]: 4  
[14C@O]: 4  
[b-]: 4  
%39: 4  
[11CH2]: 4  
[F-]: 4  
[127I]: 3  
[PH2]: 3  
[76Br]: 3  
[14C@H]: 3  
[Br+2]: 3  
[Cl+2]: 3  
[Rb+]: 3  
%42: 3  
[18O]: 3  
[O-2]: 3  
[B]: 3

[Li]: 3  
[85Sr+2]: 2  
b: 2  
[B@-]: 2  
[S-2]: 2  
[125IH]: 2  
[I+3]: 2  
[Zn-2]: 2  
[Be+2]: 2  
%41: 2  
[75Se]: 2  
[15n]: 2  
[NH3+]: 2  
[Ra]: 2  
[TeH]: 2  
[CH]: 2  
[Na]: 2  
[nH+]: 2  
[A1-3]: 2  
[Se-]: 2  
[B@@-]: 2  
[OH+]: 2  
%52: 2  
[SH-]: 2  
[KH]: 2  
[11CH]: 2  
[Zn+]: 2  
[LiH]: 2  
%46: 2  
[IH]: 2  
[11c]: 2  
[223Ra]: 2  
[H-]: 2  
[As-]: 1  
[131Cs]: 1  
[42K+]: 1  
[Ca]: 1  
[18OH]: 1  
[I+2]: 1  
[SrH2]: 1  
[15nH]: 1  
[76BrH]: 1  
[AsH3]: 1  
[11C@@H]: 1  
[18FH]: 1  
[22Na+]: 1  
[73Se]: 1  
[124I-]: 1  
[47Ca+2]: 1  
[SH2]: 1  
[Kr]: 1

[SiH3-]: 1  
[45Ca+2]: 1  
[125I-]: 1  
[11C-]: 1  
[123Te]: 1  
[AsH]: 1  
[123IH]: 1  
[OH]: 1  
[SeH2]: 1  
[Bi]: 1  
[SiH-]: 1  
[NH4+]: 1  
[Ag-4]: 1  
[81Kr]: 1  
[Mg+]: 1  
[Ba]: 1  
[85SrH2]: 1  
%45: 1  
[Si-]: 1  
%48: 1  
[32PH]: 1  
[Si@]: 1  
[PH2+]: 1  
[127Xe]: 1  
[133Xe]: 1  
[131I-]: 1  
[14C@@]: 1  
[135I]: 1  
[Rb]: 1  
[CH2]: 1  
[82Rb+]: 1  
[150H2]: 1  
[123I-]: 1  
[C]: 1  
[82Rb]: 1  
[He]: 1  
[18F-]: 1  
[Xe]: 1  
[129Xe]: 1  
[89Sr+2]: 1  
[N]: 1  
[Cs]: 1  
[13NH3]: 1

# Literaturverzeichnis

- [1] R. Filip, R. Gheorghita Puscaselu, L. Anchidin-Norocel, M. Dimian, and W. K. Savage, "Global challenges to public health care systems during the covid-19 pandemic: A review of pandemic measures and problems," *Journal of Personalized Medicine*, vol. 12, no. 8, 2022. [Online]. Available: <https://doi.org/10.3390/jpm12081295>
- [2] J.-H. Won and H. Lee, "Can the covid-19 pandemic disrupt the current drug development practices?" *International Journal of Molecular Sciences*, vol. 22, no. 11, 2021. [Online]. Available: <https://doi.org/10.3390/ijms22115457>
- [3] V. D. Mouchlis, A. Afantitis, A. Serra, M. Fratello, A. G. Papadiamantis, V. Aidinis, I. Lynch, D. Greco, and G. Melagraki, "Advances in de novo drug design: From conventional to machine learning methods," *International Journal of Molecular Sciences*, vol. 22, no. 4, 2021. [Online]. Available: <https://doi.org/10.3390/ijms22041676>
- [4] R. Dhudum, A. Ganeshpurkar, and A. Pawar, "Revolutionizing drug discovery: A comprehensive review of ai applications," *Drugs and Drug Candidates*, vol. 3, no. 1, pp. 148–171, 2024. [Online]. Available: <https://doi.org/10.3390/ddc3010009>
- [5] C. Deane and M. Mokaya, "A virtual drug-screening approach to conquer huge chemical libraries," *Nature*, 2022. [Online]. Available: <https://doi.org/10.1038/d41586-021-03682-1>
- [6] X. Tong, X. Liu, X. Tan, X. Li, J. Jiang, Z. Xiong, T. Xu, H. Jiang, N. Qiao, and M. Zheng, "Generative models for de novo drug design," *Journal of Medicinal Chemistry*, vol. 64, no. 19, pp. 14 011–14 027, 2021. [Online]. Available: <https://doi.org/10.1021/acs.jmedchem.1c00927>
- [7] N. Brown, P. Ertl, R. Lewis, T. Luksch, D. Reker, and N. Schneider, "Artificial intelligence in chemistry and drug design," *Journal of Computer-Aided Molecular Design*, vol. 34, pp. 709–715, 2020. [Online]. Available: <https://doi.org/10.1007/s10822-020-00317-x>
- [8] C. Zhu, Y. Wei, and X. Wei, "Axl receptor tyrosine kinase as a promising anti-cancer approach: functions, molecular mechanisms and clinical applications," *Molecular cancer*, vol. 18, no. 1, p. 153, 2019. [Online]. Available: <https://doi.org/10.1186/s12943-019-1090-3>
- [9] J. D. Paccez, M. Vogelsang, M. I. Parker, and L. F. Zerbini, "The receptor tyrosine kinase axl in cancer: Biological functions and therapeutic implications," *International Journal of Cancer*, vol. 134, no. 5, pp. 1024–1033, 2014. [Online]. Available: <https://doi.org/10.1002/ijc.28246>

- [10] R. R. Naik, A. K. Shakya, S. M. Aladwan, and M. El-Tanani, "Kinase inhibitors as potential therapeutic agents in the treatment of covid-19," *Frontiers in Pharmacology*, vol. 13, 2022. [Online]. Available: <https://doi.org/10.3389/fphar.2022.806568>
- [11] R. Boytz, M. Ślabicki, S. Ramaswamy, J. J. Patten, C. Zou, C. Meng, B. L. Hurst, J. Wang, R. P. Nowak, P. L. Yang, M. Sattler, R. M. Stone, J. D. Griffin, N. S. Gray, S. Gummuluru, R. A. Davey, and E. Weisberg, "Anti-sars-cov-2 activity of targeted kinase inhibitors: Repurposing clinically available drugs for covid-19 therapy," *Journal of Medical Virology*, vol. 95, no. 1, p. e28157, 2023. [Online]. Available: <https://doi.org/10.1002/jmv.28157>
- [12] A. Varnek and I. Baskin, "Machine learning methods for property prediction in chemoinformatics: Quo vadis?" *Journal of Chemical Information and Modeling*, vol. 52, no. 6, pp. 1413–1437, 2012, pMID: 22582859. [Online]. Available: <https://doi.org/10.1021/ci200409x>
- [13] J. B. O. Mitchell, "Machine learning methods in chemoinformatics," *WIREs Computational Molecular Science*, vol. 4, no. 5, pp. 468–481, 2014. [Online]. Available: <https://doi.org/10.1002/wcms.1183>
- [14] B. J. Neves, R. C. Braga, C. C. Melo-Filho, J. T. Moreira-Filho, E. N. Muratov, and C. H. Andrade, "Qsar-based virtual screening: Advances and applications in drug discovery," *Frontiers in Pharmacology*, vol. 9, 2018. [Online]. Available: <https://doi.org/10.3389/fphar.2018.01275>
- [15] Danishuddin and A. U. Khan, "Descriptors and their selection methods in qsar analysis: paradigm for drug design," *Drug Discovery Today*, vol. 21, no. 8, pp. 1291–1302, 2016. [Online]. Available: <https://doi.org/10.1016/j.drudis.2016.06.013>
- [16] M. Hartenfeller and G. Schneider, "Enabling future drug discovery by de novo design," *WIREs Computational Molecular Science*, vol. 1, no. 5, pp. 742–759, 2011. [Online]. Available: <https://doi.org/10.1002/wcms.49>
- [17] M. Hartenfeller, M. Eberle, P. Meier, C. Nieto-Oberhuber, K.-H. Altmann, G. Schneider, E. Jacoby, and S. Renner, "A collection of robust organic synthesis reactions for in silico molecule design," *Journal of Chemical Information and Modeling*, vol. 51, no. 12, pp. 3093–3098, 2011, pMID: 22077721. [Online]. Available: <https://doi.org/10.1021/ci200379p>
- [18] M. Hartenfeller, H. Zettl, M. Walter, M. Rupp, F. Reisen, E. Proschak, S. Weggen, H. Stark, and G. Schneider, "Dogs: reaction-driven de novo design of bioactive compounds," *PLoS computational biology*, vol. 8, no. 2, p. e1002380, 2012. [Online]. Available: <https://doi.org/10.1371/journal.pcbi.1002380>
- [19] M. H. S. Segler and M. P. Waller, "Modelling chemical reasoning to predict and invent reactions," *Chemistry – A European Journal*, vol. 23, no. 25, pp. 6118–6128, 2017. [Online]. Available: <https://doi.org/10.1002/chem.201604556>
- [20] ——, "Neural-symbolic machine learning for retrosynthesis and reaction prediction," *Chemistry – A European Journal*, vol. 23, no. 25, pp. 5966–5971, 2017. [Online]. Available: <https://doi.org/10.1002/chem.201605499>

- [21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. [Online]. Available: <https://doi.org/10.1162/neco.1997.9.8.1735>
- [22] N. O'Boyle and A. Dalke, "Deepsmiles: an adaptation of smiles for use in machine-learning of chemical structures," *ChemRxiv*, 2018. [Online]. Available: [10.26434/chemrxiv.7097960.v1](https://doi.org/10.26434/chemrxiv.7097960.v1)
- [23] P. Gupta, P. Malhotra, L. Vig, and G. Shroff, "Transfer learning for clinical time series analysis using recurrent neural networks," *arXiv preprint arXiv:1807.01705*, 2018. [Online]. Available: <https://doi.org/10.48550/arXiv.1807.01705>
- [24] C. Olah. (2015) Understanding lstms. [Online]. Available: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>
- [25] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *Journal of artificial intelligence research*, vol. 4, pp. 237–285, 1996. [Online]. Available: <https://doi.org/10.1613/jair.301>
- [26] D. Rogers and M. Hahn, "Extended-connectivity fingerprints," *Journal of Chemical Information and Modeling*, vol. 50, no. 5, pp. 742–754, 2010, pMID: 20426451. [Online]. Available: <https://doi.org/10.1021/ci100050t>
- [27] W. J. Wiswesser, "Historic development of chemical notations," *Journal of chemical information and computer sciences*, vol. 25, no. 3, pp. 258–263, 1985. [Online]. Available: <https://doi.org/10.1021/ci00047a023>
- [28] D. Weininger, "Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules," *Journal of chemical information and computer sciences*, vol. 28, no. 1, pp. 31–36, 1988. [Online]. Available: <https://doi.org/10.1021/ci00057a005>
- [29] M. Krenn, F. Häse, A. Nigam, P. Friederich, and A. Aspuru-Guzik, "Self-referencing embedded strings (selfies): A 100representation," *Machine Learning: Science and Technology*, vol. 1, no. 4, p. 045024, oct 2020. [Online]. Available: <https://doi.org/10.1088/2632-2153/aba947>
- [30] A. Sherstinsky, "Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network," *Physica D: Nonlinear Phenomena*, vol. 404, p. 132306, 2020. [Online]. Available: <https://doi.org/10.1016/j.physd.2019.132306>
- [31] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, pp. 1–40, 2016. [Online]. Available: <https://doi.org/10.1186/s40537-016-0043-6>
- [32] Y. Hua, M. Sevagnani, D. Yi, A. Birnie, and S. McAslan, "Fine-grained rnn with transfer learning for energy consumption estimation on evs," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 11, pp. 8182–8190, 2022. [Online]. Available: <https://doi.org/10.1109/TII.2022.3143155>
- [33] Y. Li, "Deep reinforcement learning: An overview," *arXiv*, 2018. [Online]. Available: <https://doi.org/10.48550/arXiv.1701.07274>

- [34] H. Taud and J. Mas, *Multilayer Perceptron (MLP)*. Cham: Springer International Publishing, 2018, pp. 451–455. [Online]. Available: [https://doi.org/10.1007/978-3-319-60801-3\\_27](https://doi.org/10.1007/978-3-319-60801-3_27)
- [35] M. A. Nielsen, *Neural networks and deep learning*. Determination press San Francisco, CA, USA, 2015, vol. 25. [Online]. Available: <http://neuralnetworksanddeeplearning.com/>
- [36] C. Olah. (2014) Neural networks, manifolds, and topology. [Online]. Available: <https://colah.github.io/posts/2014-03-NN-Manifolds-Topology/>
- [37] R. Burbidge, M. Trotter, B. Buxton, and S. Holden, “Drug design by machine learning: support vector machines for pharmaceutical data analysis,” *Computers Chemistry*, vol. 26, no. 1, pp. 5–14, 2001. [Online]. Available: [https://doi.org/10.1016/S0097-8485\(01\)00094-8](https://doi.org/10.1016/S0097-8485(01)00094-8)
- [38] M. A. Lill, “Multi-dimensional qsar in drug discovery,” *Drug Discovery Today*, vol. 12, no. 23, pp. 1013–1017, 2007. [Online]. Available: <https://doi.org/10.1016/j.drudis.2007.08.004>
- [39] P. M. Andersson, M. Sjöström, S. Wold, and T. Lundstedt, “Comparison between physicochemical and calculated molecular descriptors,” *Journal of Chemometrics: A Journal of the Chemometrics Society*, vol. 14, no. 5-6, pp. 629–642, 2000. [Online]. Available: [https://doi.org/10.1002/1099-128X\(200009/12\)14:5/6<629::AID-CEM606>3.0.CO;2-M](https://doi.org/10.1002/1099-128X(200009/12)14:5/6<629::AID-CEM606>3.0.CO;2-M)
- [40] M. Randic, “Characterization of molecular branching,” *Journal of the American Chemical Society*, vol. 97, no. 23, pp. 6609–6615, 1975. [Online]. Available: <https://doi.org/10.1021/ja00856a001>
- [41] A. K. Ghose, V. N. Viswanadhan, and J. J. Wendoloski, “Prediction of hydrophobic (lipophilic) properties of small organic molecules using fragmental methods: an analysis of alogp and clogp methods,” *The Journal of Physical Chemistry A*, vol. 102, no. 21, pp. 3762–3772, 1998. [Online]. Available: <https://doi.org/10.1021/jp980230o>
- [42] S. Prasanna and R. J. DoerkSEN, “Topological polar surface area: a useful descriptor in 2d-qsar,” *Current medicinal chemistry*, vol. 16, no. 1, pp. 21–41, 2009. [Online]. Available: <https://doi.org/10.2174/092986709787002817>
- [43] C. A. Lipinski, F. Lombardo, B. W. Dominy, and P. J. Feeney, “Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings,” *Advanced Drug Delivery Reviews*, vol. 23, no. 1, pp. 3–25, 1997, in Vitro Models for Selection of Development Candidates. [Online]. Available: [https://doi.org/10.1016/S0169-409X\(96\)00423-1](https://doi.org/10.1016/S0169-409X(96)00423-1)
- [44] C. Yung-Chi and W. H. Prusoff, “Relationship between the inhibition constant ( $k_i$ ) and the concentration of inhibitor which causes 50 per cent inhibition ( $i_{50}$ ) of an enzymatic reaction,” *Biochemical Pharmacology*, vol. 22, no. 23, pp. 3099–3108, 1973. [Online]. Available: [https://doi.org/10.1016/0006-2952\(73\)90196-2](https://doi.org/10.1016/0006-2952(73)90196-2)
- [45] S. N. A. Bukhari, M. A. Elsherif, K. Junaid, H. Ejaz, P. Alam, A. Samad, R. D. Jawarkar, and V. H. Masand, “Perceiving the concealed and unreported pharmacophoric features of the 5-hydroxytryptamine receptor using balanced qsar analysis,” *Pharmaceuticals*, vol. 15, no. 7, 2022. [Online]. Available: <https://doi.org/10.3390/ph15070834>

- [46] C. C. G. Inc. (2008) Quasar-descriptor. [Online]. Available: <https://cadaster.eu/sites/cadaster.eu/files/challenge/descr.htm>
- [47] L. H. Hall and L. B. Kier, *The Molecular Connectivity Chi Indexes and Kappa Shape Indexes in Structure-Property Modeling*. John Wiley Sons, Ltd, 1991, pp. 367–422. [Online]. Available: <https://doi.org/10.1002/9780470125793.ch9>
- [48] S. A. Wildman and G. M. Crippen, “Prediction of physicochemical parameters by atomic contributions,” *Journal of Chemical Information and Computer Sciences*, vol. 39, no. 5, pp. 868–873, 1999. [Online]. Available: <https://doi.org/10.1021/ci990307l>
- [49] J. Gasteiger and M. Marsili, “Iterative partial equalization of orbital electronegativity—a rapid access to atomic charges,” *Tetrahedron*, vol. 36, no. 22, pp. 3219–3228, 1980. [Online]. Available: [https://doi.org/10.1016/0040-4020\(80\)80168-2](https://doi.org/10.1016/0040-4020(80)80168-2)
- [50] D. F. Veber, S. R. Johnson, H.-Y. Cheng, B. R. Smith, K. W. Ward, and K. D. Kopple, “Molecular properties that influence the oral bioavailability of drug candidates,” *Journal of Medicinal Chemistry*, vol. 45, no. 12, pp. 2615–2623, 2002, pMID: 12036371. [Online]. Available: <https://doi.org/10.1021/jm020017n>
- [51] T. J. Ritchie and S. J. Macdonald, “The impact of aromatic ring count on compound developability – are too many aromatic rings a liability in drug design?” *Drug Discovery Today*, vol. 14, no. 21, pp. 1011–1020, 2009. [Online]. Available: <https://doi.org/10.1016/j.drudis.2009.07.014>
- [52] W. Wei, S. Cherukupalli, L. Jing, X. Liu, and P. Zhan, “Fsp3: A new parameter for drug-likeness,” *Drug Discovery Today*, vol. 25, no. 10, pp. 1839–1845, 2020. [Online]. Available: <https://doi.org/10.1016/j.drudis.2020.07.017>
- [53] G. Landrum and other RDKit contributors. (2024) rdkit.chem.descriptors module. [Online]. Available: <https://www.rdkit.org/docs/source/rdkit.Chem.Descriptors.html>
- [54] J. Clayden, N. Greeves, and S. Warren, *Organic chemistry*. Oxford University Press, USA, 2012.
- [55] A. T. Balaban, “Highly discriminating distance-based topological index,” *Chemical Physics Letters*, vol. 89, no. 5, pp. 399–404, 1982. [Online]. Available: [https://doi.org/10.1016/0009-2614\(82\)80009-2](https://doi.org/10.1016/0009-2614(82)80009-2)
- [56] S. H. Bertz, “The first general index of molecular complexity,” *Journal of the American Chemical Society*, vol. 103, no. 12, pp. 3599–3601, 1981. [Online]. Available: <https://doi.org/10.1021/ja00402a071>
- [57] P. Contributor. (2023) Pytorch documentation. [Online]. Available: <https://pytorch.org/docs/stable/index.html>
- [58] N. Corporation. (2024) Nvidia cuda. [Online]. Available: <https://docs.nvidia.com/cuda/doc/index.html>
- [59] EMBL-EBI. (2018) Chemb3D. [Online]. Available: <https://chembl.gitbook.io/chembl-interface-documentation/about>

- [60] N. I. of Health (NIH). (2024) Pubchem. [Online]. Available: <https://pubchem.ncbi.nlm.nih.gov/docs/about>
- [61] W. Tendong, P. Lebrun, and B. Verbist, "Controlling the reproducibility of ac50 estimation during compound profiling through bayesian -expectation tolerance intervals," *SLAS Discovery*, vol. 25, no. 9, pp. 1009–1017, 2020. [Online]. Available: <https://doi.org/10.1177/2472555220918201>
- [62] C. Cardie and N. Howe, "Improving minority class prediction using case-specific feature weights," 1997, computer Science: Faculty Publications, Smith College, Northampton, MA. [Online]. Available: [https://scholarworks.smith.edu/csc\\_facpubs/102](https://scholarworks.smith.edu/csc_facpubs/102)
- [63] M. S. Shelke, P. R. Deshmukh, and V. K. Shandilya, "A review on imbalanced data handling using undersampling and oversampling technique," *Int. J. Recent Trends Eng. Res*, vol. 3, no. 4, pp. 444–449, 2017. [Online]. Available: [https://www.academia.edu/71149653A\\_Review\\_on\\_Imbalanced\\_Data\\_Handling\\_Using\\_Undersampling\\_and\\_Oversampling\\_Technique](https://www.academia.edu/71149653A_Review_on_Imbalanced_Data_Handling_Using_Undersampling_and_Oversampling_Technique)
- [64] L. Taylor and G. Nitschke, "Improving deep learning with generic data augmentation," in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2018, pp. 1542–1547. [Online]. Available: <https://doi.org/10.1109/SSCI.2018.8628742>
- [65] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008. [Online]. Available: <https://jmlr.org/papers/v9/vandermaaten08a.html>
- [66] Z. Xu. (2024) Prevent the vanishing gradient problem with lstm. [Online]. Available: <https://www.baeldung.com/cs/lstm-vanishing-gradient-prevention>
- [67] M. H. S. Segler, T. Kogej, C. Tyrchan, and M. P. Waller, "Generating focused molecule libraries for drug discovery with recurrent neural networks," *ACS Central Science*, vol. 4, no. 1, pp. 120–131, 2018, pMID: 29392184. [Online]. Available: <https://doi.org/10.1021/acscentsci.7b00512>