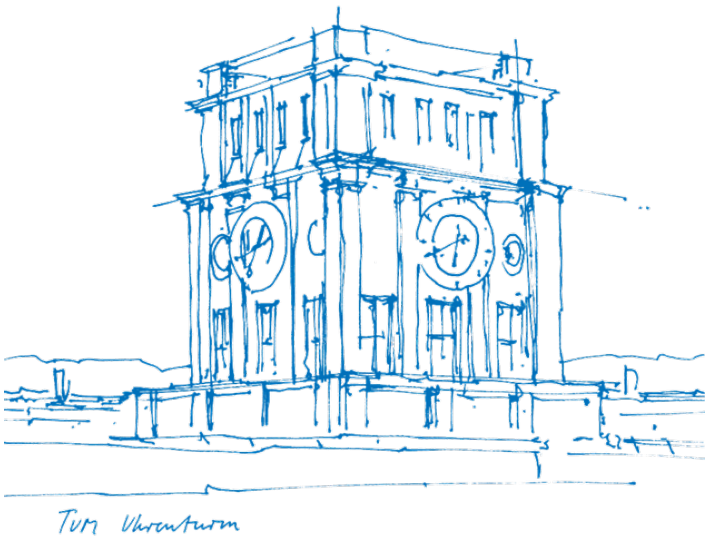
TUM School of Life Sciences
Technical University of Munich

Current Limitations of Text-based Molecular Representations for Machine Learning in Small Molecule Drug Discovery

Peter Bart Rudolf Hartog



TUM School of Life Sciences
Technische Universität München

Current Limitations of Text-based Molecular Representations for Machine Learning in Small Molecule Drug Discovery

Peter Bart Rudolf Hartog

Vollständiger Abdruck der von der TUM School of Life Sciences der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitz:
Prof. Dr. Markus List

Prüfer der Dissertation:
1. Prof. Dr. Dr. Fabian Theis
2. Priv.-Doz. Dr. Igor V. Tetko

Die Dissertation wurde am 10.03.2025 bei der Technischen Universität München eingereicht und durch die TUM School of Life Sciences am 28.04.2025 angenommen.

Abstract

Text-based molecular representations have become commonplace in machine learning (ML) applications in drug discovery, offering compatibility with established natural language processing (NLP) methods and scalability in computational frameworks. However, for ML to continue driving advancements, its methodologies must be both efficient and interpretable. Molecules can be represented in a variety of way, including through characteristics, image-based, graph-based or text-based. The most-used text-based molecular representations are SMILES (Simplified Molecular Input Line Entry System)-based molecular representations, as it maintains human readability while being compatible with chemical software. SMILES can represent the same molecule in different ways and the canonical representations are highly sensitive to atom ordering, leading to similar molecules having varying representations. To alleviate variations in SMILES representations, and improve model accuracy, SMILES-based representations first pre-trained in an unsupervised manner before fine-tuning to smaller tasks. These pre-trained models are often significantly larger and more complex, sometimes resulting in prohibitively large compute times. In the field of computer-aided synthesis planning (CASP), SMILES-based models have achieved state-of-the-art performance. But as the field progresses to larger route-finding systems, in which the model is queried multiple times, speed has become a significant issue.

This cumulative thesis investigates if current explainable artificial intelligence (XAI) methods are faithful to the internal model reasoning of these pre-trained text-based molecular representations, and how these representations can be successfully integrated into larger systems.

XAI promises to faithfully explain internal ML reasoning in a human-understandable format. To validate XAI methods, explanations are often compared to human expert opinions for validity. SMILES-based molecular representations are non-unique, which means we can test whether explanations for the same molecule in different scenarios. Using this technique, inspired by rotations from the field of image-based ML, for the first time applied to SMILES, I probe the consistency of explanations for computational toxicity prediction models. Results revealed significant variability in explanations across representations of the same molecule, different XAI methods, and alignment with expert-derived structural alerts, often reflecting tokenization artifacts rather than learned parameters. The study highlights a need to validate XAI methods to ensure the faithfulness of model explanations.

Furthermore, SMILES-based retrosynthesis models have revolutionized the field of CASP, offering more variety in their predictions. Currently, these single-step retrosynthesis models are used in a larger multi-step CASP system. However, their computational efficiency, as opposed to significantly smaller reaction classification models, have halted their successful integration into pharmaceutical production. To explore possible increases in computational efficiency, I optimized single-step retrosynthesis models using alternative Transformer architectures, knowledge distillation, and hyper-parameter tuning. While single-step improvements reduced inference times, the results emphasized that multi-step retrosynthesis efficiency depends more on prediction diversity and exploration than on single-step model speed. These findings point to the importance of balancing computational resources with the quality of model predictions in synthetic planning pipelines, and tailoring larger systems to the ML models used within.

The research identifies limitations in SMILES representations, particularly their lack of structural invariance, and advocates integration of alternative modalities such as graph-based and image-based representations alongside text-based representations to improve interpretability and stability. Future directions propose integrating multi-modal architectures to unify information across diverse molecular formats and exploring emerging modalities like proteins and larger bio-molecules to address the expanding landscape of drug discovery challenges. Furthermore, to further the use of larger ML models in overarching systems, these systems will need to be optimized to use the internal models rather then using them passively as oracles, through a combination of reinforcement learning and agentic work-

flows. This work provides actionable strategies to improve both the interpretability and efficiency of text-based ML systems in cheminformatics.

Zusammenfassung

Textbasierte molekulare Repräsentationen sind in maschinellen Lernanwendungen (ML) zur Wirkstoffforschung weit verbreitet, da sie mit etablierten Methoden der natürlichen Sprachverarbeitung (NLP) kompatibel und in rechnergestützten Frameworks skalierbar sind. Damit ML weiterhin Fortschritte vorantreiben kann, müssen seine Methoden jedoch sowohl effizient als auch interpretierbar sein. Moleküle können auf verschiedene Weise dargestellt werden, darunter durch Merkmale, bildbasierte, graphenbasierte oder textbasierte Ansätze. Die am häufigsten verwendeten textbasierten molekularen Repräsentationen basieren auf SMILES (Simplified Molecular Input Line Entry System), da sie die menschliche Lesbarkeit bewahren und gleichzeitig mit chemischer Software kompatibel sind. SMILES kann dasselbe Molekül auf unterschiedliche Weise darstellen, und kanonische Repräsentationen sind hochsensitiv gegenüber der Atomreihenfolge, sodass ähnliche Moleküle unterschiedliche Repräsentationen erhalten. Um diese Variationen in SMILES-Darstellungen zu verringern und die Modellgenauigkeit zu verbessern, werden SMILES-basierte Repräsentationen zunächst in einer unüberwachten Weise vortrainiert, bevor sie für spezifischere Aufgaben feinabgestimmt werden. Diese vortrainierten Modelle sind oft erheblich größer und komplexer, was manchmal zu untragbar langen Rechenzeiten führt. Im Bereich der rechnergestützten Syntheseplanung (CASP) haben SMILES-basierte Modelle eine führende Leistung erzielt. Doch mit der Weiterentwicklung des Feldes hin zu größeren Routenfindungssystemen, bei denen das Modell mehrfach abgefragt wird, ist Geschwindigkeit zu einem entscheidenden Problem geworden.

Diese kumulative Dissertation untersucht, ob aktuelle Methoden der erklärbaren Künstlichen Intelligenz (XAI) die internen Entscheidungsprozesse dieser vortrainierten textbasierten molekularen Repräsentationen getreu wiedergeben und wie diese Repräsentationen erfolgreich in größere Systeme integriert werden können.

XAI verspricht, interne ML-Entscheidungen in einer für Menschen verständlichen Form zu erklären. Zur Validierung von XAI-Methoden werden Erklärungen häufig mit den Einschätzungen menschlicher Experten verglichen. Da SMILES-basierte molekulare Repräsentationen nicht eindeutig sind, können wir testen, ob Erklärungen für dasselbe Molekül in unterschiedlichen Szenarien konsistent bleiben. Mithilfe dieser Technik, inspiriert von Rotationen aus dem Bereich des bildbasierten ML und erstmals auf SMILES angewandt, untersuche ich die Konsistenz von Erklärungen für computergestützte Modelle zur Toxizitätsvorhersage. Die Ergebnisse zeigen eine erhebliche Variabilität in den Erklärungen über verschiedene Repräsentationen desselben Moleküls hinweg, zwischen verschiedenen XAI-Methoden und in der Übereinstimmung mit von Experten abgeleiteten strukturellen Warnsignalen, wobei sich oft Tokenisierungsartefakte anstelle gelernter Parameter widerspiegeln. Die Studie unterstreicht die Notwendigkeit, XAI-Methoden zu validieren, um die Vertrauenswürdigkeit von Modellerklärungen sicherzustellen.

Darüber hinaus haben SMILES-basierte Retrosynthesemodelle das CASP-Feld revolutioniert, indem sie eine größere Vielfalt an Vorhersagen ermöglichen. Derzeit werden diese einstufigen Retrosynthesemodelle in ein größeres mehrstufiges CASP-System eingebettet. Ihre Rechenleistung ist jedoch im Vergleich zu wesentlich kleineren Reaktionsklassifikationsmodellen ein Hindernis für ihre erfolgreiche Integration in die pharmazeutische Produktion. Um mögliche Effizienzsteigerungen zu untersuchen, habe ich einstufige Retrosynthesemodelle durch alternative Transformer-Architekturen, Wissensdistillation und Hyperparameteroptimierung optimiert. Während einzelne Verbesserungen die Inferenzzeiten verringerten, zeigten die Ergebnisse, dass die Effizienz der mehrstufigen Retrosynthese stärker von der Vorhersagevielfalt und -exploration abhängt als von der Geschwindigkeit einzelner Modelle. Diese Erkenntnisse verdeutlichen die Bedeutung eines ausgewogenen Einsatzes von Rechenressourcen und

der Qualität der Modellvorhersagen in Syntheseplanungs-Pipelines sowie die Notwendigkeit, größere Systeme an die verwendeten ML-Modelle anzupassen.

Die Forschung identifiziert Einschränkungen in SMILES-Repräsentationen, insbesondere deren mangelnde strukturelle Invarianz, und plädiert für die Integration alternativer Modalitäten wie graphenbasierter und bildbasierter Repräsentationen neben textbasierten Ansätzen, um Interpretierbarkeit und Stabilität zu verbessern. Zukünftige Entwicklungen schlagen die Integration multimodaler Architekturen vor, um Informationen aus verschiedenen molekularen Formaten zu vereinen, sowie die Erforschung neuer Modalitäten wie Proteinen und größeren Biomolekülen, um die wachsenden Herausforderungen in der Wirkstoffforschung zu bewältigen. Darüber hinaus müssen übergeordnete Systeme optimiert werden, um größere ML-Modelle nicht nur passiv als Orakel zu nutzen, sondern aktiv in Entscheidungsprozesse einzubinden – durch eine Kombination aus Reinforcement Learning und agentenbasierten Workflows. Diese Arbeit bietet konkrete Strategien zur Verbesserung sowohl der Interpretierbarkeit als auch der Effizienz textbasierter ML-Systeme in der Cheminformatik.

List of Publications

This thesis is a publication-based thesis and is built on the following core publications as a main author. I, Peter Hartog, am the main authors of the below listed manuscripts [1, 2]. Summaries of these articles and descriptions of my contributions can be found in section 3.1.

Publications in the context of my doctoral thesis as main author in peer-reviewed journals as listed below:

Using test-time augmentation to investigate explainable AI: inconsistencies between method, model and human intuition

Peter BR Hartog, Fabian Krüger, Samuel Genheden, Igor V Tetko

Journal of Cheminformatics, 16(1), 39. 10.1186/s13321-024-00824-1
(See also publication [1] in the bibliography and in Appendix A)

Investigations into the efficiency of computer-aided synthesis planning

Peter BR Hartog, Annie M Westerlund, Igor V Tetko, Samuel Genheden

Journal of Chemical Information and Modeling. 10.1021/acs.jcim.4c01821
(See also publication [2] in the bibliography and in Appendix B)

Further publications performed during the doctorate but not included in this thesis:

Registries in Machine Learning-Based Drug Discovery: A Shortcut to Code Reuse

Peter BR Hartog*, Emma Svensson*, Lewis Mervin, Samuel Genheden, Ola Engkvist, Igor V Tetko

AI in Drug Discovery. AIDD 2024. Lecture Notes in Computer Science, vol 14894. Springer, Cham. 10.1007/978-3-031-72381-0_9
(See also publication [3] in the bibliography)

The openOCHEM consensus model is the best-performing open-source predictive model in the First EUOS/SLAS joint compound solubility challenge

Andrea Hunklinger, **Peter BR Hartog**, Martin Šícho, Guillaume Godin, Igor V Tetko

SLAS Discovery 29 (2), 100144 10.1016/j.slasd.2024.01.005
(See also publication [4] in the bibliography)

* *These authors contributed equally*

Acknowledgements

Completing this doctoral journey has been truly transformative, made possible only through extensive support from loved ones, guidance by both colleagues, supervisors and students, and inspiration from numerous individuals both scientific and personal.

First and foremost, I would like to thank my partner for her support, patience and the life we are building together. Your love and patience have been the bedrock of my resilience and patience to others. I could not have done this without you, always my equal as two peas in a pod, navigating this journey together. I am endlessly grateful for your unwavering support and belief in me.

Furthermore, I am grateful to my three supervisors. To my primary supervisor, Dr. Igor Tetko, for their steadfast support, invaluable mentorship, and for always challenging me to strive for excellence. To my industrial supervisor, Dr. Samuel Genheden, for providing me with a unique perspective from an industrial setting, guidance in navigating the field and collaborations, and for being a dear friend. And to my academic supervisor, Prof. Dr. Dr. Fabian Theis for their critical feedback and invaluable insights that shaped this thesis.

I extend my heartfelt thanks to the institutions that hosted me during my doctoral journey. At Helmholtz Munich, I found a stimulating academic environment that nurtured my scientific curiosity. At AstraZeneca, I gained invaluable industry experience and developed a deep appreciation for applied research, with an amazing team at Molecular AI. I also wish to thank Günter Klambauer and the members of his group at Johannes Kepler University Linz, Austria, for integrating me into the group my secondment.

To my colleagues and friends at Helmholtz Munich and AstraZeneca, thank you for your camaraderie, stimulating discussions, and for making the challenges of research more enjoyable. A special thanks to Varvara and Paula, which were instrumental in making our group great in STB Helmholtz, my students Andi, Fabian, Isak and Erik for the lessons they helped me learn, all AIDD fellows and to all my co-authors, Emma, Annie and Fabian for their invaluable contributions and support.

On a personal note, I am deeply thankful to my family and friends for their unwavering belief in me. From my childhood friends back home in the Netherlands, Tom, Elisa, Jim, Konijn, Ilena, Jelte and Sanaz for sticking with me even though I'm far away. For my new friends in Munich, Arvid, Joël, Maria, Jenny, Ruben, Ernst and Sophie for our time together making Munich a home. And finally to my family, Linda, Arno, Maarten, Wilma, Saskia, Frannie, Nicki en Kasper, for their love and support (including three cross-country moves).

This thesis is the culmination of collective effort, and I am deeply indebted to everyone who has been part of this journey. Thank you for believing in me and for shaping my path as a researcher.

The project(s) leading to this thesis had received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie Innovative Training Network - European Industrial Doctorate No 956832, "Advanced machine learning for Innovative Drug Discovery" (AIDD). The thesis reflects only my view and neither the European Commission nor the Research Executive Agency (REA) are responsible for any use that may be made of the information it contains.

Contents

Abbreviations

ADMET	Adsorption, distribution, metabolism, excretion and toxicity
AI	Artificial Intelligence
BERT	Bidirectional Encoder Representations from Transformers
BART	Bidirectional and Auto-Regressive Transformers
CEM	Concept Embedding Models
CASP	Computer-aided Synthesis Planning
DEL	DNA-encoded Library
DL	Deep Learning
DMTA	Design-make-test-analyze
DT	Decision Tree
EMA	European Medicines Agency
FGA	Functional Group Interconversion
FGI	Functional Group Addition
FDA	United States Food and Drug Administration
HTS	High-throughput screening
IG	Integrated Gradients
InChI	International chemical identifier
IND	Investigational New Drug
KD	Knowledge Distillation
LIME	Local Interpretable Model-agnostic Explanations
LSTM	Long Short-Term Memory
MAE	Mean Square Error
MCC	Matthews Correlation Coefficient
MCTS	Monte Carlo Tree Search
MD	Molecular Dynamics
ML	Machine Learning
MLP	Multi-layer Perceptron

NLP Natural Language Processing

OoC Organ-on-chips

PDD Phenotypic Drug Discovery

PKPD Pharmacodynamics and pharmacokinetics

QSAR Quantitative Structure-Activity Relationship

RAG Retrieval-Augmented generation

RLHF Reinforcement Learning with Human Feedback

RMSE Root Mean Absolute Error

RNN Recurrent neural networks

ROC AUC Area Under Receiving Operating Characteristic Curve

SAE Sparse auto-encoder

SELFIES Self-Referencing Embedded Strings

Seq2Seq Sequence-to-sequence

SHAP Shapley Additive Explanations

SMARTS SMILES Arbitrary Target Specification

SMILES Simplified Molecular Input Line Entry System

SBDD Structure-based Drug Design

TDD Target-based Drug Discovery

USPTO United States Patent and Trademark Office Dataset, extracted by Lowe[5]

XAI Explainable AI

1 Introduction

> When we try to pick out anything by itself, we find it hitched to everything else in the Universe
>
> John Muir, *My First Summer in the Sierra*

1.1 Impact of Machine Learning in Drug Discovery

There is a decline in the number of molecules successfully reaching the clinic started in the 1980s [6]. Recently, studies have indicated that less than 1 in 10,000 compounds identified as potential drugs make it to clinical trials [7]. One argument for this decline is historical discoveries were drugs that were easier to find [8]. This was also echoed by the European SPC Manufacturing Waiver (Regulation (EU) 2019/933), which recently decreased the extended patent protection for simple medicinal products such as small molecule drugs, moving the entire industry to new(er) modalities [9]. The field of small molecule drug discovery is changing rapidly, focusing on faster discovery on more difficult targets. To understand the field, the drug discovery and development pipeline is presented as it is traditionally explained. Subsequently, notable advancements in the drug discovery pipeline relevant to small molecule drug research are discussed, focusing on advancements that involve the use of machine learning (ML).

1.1.1 The Traditional Drug Discovery and Development Pipeline

The drug discovery and development pipeline encompasses the entire path from molecule to approved medicine [10]. Traditionally, the pipeline starts with the discovery of a viable target and ends with an approved drug (Figure 1.1).

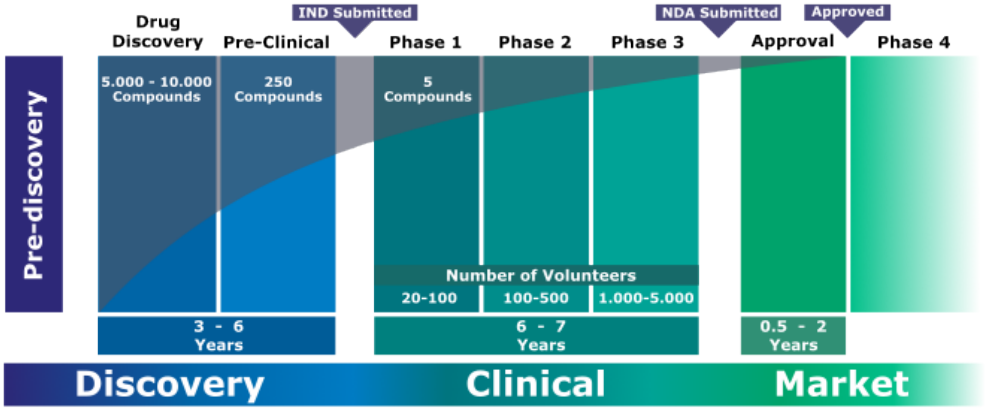


Figure 1.1 Average Drug Discovery and Development Pipeline for one approved drug spanning 15 years. On average 10.000 compounds are analyzed for every single approved drug. *Adapted from PharmaSource [11].*

Pre-discovery The drug discovery and development pipeline starts with pre-discovery [12]. This stage focuses on understanding the underlying biology of a disease. Researchers identify potential drug targets, such as proteins or genes that play a critical role in disease progression. Through steps of target discovery and validation, researchers attempt to explain disease patterns and progression using biological functions and pathways. These functions and pathways are subsequently used to identify possible targets to slow, halt or reverse disease progression. Through target identification and validation, the goal is to ensure that modulating these targets can provide a therapeutic benefit. This stage is crucial for laying the foundation for all subsequent stages.

Drug discovery The drug discovery and development pipeline begins once a valid target has been identified [10]. Methods employing rough screening of large amounts of possible molecules, including high-throughput screening (HTS) [13] or computational methods [14] are used to identify hits, which are molecules that show activity against the target. Identified hits are then further developed and validated in more reliable and time-intensive assays to produce lead compounds. Assay validation includes efficacy (i.e., does the drug have the intended effect on the target), but also includes safety and investigations into the desired properties of these possible therapeutic agents. The most promising lead therapeutic agents are then optimized for efficacy, specificity, safety, and often directed to chemical spaces which can be patented.

The drug discovery pipeline may take up to five years to produce a selection of candidate drugs. This stage involves multiple iterations of Design-Make-Test-Analyze (DMTA) cycles to generate potential

therapeutic candidates from initial hits (Figure 1.2). DMTA cycles are interdisciplinary, multifaceted and include a combination of wet lab chemistry, in vitro and in vivo experimentation. The DMTA cycle iteratively improves on previous design through four stages. Firstly, researchers design new chemical compounds based on data or predictive models. Secondly, chemists attempt to synthesize the newly designed compounds. Thirdly, synthesized candidates are tested in biological assays to evaluate their activity, efficacy, and safety. Finally, these tests are analyzed and further refined in the next DMTA cycle.

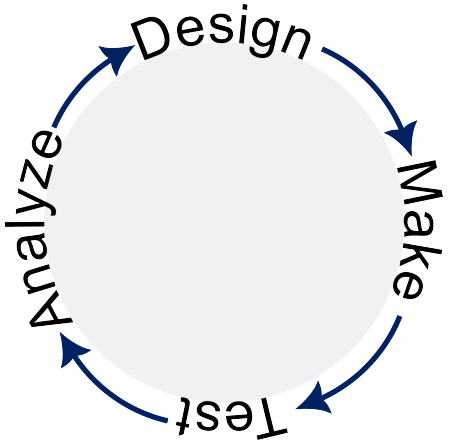


Figure 1.2 Design-Make-Test-Analayze (DMTA) cycle visualized. DMTA cycle is how the drug discovery pipeline iteratively improves candadite drugs.

Pre-clinical Before human clinical trials, pre-clinical testing evaluates the lead compound pharmacodynamics and pharmacokinetics (PKPD), together with toxicity in vitro and in vivo. Compound PKPD describes how the drug distributes throughout the body over time. Animal models are often used to assess the drug's safety profile and biological activity. This stage provides crucial data to support Investigational New Drug (IND) applications, ensuring the compound is safe enough to progress into human trials.

Clinical trials Clinical trials are used once a drug candidate has Clinical trials follow the pre-clinical phase and validate in three phases whether the drug is safe, effective and monitors adverse reactions, respectively. Phase 1 includes a small groups of healthy volunteers test the drug for safety, dosage, and side effects. Phase 2 expands to patients with the target disease to evaluate efficacy and further safety measures. Finally, phase 3 involves larger patient populations to confirm efficacy, monitor adverse reactions, and compare the drug to standard treatments, culminating in data for regulatory submission.

Approval and post-market surveillance Finally, once clinical trials are successfully completed, regulatory authorities such as the United States Food and Drug Administration (FDA) or the European Medicines Agency (EMA) review the data for approval. Upon approval, Phase 4 (post-marketing surveillance) begins to monitor the drug's long-term effects, effectiveness in broader populations, and any rare or unexpected side effects that arise during widespread use.

Problem: Efficacy and efficiency of the drug discovery pipeline One of the major problems facing the pharmaceutical industry is translating early discovery results into clinical settings [15]. Clinical results often highlights discrepancies in efficacy or safety between laboratory/animal models and human subjects, creating significant bottlenecks. Despite the rigorous process, only a fraction of the candidate drugs from the discovery phase successfully navigate clinical trials and achieve regulatory approval.

Additionally, the drug discovery pipeline may take up to five years. As the field moves to other modalities [8, 9] most advancements in drug discovery for small molecule research have moved towards the efficiency of this pipeline. Many promising compounds fail in these stages due to unforeseen toxicities, lack of efficacy in diverse populations, or challenges in manufacturing [16]. Successfully connecting

early-stage discovery to clinical outcomes remains one of the most formidable tasks in drug development.

1.1.2 Notable Advancements in Drug Discovery and Development

Recent technological advancements in drug discovery have significantly accelerated the identification and optimization of small molecules. Among these, the integration of ML has emerged as a transformative force, enabling data-driven insights and automation across multiple stages of the drug discovery pipeline.

A Brief History of Machine Learning in Drug Discovery The earliest applications of ML in this field date back to the 1960s and 1970s, when researchers began using rule-based expert systems to predict biological activity [17, 18]. These systems relied on knowledge bases crafted by medicinal chemists to link molecular structures to pharmacological effects. While limited by extensive manual input and simple heuristics, this marked a shift towards systematic computational methods in drug development.

By the 1980s and 1990s, the growing availability of molecular and biological data, alongside increased computational power, facilitated the rise of ML techniques such as quantitative structure-activity relationship (QSAR) modeling [19–21]. QSAR models used molecular descriptors to establish correlations between chemical structures and biological responses, enabling more data-driven predictions. While early ML methods were constrained by dataset size and feature engineering challenges, they laid the foundation for AI-driven molecular exploration.

The late 2000s and early 2010s saw a significant leap in ML applications for early drug discovery, largely due to the adoption of deep learning (DL). DL models, with their ability to capture complex, nonlinear relationships in molecular data, enabled large-scale virtual screening and predictive modeling [22, 23]. Additionally, innovations in natural language processing (NLP) allowed molecular representations to be processed as text (e.g., SMILES strings), further expanding ML's utility in predicting chemical and biological properties.

Structure-Based Drug Design Structure-based drug design (SBDD) models molecules which interacts with a target using an elucidated protein structure. Elucidating the protein structure is the typical bottleneck in this approach. AlhphaFold, recently celebrated with a Nobel price, has overcome the age-long problem of protein folding, allowing SBDD to design molecules for protein structure solely using the amino-acid sequence [24]. Furthermore, cryo-electron microscopy (cryo-EM) enables the experimental visualization of bio-molecules at near-atomic resolution, capturing multiple conformational states [25]. However, interpreting complex cryo-EM density maps remains challenging. To address this, ML models such as DeepMainmast and DeepTracer-Refine have been developed to automate protein structure determination [26, 27]. These models assist in main-chain tracing and refine AlphaFold-predicted structures using cryo-EM data, improving structural accuracy. These structure-elucidation advancements have further accelerated ML-driven molecular docking and free-energy perturbation methods in hit identification and lead optimization.

Phenotypic Drug Discovery Phenotypic Drug Discovery (PDD) screens compounds based on their observable effects on biological systems without prior knowledge of molecular targets [28]. Unlike Target-Based Drug Discovery (TDD), which focuses on modulating a predefined biological target, PDD enables the identification of first-in-class drugs with novel mechanisms of action [28]. The integration of ML has significantly improved PDD by enhancing data analysis and interpretation. DL models have been trained on single-cell genomics and high-content imaging data to predict cellular responses to diverse chemical perturbations [29]. Additionally, single-cell data have been used to make complete transcriptional cross species maps of cell types [30] allowing ML to aid in the elucidation of disease mode of action [31, 32]. These approaches facilitate the identification of active compounds while providing insights into complex biological mechanisms.

DNA-Encoded Libraries High-throughput screening (HTS) has traditionally required the individual synthesis and testing of thousands of molecules, demanding substantial resources. DNA-Encoded Libraries (DELs) [33] offer an alternative by tagging small molecules with DNA barcodes, allowing simultaneous screening of billions of compounds in a single experiment [34]. The integration of ML in DEL analysis has further streamlined hit identification. Deep learning models trained on DEL selection data can predict active compounds from large libraries, reducing the reliance on costly wet-lab experiments [35]. Additionally, ML methods have been applied to predict the reactivity of building blocks in DEL synthesis, ensuring high-fidelity library generation [36]. DELs in combination with ML promise to significantly increase the efficiency of initial hit screens navigating through a large chemical space.

Organ-on-Chips and Microfluidics Organ-on-chips (OoCs), or Organ-on-a-chip, simulate the physiological environment and functionality of human organs by integrating living cells within microfluidic systems. These platforms offer a more accurate representation of human tissue behavior than traditional cell cultures [37]. The integration of ML enhances OoC applications by automating the analysis of high-content complex imaging from microfluidic platforms, enabling precise assessment of drug effects [38]. Moreover, ML-driven optimization of fluidic system parameters has improved reproducibility and efficiency in drug testing [39]. This synergy accelerates preclinical evaluation while promising to reduce the reliance on animal models.

Automated Chemistry and Synthesis Planning ML-driven synthesis tools are starting to revolutionize automated chemistry and synthesis planning by enabling efficient planning of complex molecules through simpler precursors. Retrosynthesis prediction uses ML to predict precursors for a particular molecule [40–45]. These individual predictions are combined in computer-aided synthesis planning to predict full reaction pathways, significantly reducing synthesis time and improving synthetic accessibility [46–49]. Moreover, autonomous robotic systems have been developed to perform chemical synthesis and reaction optimization. These systems use reinforcement learning to iteratively refine reaction conditions, exploring chemical space efficiently [50–52]. The integration of ML and robotics has led to a more streamlined drug discovery pipeline, enhancing reproducibility and reducing material waste.

Future Directions Beyond these technological advancements, Current failures in translating early drug discovery into clinical success reinforce the need to closer align early steps with clinical efficacy [15, 16]. ML is continuously reshaping drug discovery, with increasing focus on optimizing early-stage processes such as hit identification, lead optimization, and absorption, distribution, metabolism, excretion, and toxicity (ADMET) predictions. ML models now allow for concurrent prediction of multiple properties, integrating synthesis planning, toxicity assessment, and biological activity from the outset [53, 54].

Additionally, foundation models trained on multiple modalities, encompassing molecular structures, high-content imaging, and textual data, are poised to drive the next wave of advancements [55, 56]. These models enable transfer learning across diverse chemical tasks, improving generalizability and reducing data requirements for novel applications.

In the coming years, the convergence of ML with automated laboratories, synthetic biology, and generative chemistry is expected to further enhance the efficiency and creativity of small molecule drug discovery. Large-scale open-source databases like ChEMBL [57] and PubChem [58] have further amplified ML's impact by providing extensive repositories of chemical and biological data. By leveraging these advances, researchers can accelerate the identification of novel therapeutics while reducing costs and improving success rates in early-stage drug development.

1.2 Machine Learning Models for Small Molecule Drug Discovery: Representations, Scaling and Adaptation

ML in drug discovery enables the rapid analysis and prediction of molecular properties and interactions. Central to many ML applications is the *similar property principle*, which hypothesizes that structurally similar molecules tend to exhibit similar biological activities [59]. This principle underpins quantitative structure-activity relationship (QSAR) modeling, where ML algorithms establish correlations between molecular representations and biological activity, facilitating the prediction of new drug candidates [19–21]. By leveraging vast chemical and biological datasets, ML accelerates drug discovery pipelines, optimizes resource allocation, and opens avenues for innovative drug development approaches.

1.2.1 Molecular Representations in Machine Learning

To capture similarity for predictive modeling, molecules can be represented in various forms (Figure 1.3).

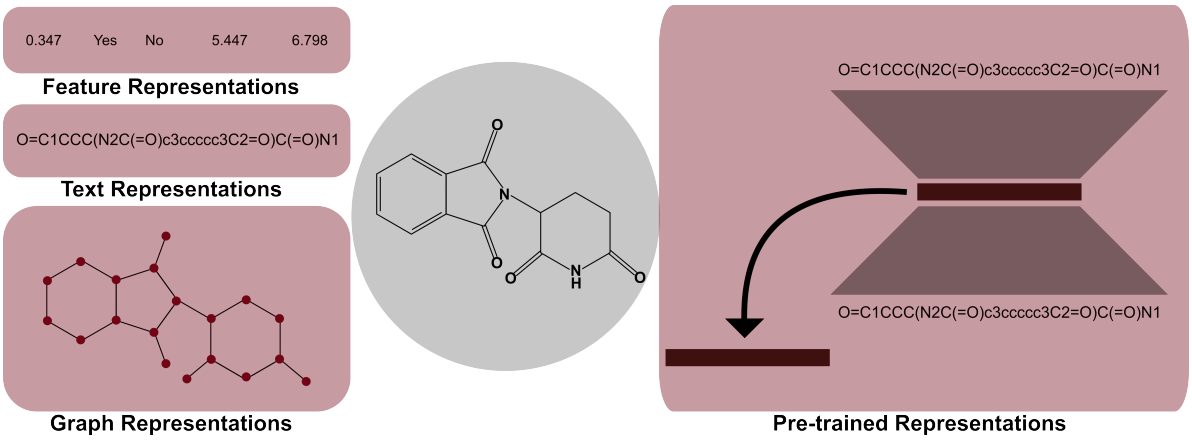


Figure 1.3 An illustrative example of some possible molecular representations. Examples include features, text, graph and pre-trained representations.

Molecular descriptors and fingerprints The simplest forms of representation are molecular descriptors, which are numerical values summarizing specific properties of molecules, such as molecular weight, hydrophilicity, or the number of hydrogen bond donors. These descriptors serve as inputs for traditional ML algorithms, enabling predictions based on quantified molecular features. Building on descriptors, fingerprints represent molecules as fixed-length binary or numerical vectors, encoding the presence or absence of specific substructures or features. Fingerprints, such as Morgan fingerpritns[60], are computationally efficient and widely used in cheminformatics for similarity searches and classification tasks.

Text-based representations As ML methodologies advanced, text-based representations like SMILES (Simplified Molecular Input Line Entry System) [61] and SELFIES (Self-Referencing Embedded Strings) [62] gained popularity. These linear string formats encode molecular structures which are human-understandable and easy to store, making them popular for uses in databases [61]. Their encoding are also naturally compatible with natural language processing (NLP) models, enabling the use of pre-trained architectures to predict molecular properties or design novel compounds. By translating the complex three-dimensional arrangement of atoms into simple text strings, these representations provide an accessible bridge between chemical data and computational frameworks. This adaptability makes text-based representations a cornerstone for modern molecular modeling.

Graph-based representations While text-based approaches emphasize sequential connectivity, graph-based representations capture the relational structure of molecules more comprehensively. In these representations, atoms are modeled as nodes and bonds as edges, allowing graph neural networks (GNNs) to learn from the topology and connectivity of the molecular structure. For more complex tasks, such as protein-ligand binding or reaction prediction, 3D and 4D representations are increasingly important. 3D representations used in ML have explicit spatial information encoded as Cartesian coordinates or distance matrices, these formats describe molecular geometry and provide insights into how molecules interact with their environment [63]. This is particularly important for quantum chemistry and molecular docking studies which often rely on 3D representations to evaluate binding affinities or predict reaction mechanisms. Furthermore, graph-based neural networks, which extend traditional graph representations to include geometric features, have demonstrated success in leveraging 3D data for tasks like property prediction and molecular design. Extending to 4D representations by incorporating time simulations adds an additional dimension. Molecular dynamics (MD) simulations to capture conformational changes and molecular flexibility under physiological conditions by modeling time-dependent trajectories [64]. Together, these representations provide a spectrum of options, balancing expressiveness and computational feasibility, tailored to the specific needs of ML tasks in drug discovery.

1.2.2 Advantages of Pitfalls of Text-based Molecular Representations

More generally, text-based representations play a foundational role in ML, offering a means to transform ubiquitous language data into structured, machine-readable formats that algorithms can effectively process. The field of NLP captures meaning, context, and relationships between words and phrases through embeddings which represent language as dense vector spaces, where the position and proximity of each word reflect semantic relationships based on context and meaning. ML models can subsequently use these embeddings to learn not just individual words but also the complex syntactic and semantic patterns that emerge in language, which are crucial for tasks such as sentiment analysis, translation, and summarization. Beyond language, the ability to convert complex data into structured, machine-readable formats have proved invaluable in drug discovery, where text-based representations enable the analysis of fields including protein design [65] and molecular chemistry [44, 66, 67]. Molecules are complex structures with numerous atomic and structural interactions that are challenging to model directly.

Advantages of text-based molecular representations By using text-based encodings of molecules researchers can represent chemical structures as linear sequences. These representations are especially advantageous for ML models because they allow standard text-processing architectures, such as recurrent neural networks (RNNs) [68, 69], transformers [70–72], and other deep learning models, to work with chemical data [73]. Their compatibility with existing NLP techniques, allows researchers to apply mature, well-optimized language models to molecular data without requiring specialized architectures.

Text-based molecular representations also introduce an efficient way to compress and share chemical information, making it possible to capture and encode structural and stereochemical data with relatively simple syntax [61]. For instance, SMILES strings can compactly represent complex molecules, enabling faster computation and lower memory usage. These representations make it easier to search for and compare chemical structures in vast molecular databases. Paired with advanced text-processing algorithms, text-based representations allow for the efficient retrieval, comparison, and prediction of molecular properties. This efficient representation also offers advantages in generative machine learning models. Generative models often use text-based representations and leverage large datasets to generate diverse chemical structures that adheres to the learned structural rules [74]. This ability ultimately enhances discovery by efficiently covering vast chemical space and optimizing for desirable molecular properties directly within the model framework.

Pitfalls of text-based molecular representations There are notable pitfalls to text-based molecular representations. Converting three-dimensional molecular structures to a one-dimensional string can lose valuable spatial and stereochemical information, which can be critical for predicting molecular interactions accurately. Additionally, not modeling time interactions lacks information necessary for understanding processes like ligand binding, enzyme catalysis, and allosteric regulation.

Furthermore, SMILES and similar formats are inherently sensitive to changes in atom ordering, meaning that small modifications in a molecule's structure can lead to significant changes in the representation, potentially confusing the model [75]. This sensitivity can introduce noise and reduce the generalizability of models trained on these representations. Optimized SMILES representations have been developed to mitigate certain ambiguities and improve model performance, such as canonical SMILES [42, 76–78] for consistent representation of molecules and SELFIES [62], a SMILES-like language that ensures valid chemical structures by design. Researchers are also actively exploring enhancements to text-based encodings, such as self-supervised learning and contrastive learning, to better capture molecular context [79, 80]. However, current implementations still suffer from this inherent dissimilarity, leading to a possible violation of the similar property principle.

As ML models in drug discovery continue to evolve, text-based representations remain a cornerstone of molecular modeling, providing a bridge between complex molecular structures and computational frameworks. By translating intricate molecular information into text-based formats, these representations facilitate the use of sophisticated ML techniques and bring fast powerful NLP and representation learning tools into the realm of chemistry and drug discovery. However, despite their computational intensity, the integration of 3D and short-term 4D data into machine learning models is becoming increasingly feasible with advancements in hardware and algorithms [63, 64]. These alternative representations highlight the growing need for multidimensional and possibly multi-modal approaches in cheminformatics and drug discovery, enabling deeper insights into molecular properties and interactions.

1.2.3 Pre-trained representations

Pre-trained ML models are models used as a general foundation to learn more specific tasks faster and more generalizable [81]. These ML models are large-scale machine learning models designed to perform well on a wide range of tasks across various domains. Scaling laws in ML describe how model performance improves as parameters, data, and compute scale up [82]. These empirical observations show that larger models trained on massive datasets often generalize better, making them particularly suited for foundational tasks. Trained on massive datasets with a large number of parameters, these models learn generalizable patterns and representations [82]. Pre-trained representations capitalize on these scaling laws to capture broad patterns across diverse domains [83]. Their recent popularity comes as they can be quickly adapted to new tasks with minimal additional training, offering both versatility and efficiency [84].

Text-based representations serve as a particularly influential subset. These models are pre-trained on large training sets of unlabeled text, which are available in abundance. These large corpora of training data enables them to capture nuanced linguistic patterns, including syntax, semantics, and even pragmatics [85]. The capabilities of text-based large ML models have led to their adaptation across disciplines outside of traditional NLP, as unlabeled text often has higher information density and causality than other modalities like images [86]. In fields like biology and chemistry, pre-trained ML models trained on natural language have inspired the use of similar architectures for sequence-based data. In drug discovery, text-like representations of molecules, enables the direct application of NLP models for tasks like property prediction, molecule generation, and synthesis planning. By training on vast libraries of unlabeled molecular data, these models can identify patterns in chemical properties and activities that would be difficult to extract manually. Leveraging pre-trained representations in this way opens up new opportunities for cross-disciplinary insights and optimizes performance on tasks where annotated data is not, or scarcely available.

1.2.4 Training and Adaptation of ML Models

Pre-training text-based models are often unsupervised or self-supervised, where the model learns from raw, unstructured text data without labeled samples [87]. Through techniques like masked language modeling or autoregressive training, where the model tries to construct the original input from noisy or incomplete sentences, they develop contextual embedding representations of words and sentences that retain meaning and contextual information, making them highly effective for downstream tasks. The pre-training process not only allows the models to learn relationships within sentences but also across a broader context, enhancing their understanding and enabling more general and coherent performance.

Fine-tuning an ML model is when the model undergoes additional training on a smaller, task-specific dataset. Fine-tuning adjusts the model's parameters, allowing it to refine its understanding and focus on particular characteristics relevant to the target task. For instance, in biomedical or chemical applications, a pre-trained model initially trained on general molecular structures to learn general molecular properties can be fine-tuned on task-specific data, such as toxicity or synthesis. This approach builds on the pre-existing knowledge of the model, reducing the data and computational resources needed to achieve high accuracy in more niche fields.

Transfer learning is another method for adapting pre-trained models, utilizing internal representations as feature encodings for downstream tasks. ML models generate embeddings—dense vector representations that capture relationships within data. These embeddings, produced from various layers within the model, can serve as universal encodings or be tailored for specific use cases by adjusting the layer from which they are extracted. For example, embeddings from text-based ML models can be used as input features in separate machine learning models, enabling researchers to leverage the rich, contextual knowledge of the ML model without additional fine-tuning. This approach is particularly useful in scenarios where computational resources are limited, or for multi-task learning, where a single model is trained to perform several related tasks simultaneously.

Context-driven adaption is where ML models can be adapted without further training by changing the context given to the input. Techniques including CLIP and Retrieval-Augmented generation (RAG) have been used to apply different contexts to model inputs to achieve high zero-shot accuracies, where the model has never seen anything similar to what it predicts. In these models, context is added or changed from the input formats the model has seen before, and translated to the new task at hand. This flexibility is particularly valuable in scientific domains, where access to labeled datasets is often limited. The ability to apply a model pre-trained on broad data to smaller, domain-specific tasks reduces the barriers to entry for high-quality, domain-aware models and allows researchers to build on top of established, powerful architectures.

1.2.5 Pitfalls when using pre-trained Models

While pre-training ML models have demonstrated remarkable versatility and power, they come with several significant pitfalls that can limit their effectiveness in specific applications. One primary concern is overfitting, particularly when models are fine-tuned on smaller, domain-specific datasets. large ML models are often highly parameterized and, when fine-tuned on limited data, can become overly specialized capturing noise rather than meaningful patterns. This overfitting undermines the ability to generalize and perform accurately on new, unseen data. Additionally, fine-tuning on narrow datasets can lead the model to forget more general knowledge learned during pre-training, also known as *catastrophic forgetting*. This trade-off between specialization and generalization presents a substantial challenge, especially in scientific and technical fields where high accuracy and generalizability are equally critical.

Overparametrization is another pitfall, as large ML models are typically built with billions of parameters [82]. While this scale allows the model to capture complex patterns, it also leads to inefficiencies and the need for substantial computational resources. Large models demand more storage, memory, and processing power, making them inaccessible for many researchers and organizations, particularly those with limited computational budgets. Moreover, overgeneralization can contribute to diminishing returns in performance. As models increase in size, the improvements in accuracy and robustness become marginal in relation to the added computational costs. This phenomenon has led to debates over scaling laws in ML, where simply increasing model size does not always translate to proportionate gains in practical utility, especially for specific tasks with limited data [83].

Scaling laws and model size pose additional challenges in terms of interpretability and alignment with the intended applications. As ML models become larger, they become harder to interpret and control, often with unclear reasoning behind their predictions. This opacity can be problematic in domains like healthcare or scientific research, where model decisions must be both accurate and explainable in order to be used. Additionally, larger models can sometimes exhibit unpredictable or biased behavior, particularly when they are pre-trained on vast, uncurated datasets. This lack of data transparency complicates efforts to understand and mitigate biases, creating ethical and practical concerns around the deployment of ML in sensitive or regulated fields.

1.3 Explainability in AI: Concepts, Challenges, and Methods

Explainability, often referred to as Explainable AI (XAI), has become a central focus in the field of machine learning, emphasizing the need for models that are not only accurate but also interpretable [88]. Opacity has become a big problem as ML models grow in complexity, particularly with the rise of deep learning and foundation models. This opacity can lead to mistrust, especially in fields such as healthcare, finance, and law, where understanding the reasoning behind predictions is crucial for ethical and reliable decision-making. Regulatory frameworks, such as the European Union's General Data Protection Regulation (GDPR), are increasingly emphasizing *the right to explanation*, particularly for automated decision-making systems [89, 90]. This regulatory pressure underscores the need for models that are not only high-performing but also transparent. Explainability in ML seeks to bridge this gap by providing insights into how models arrive at their predictions, highlighting the features, patterns, or data structures that influence outcomes.

1.3.1 Core concepts and taxonomy of XAI

XAI encompasses several core concepts that guide the development of interpretable and transparent machine learning models. Here, specific attention is given to scope and application taxonomies in XAI.

Local versus global explanations is a key concept in XAI around the scope of an explanation [91]. Global explanations provide an overview of how the model makes decisions across the entire dataset or as a whole, offering insights into its overall structure and patterns. These explanations are useful for understanding the model's general behavior and checking for systemic biases. In contrast, local explanations focus on individual predictions, aiming to elucidate the decision-making process for a single data point. Local explanations are particularly useful for high-stakes applications where understanding specific predictions, rather than general model behavior, is essential. Together, these concepts form the backbone of XAI, guiding researchers and practitioners in creating models that are not only powerful but also transparent, interpretable, and aligned with ethical standards.

Post-hoc explanations are the current standard in XAI, which are explanations of predictions after the fact [91]. Most current XAI techniques fall into this category. Techniques in this field often focus on quantifying the contribution of each input feature to a model's prediction. Feature importance methods are used to identify which features the model relies on most heavily, making it possible to understand the underlying drivers of model behavior. Feature importance not only aids in interpretability but also provides insights into the relationships within the data, helping users assess whether the model aligns with domain knowledge and expected patterns.

Ante-hoc/intrinsic model interpretability refers to the degree to which a model's internal processes and outputs can be understood by humans [92]. Interpretability varies widely depending on model complexity, with simpler models like linear models, decision trees, or rule-based classifiers being inherently more interpretable than deep neural networks or ensemble methods. Although such models may not capture complex patterns as effectively as deep learning models, they can still provide robust performance in many applications. Researchers are also developing new architectures, such as concept embedding models (CEM) [93, 94], that balance model complexity with built-in interpretability. This shift is particularly relevant in high-stakes applications, where transparency is prioritized over marginal gains in accuracy.

1.3.2 Common XAI Techniques

Generally, XAI techniques can be classified into model-agnostic and model specific approaches. Model-agnostic can be applied to any model architecture or ML technique and often involve the use of sur-

rogate models, simpler models that explain the prediction of a more complex model or explain the behavior by contrasting similar inputs variables. Model-specific approaches range from using inherent interpretable models to utilizing specific model attributes as approximate explanations. Furthermore, visual explanations are used to better allow end-users to use their own intuition to validate specific predictions. Finally, work on dissecting models has lead induced sparsity to be used for XAI, explaining relationships between learned features and predicted outcomes.

Model-agnostic approaches

Surrogate model techniques like SHAP (SHapley Additive exPlanations) [95] and LIME (Local Interpretable Model-agnostic Explanations) [96] approximate model predictions by locally perturbing inputs and observing changes in output. These tools allow users to explore which features are most influential in the model's decision for a specific prediction, providing a semblance of transparency.

Recently, counterfactual analysis has gained traction, which helps researchers evaluate what changes to a molecule would lead to a different prediction [97, 98]. For instance, if a model predicts high toxicity for a compound, counterfactual analysis contrasts two compounds with differing functional groups or substitutions to see at which point prediction deteriorates or improves. This technique is especially useful in lead optimization, where scientists seek to improve specific properties without altering the molecule's overall function.

Model-specific approaches

Model specific approaches can range from the analyzing models/model components regarded as inherently interpretable, such as decision trees (DTs) [99], to using Integrated Gradients (IG) [100] to analyze the flow of gradients through a model. In models based on transformer architectures, attention mechanisms allow the model to focus on specific parts of the input sequence (such as atoms in a SMILES string or nodes in a molecular graph) that are most relevant to the prediction [70, 101]. Attention maps show which atoms or molecular substructures the model focuses when making decisions, providing insights into the model's internal reasoning. This capability is particularly useful in drug discovery because it reveals how the model interprets molecular representations, helping chemists understand which structural features may be linked to a compound's pharmacological properties or off-target effects.

Visual explanations

Additionally, visualization techniques, like saliency maps [102] and Class-activation Maps [103, 104] for image models or attention heatmaps for NLP models [70, 101], allow researchers to pinpoint the contributions of individual inputs or variables to the model's output. This is especially important in drug discovery, where understanding which parts of a molecule contribute to activity, toxicity, or binding affinity can guide chemical modifications to optimize a candidate molecule's properties.

Dissecting Neural Networks

Recently, post-hoc global interpretability by dissecting neural networks has been gathering interest [105]. The general idea is that overall model features are consistent, but not necessarily related to individual neurons. Explaining global model behavior is performed through a sparse auto-encoders (SAE) which learn a sparse and interpretable concept map from high-dimensional intermediate layers [106]. SAEs learn a concept map by compressing the intermediate layers of a trained ML model to a sparse mapping, often binary, and then decompresses the output. This can then be further analyzed by altering the mapping to investigate the effects on the output [107]. In XAI, SAE can be used to visualize latent features or identify critical input dimensions that influence model decisions. For example, in large language models, SAEs have been used to identify and unlearn certain facts [108].

1.3.3 Challenges in explainability

Explainability in ML faces several challenges. One is the trade-off between accuracy and interpretability. Simpler models tend to be more interpretable but often under perform compared to more complex models [109, 110]. For example, deep neural networks, which can capture nuanced patterns in data, are typically less interpretable than linear models. Another challenge is ensuring that explanations are both faithful and intelligible [93, 111]. Faithful explanations accurately reflect the model's internal reasoning, while intelligible explanations are accessible and meaningful to the user. Ensuring both can be difficult, as explanations that simplify the model's behavior for clarity may unintentionally distort the underlying decision-making process. Finally, current XAI methods often neglect to include the influence of prediction uncertainty, a critical aspect of model reliability [112]. Uncertainty quantification is a complete field of ML, but XAI methods often ignore uncertainty in predictions and contrasting important features with other possible predictions. This is especially pronounced in most current post-hoc XAI methods.

1.4 Objectives of the Thesis

Text-based molecular representations are often used in ML for drug discovery to great success. However, most ML models for drug discovery rely on the similar property principle which hypothesizes that structurally similar molecules exhibit similar effects. As SMILES-based molecular representations are inherently sensitive to changes in atom ordering, small modifications in molecular structure can lead to significant changes in input. Current state-of-the-art methods often use ML models pre-trained to learn to generate SMILES from randomized SMILES and then fine-tuned the latent representations for down-stream tasks [41, 42]. As such, the use text-based molecular representations for ML models are presumed to be at least partially protected from the invariance of SMILES. However, the field has yet unexplored how different pre-training strategies affect trained representations and how these effect XAI and downstream predictions. Additionally, the best ML models for text-based molecular representations use Transformer architectures which are usually significantly larger and often prohibitively computationally expensive than their traditional counterparts. In this cumulative thesis, I investigate the reliability and usability of text-based molecular ML in drug discovery, mainly through two first-author papers.

Firstly, text-based models are large, highly parameterized networks. As complexity is often inversely related with interpretability and general applicability of XAI methods, a crucial research question is whether current out-of-the-box XAI methods are reliable in text-based molecular models. To investigate the effectiveness of text-based representations in ML models with XAI, we explore current XAI methods and apply them to the field of toxicity prediction. Our objectives here are to compare explanations from the text-based models trained to predict toxicity with previously established intuitions of toxicity. Additionally, we want to know the influence of some predominant pre-training techniques on the stability of XAI explanations. Finally, we want a method that explanations are not just intelligible, but also faithful to internal model reasoning. Through these methods, I attempt to answer the following research questions:

1. Can these models provide reliable predictions that are interpretable and actionable by drug discovery scientists?
2. To what extent can text-based molecular representations, like SMILES, be adapted for robust predictive modeling?
3. Are SMILES representations robust when used for interpretation?

Secondly, as text-based representation ML models need to be adapted for specific tasks, both their adaptation as well as their effectiveness as a tool in a chemists' toolbox needs to be investigated. We therefore question whether large ML models can be effectively used in larger systems and whether we can increase this efficiency. I explore using text-based ML models in larger systems to synthesize drug candidates. Here, I aim to find the environmental impact in terms of carbon footprint of ML models. Additionally, my objective was to find the applicability of these slower ML models as part of a large system. Finally, I aim to investigate whether we can increase the efficiency of text-based models and what downstream effects that has on the larger system on a whole. Within this framework, I attempt to answer the following research questions:

1. Are text-based models better than use of smaller models when used within larger systems?
2. What are the environmental impact of choosing models over smaller neural networks?
3. Are there ways in which we can increase the efficiency of models for their use in larger systems?

1.4.1 Contributions

The articles presented in this cumulative thesis have contributions both in empirical findings and methodological advancements. My critical analysis of current explainability techniques provides insight into their limitations when applied to molecular modeling, highlighting the need for better-aligned model-human interpretability tools. I investigate in section 3.1 if XAI aligns with what experts in the field of toxicity have deemed important to challenge the belief that XAI methods can be validated by thinking as experts would [1]. Here, I draw the crucial conclusion that faithfulness can not be evaluated through alignment with human intuition. Additionally, I expand current methodology for analyzing XAI methods

through test-time augmentation, where I compare the same ground-truth showing inconsistencies within models for identical representations. Furthermore, I provide a critical analysis of the use of larger ML models for synthesis planning with Chemformer [44] in section 3.2, demonstrating the current feasibility of high-accuracy, low-latency models tailored for practical drug discovery use cases [2]. I analyze ML in the larger route-finding environment and attempt to optimize the model architecture. I show that in isolation, techniques like knowledge distillation, advanced architectures or even standard hyper-parameter optimization can increase efficiency of pre-trained models. However, the use of larger ML in the field of computer-aided synthesis planning requires a complete change to become computationally viable for high-throughput use. Finally, I provide insights into the methods in which models are used, further the field towards an overall more efficient use of models in larger more complex systems.

Overall, the contributions of this cumulative thesis aim to explore and develop strategies that enhance the interpretability and reliability of text-based molecular models, thus making them more applicable to drug discovery tasks.

2 Methods

> The method of science, as stodgy and grumpy as it may seem, is far more important than the findings of science.
>
> Carl Sagan, *The Demon-Haunted World*

2.1 Datasets

2.1.1 Pre-training Datasets

For section 3.1 and section 3.2, models were first pre-trained on a large database of molecular structures using either the full small molecule data from ChEMBL 29 [57, 113], and approximately 100 million randomly selected SMILES strings from the around 1.5 billion molecules in the ZINC 15 [114] dataset, respectively.

ChEMBL is a manually curated database of bioactive molecules with drug-like properties. It includes chemical structures, bioactivity data, and associated metadata, such as target information and assay conditions. The dataset provides a robust resource for pre-training models, which focus on learning small molecule structural information from a broad range of drug-like chemical structures.

ZINC, similarly, database contains millions of commercially available small molecules designed for virtual screening. It includes both lead-like and drug-like compounds, along with molecular descriptors such as molecular weight and LogP. This dataset supports training by exposing models to a large and diverse chemical space.

2.1.2 Retrosynthesis

For section 3.2, retrosynthetic models were trained and evaluated using datasets derived from the United States Patent and Trademark Office (USPTO). The USPTO 50K [5] dataset includes approximately 50,000 reactions categorized into the ten most common reaction classes, including: Acylation and related processes; Heteroatom Alkylation and Arylation; C-C Bond Formation; Heterocycle Formation; Protections; Deprotections; Reductions; Oxidations; Functional Group Interconversion (FGI); and Functional Group Addition (FGA). Its balanced class distribution facilitates benchmarking of single-step retrosynthesis methods.

In contrast, the USPTO Full [5] dataset provides a more extensive and diverse reaction collection, encompassing both common and rare reaction types. Its inclusion of imbalanced classes makes it better for training models to generalize retrosynthetic predictions across a broader chemical space.

2.1.3 Toxicity Prediction

For section 3.1, a toxicity prediction dataset was used to evaluate models on their ability to identify harmful chemical properties. Specifically, Therapeutics Data Commons (TDC) [115] was employed to gather the toxicity dataset with Ames mutagenicity data. Furthermore, expert-derived structural alerts for Ames toxicity were gathered from Kazius et al. [116].

2.1.4 Data Preparation

Datasets were processed to remove stereochemistry and salts; to correct invalid hybridization, conjugation, chirality, and valency; and to set correct chirality, aromaticity and chemical property flags. Corresponding canonical SMILES were then generated and duplicates were removed. SMILES were tokenized based on the character representations of the string format (section 3.1) or using a vocabulary generated from ChEMBL27 [113] (section 3.2). Tokenizations were then given both a beginning-of-sequence (BOS) and end-of-sequence (EOS) token together with optional padding (PAD) tokens to make sequences of equal length, according to the original paper from Karpov et al. [78] and Irwin et al. [44]. Data augmentation methods, including SMILES enumeration and randomization, were applied to enhance the diversity of the training data and improve model generalization.

2.2 SMILES (Simplified Molecular Input Line Entry System)

SMILES provides a method to represent a molecule into a linear string of characters, facilitating its use in computational chemistry and cheminformatics applications [61]. Introduced by Weininger [61], SMILES was developed to overcome the challenges posed by traditional chemical nomenclature systems, such as the IUPAC nomenclature [117], which are difficult for use in database queries and substructure searches. Unlike these complex naming systems, SMILES maintains human readability while being compatible with chemical software.

One of the key reasons SMILES gained popularity is its simplicity compared to other early chemical representations, such as the Wiswesser Line Notation (WLN), which employed a more complex rule set [118]. In contrast, SMILES uses a graph-theoretical approach to chemical structures, which reduces ambiguity and is easier to encode and interpret. Moreover, SMILES provides a more human-readable and chemically understandable than alternative representations during its inceptions such as the Chemical Abstracts Service (CAS) system [119]. Additionally, the introduction of unique SMILES notation ensured that each molecule could be assigned a canonical string, allowing for consistent representation across different databases and applications [76]. This balance of simplicity, readability, and computational efficiency has contributed to the continued widespread use of SMILES in cheminformatics today, ensuring its relevance in fields ranging from drug discovery to materials science.

2.2.1 Structure and Syntax of SMILES

SMILES operates by converting molecular structures into a linear string of characters based on graph theory. Each molecule is first represented as a graph, where atoms are nodes and bonds are edges. The SMILES algorithm performs a depth-first traversal on this graph, sequentially encoding atoms, bonds, and ring closures into characters. SMILES also allows for different variations. Firstly, canonical SMILES is a SMILES format which is a unique SMILES string for each molecule, enabling consistency across software platforms. Secondly, isomeric SMILES includes stereochemical and isotopic information for more detailed molecular descriptions.

Atom Representation

SMILES notation consists of a series of characters containing no spaces (ASCII codes of corresponding symbols). Atoms, nodes in the graph search, are represented by the standard abbreviation of chemical elements inside square brackets (e.g., [Pd] for palladium). Brackets are used to add additional information as well, including to formalize charges (e.g., [NH4+] or [NH++++] for ammonium), denote alternative or explicit hydrogens (e.g., [CH4] for methane), radicals (e.g., [CH3]) or isotopes (e.g., [14C] for carbon-14) in isomeric smiles.

Square brackets may be omitted if an atom adheres to the following criteria. Firstly, atoms must be in the so-called organic subset of B, C, N, O, P, S, F, Cl, Br, or I. Secondly, atoms must have no formal charge. Thirdly, molecules must have the number of attached hydrogens implied by the SMILES valence model. Fourthly, atoms must denote the standard isotope. And, finally, atoms must not be chiral centers. In these cases, brackets may be omitted, e.g., CC in the case of ethane. Hydrogen atoms may be omitted as long as they satisfy the above requirements and are not connected to other hydrogens.

Furthermore, SMILES encode aromaticity by using lowercase letters for aromatic atoms, such as c for carbon and n for nitrogen. An atom is considered aromatic if it meets the criteria based on Hückel's rule (4n+2 π electrons in a cyclic, planar conjugated system). For example, benzene is represented as c1ccccc1, where the lowercase c indicates an aromatic carbon.

Bond Representation

Bonds are represented by symbols (e.g., '-' for single, '=' for double), and parentheses indicate branching points (Table 2.1). Ring closures are marked by numbers to represent cyclic structures. Furthermore, SMILES encodes chirality using the '@' symbol to denote tetrahedral stereocenters. For example, the molecule (R)-lactic acid is represented as CC(O)C(=O)O without chirality and C[C@@H](O)C(=O)O with chirality to identify the chiral center. The symbols @ and @@ are used to differentiate between clockwise (R) and counterclockwise (S) configurations. Finally, E/Z stereochemistry of double bonds can be specified using / and \to indicate cis and trans configurations, respectively. For example, (E)-butene is represented as C/C=C/C, while (Z)-butene is C/C=C\C. Stereochemistry is often optional but preffered in chemical notation.

Table 2.1 Summary of Key SMILES Atom and Bond Representations.

Entity	SMILES Representation
Atoms	Periodic element (E.g., H, He, Li, ...)
Orcanic subset	B, C, N, O, P, S, F, Cl, Br, I
Charged atom	[N+], [O-]
Explicit hydrogens	[CH3], [CH4]
Isotopes	[14C], [2H]
Atom Mapping	[C:1]
Single bond	- (omitted in most cases)
Double bond	=
Triple bond	#
Aromatic bond	:
Chiral centers	@, @@
Stereochemistry	/, \
Ring closure	Numbers (E.g., 1,2,...)
Branching paths	()

2.2.2 SMILES procedure

SMILES are generated from a graph-based depth-first tree traversal [61]. The graph structure is converted into a spanning tree by breaking cycles. Broken bonds are annotated with numerical labels that allow for the reconstruction of the cyclic structure. The molecule is then traversed in a depth-first manner, and the traversal is encoded into the SMILES string.

Generic SMILES are generated from a graph traversal where there is no specific algorithm in place to define branch order, which bonds to break and what additional information to encode. Typically, hydrogen atoms are omitted from the graph, and only remaining structure is converted. Graph nodes are converted into a count vector of atomic invariants with information about 1. the number of connections, 2. the number of non-hydrogen bonds, 3. the atomic number, 4. the sign of the charge, 5. the absolute charge, 6. the number of attached hydrogens. This set of six variables is used to describe a unique notation for simple SMILES strings.

Isomeric SMILES are written with isotopic and chiral specifications. Isomeric SMILES typically add the information on 7. isotopic mass, 8. bond directionality, and 9. local chirality to the invariants.

Unique or canonical SMILES are written in a manner that is unique for each chemical structure. Any hydrogen atoms are removed from the graph and the graph is then traversed from a beginning node in a depth-first manner. The difference between unique SMILES and generic SMILES is that unique

SMILES use a specific method of determining the beginning node and branching path at branching nodes.

Absolute SMILES are SMILES which are both unique and isomeric.

Tautomerization and Bonding Conventions

SMILES does not inherently distinguish between tautomeric forms of a molecule, as it provides a single representation for each specific structure. However, alternative bonding conventions can be used to represent different resonance or tautomeric states. For example, nitrobenzene can be represented as O=[N+]([O-])c1ccccc1 or as a resonance structure O=n1ccccc1[N+]. Canonical SMILES resolve tautomer ambiguities by choosing a standardized form. For example, keto-enol tautomerism is resolved to favor the keto form. This can ensure consistent representation for downstream applications, even if input molecules are provided in different tautomeric states.

SMILES extensions: SMARTS and SMIRKS

SMARTS (SMILES Arbitrary Target Specification) extends SMILES to enable substructure searching by allowing wildcard atoms (*) and logical expressions. For example, [#6] matches any carbon atom, while [#6H3] matches carbon with three hydrogens.

SMIRKS (SMILES Reaction Kinetics Specification) provides a way to represent chemical reactions, defining reactants, products, and transformation rules. For instance, a generic substitution reaction can be written as [C:1][H]»[C:1][O], where [C:1] represents a mapped carbon atom undergoing substitution. These extensions are widely used in reaction databases and cheminformatics toolkits to facilitate reaction modeling and virtual screening.

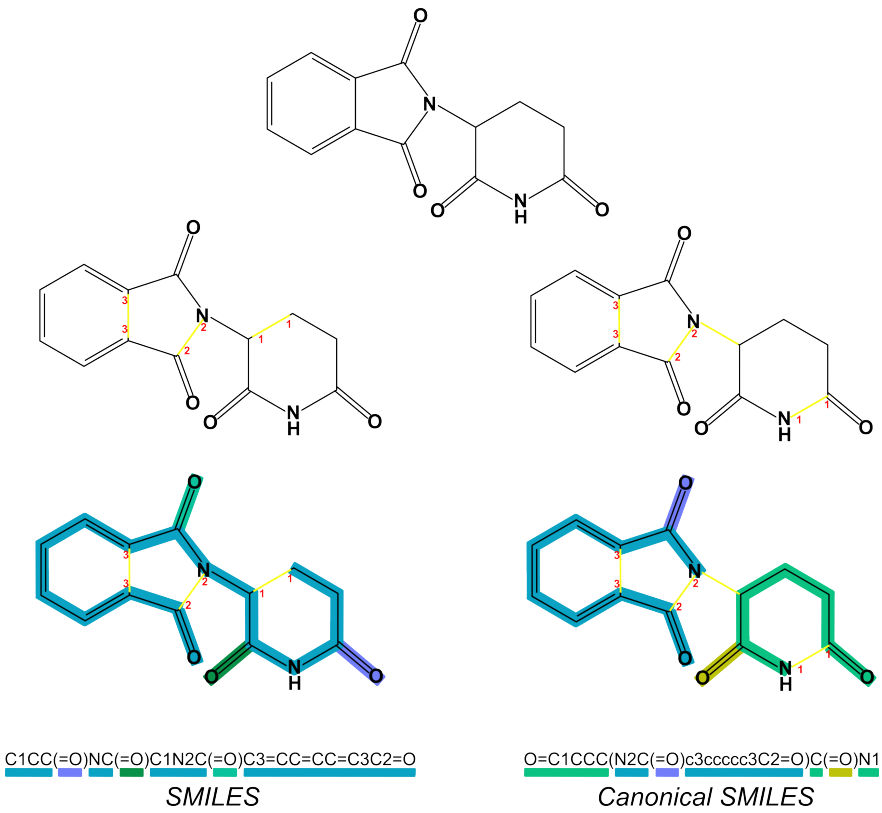


Figure 2.1 Graph-to-SMILES encoding procedure. Example shows Thalidomide traversals for a random SMILES (left) and a canonical SMILES (right) through the steps of identifying and breaking rings, and traversing the graph.

2.2.3 SMILES augmentation for model training

Utilizing multiple SMILES as a data augmentation is a data enhancement technique used to improve the performance of machine learning models by increasing the diversity of training data. Since SMILES representations can be non-canonical, multiple valid SMILES strings can describe the same molecule. By randomly sampling different SMILES during training, models learn to generalize over a wider range of molecular structures, reducing overfitting and improving their robustness to variations in input representation. Augmentation involves systematically or stochastically generating such variations. Recent studies have shown that SMILES augmentation significantly enhances the performance of sequence-based models, such as Transformers, by introducing variability into the training process [42, 77].

2.3 Transformer Architecture

Transformer-based architectures have emerged as the state-of-the-art framework for sequence modeling and representation learning, and serve as the backbone of this thesis. The Transformer architecture excels due to the attention mechanism, which dynamically adjusts the focus on different parts of the input sequence, enabling the model to capture contextual relationships without relying on recurrence or convolution [70]. Transformers offer unparalleled performance using parallelization to dominate in tasks including classification and generative modeling, with their seminal attention mechanism aiding interpretability.

2.3.1 Core Components of the Transformer

Self-attention allows the Transformer architecture to model relationships within and across sequences without relying on recurrence or convolution. Self-attention dynamically weighs the relevance of each token in a sequence to every other token. Mathematically, self-attention captures these dependencies by comparing query, key, and value representations of tokens, scaling their dot-product interaction, and normalizing the result. For a sequence with d-dimensional embeddings, the scaled dot-product attention is defined as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V, \tag{2.1}$$

where Q, K, and V represent query, key, and value matrices, respectively.

Multi-head Attention (MHA) is an extention to diversify the focus of the model. Each attention head learns distinct relationships in parallel, and their outputs are merged and refined through linear projections. This multiplicity enables richer representations, enhancing the architecture's ability to capture subtle patterns.

$$\text{MHA}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O, \tag{2.2}$$

where each head computes attention independently and W^O projects concatenated outputs to the target dimension.

Multi-layer perceptrons (MLPs) enriches the relationships learned from attention using a more localized computations. These fully connected networks transform the outputs with non-linear activations, further abstracting and refining the learned features. To ensure smooth gradient flow and combat challenges in training deep models, skip connections tie the inputs and outputs of each layer together. These residual pathways stabilize learning and encourage the network to extract deep, hierarchical representation

Encoder-decoder structure enables the Transformer architecture to be more modular, comprising an encoder to process input sequences and a decoder to generate outputs. The encoder stacks self-attention and feed-forward layers to distill the input into dense embeddings, while the decoder employs cross-attention to link these embeddings with its generative process. This design enables the Transformer to handle a wide variety of tasks, from classification to complex sequence generation. Each component operates harmoniously, contributing to the architecture's exceptional adaptability and efficiency.

2.3.2 Architecture adaptations

Central to section 3.1 are two specific Transformer variants: the encoder-only Bidirectional Encoder Representations from Transformers (BERT) [120], which consists solely of an encoder stack. It is pretrained using two tasks, and Bidirectional and Auto-Regressive Transformers (BART) [72], which uses

the encoder-decoder architecture central to sequence-to-sequence modeling [71] (Figure 2.2). While both leverage the core Transformer structure, their pretraining objectives and use cases diverge. BERT, with its encoder-only design, learns bidirectional representations by predicting masked tokens within the input sequence. This approach emphasizes understanding the input holistically, making it suitable for tasks like classification and regression. In contrast, BART employs a full encoder-decoder architecture designed to reconstruct corrupted text, excelling in sequence-to-sequence tasks.

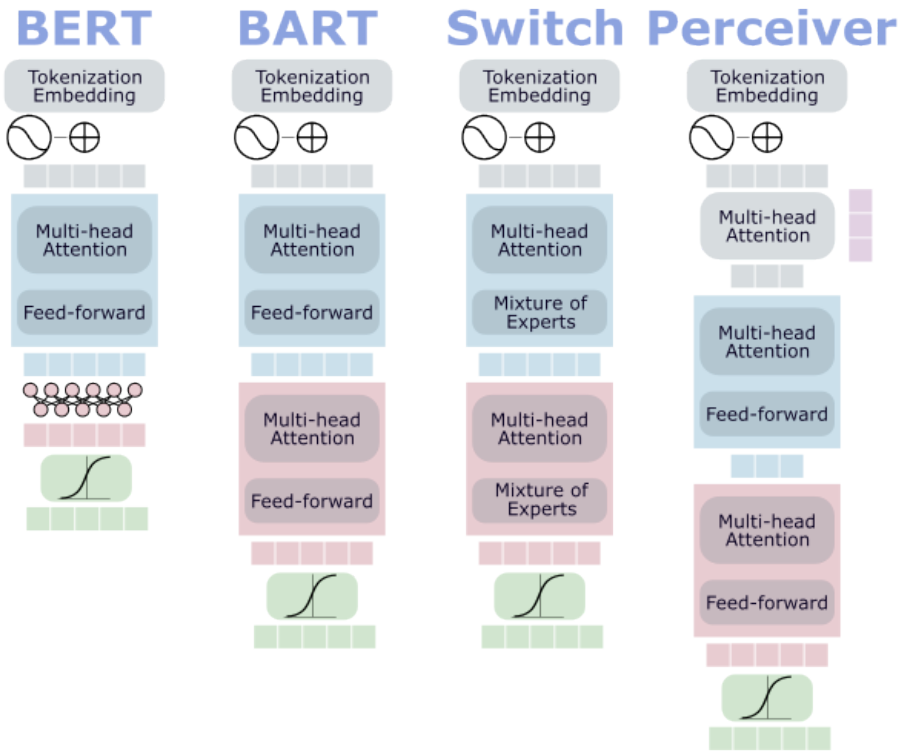


Figure 2.2 Overview of Transformer architectures used in this thesis, including BERT, BART, Perceiver, and Switch Transformer variations.

After pre-training, fine-tuning focuses these models to specific tasks. The simplest way is to separate the encoder and add an MLP head on top for classification or regression. The encoded representations of the encoder pass through the MLP, producing predictions based on the learned features.

2.3.3 Architecture variations

Two architectural variations were explored in section 3.2 to enhance computational efficiency (Figure 2.2). Firstly, the Perceiver model which addresses dimensionality challenges using a cross-attention layer [121]. The cross-attention was introduced at the start of the encoder, where a learnable small projection tensor served as the query to project the large incoming tensor onto a smaller tensor. The embedded source sequences functioned as key-value pairs, enabling efficient dimensionality reduction while maintaining the richness of contextual information.

Furthermore, the switch Transformer replaces the traditional feed-forward networks of the Transformer with a sparse mixture-of-experts approach, where specialized smaller expert layers processes tokens routed by a dynamic gating mechanism [122]. This design not only reduced computational overhead but also retained model performance, ensuring a balance between scalability and accuracy. These adaptations underscored the flexibility of Transformer-based architectures in meeting diverse computational and modeling needs.

2.4 Metrics

2.4.1 Classification Metrics

Classification metrics provide a comprehensive lens through which the performance of binary classification models is evaluated. At their core, these metrics serve to quantify the balance between correctly identifying positive instances and avoiding incorrect predictions.

The first and most intuitive metric, **accuracy**, is the fraction of all correct predictions over the total number of instances. While straightforward, accuracy can obscure nuances in imbalanced datasets, where minority classes are often of greater interest. To address this, the **Matthews Correlation Coefficient (MCC)** is employed as a balanced measure, leveraging all four components of the confusion matrix:

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}. \tag{2.3}$$

Here, TP and TN represent the true positives and negatives, while FP and FN denote false positives and negatives. By incorporating all these elements, MCC effectively captures both precision and recall in a single metric.

Precision, also known as **specificity**, hones in on the accuracy of positive predictions:

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{2.4}$$

In contrast, **recall** (or selectivity) focuses on the model's ability to identify all actual positives:

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{2.5}$$

For a holistic view, the **Area Under the Receiver Operating Characteristic Curve (AUROC)** combines recall and specificity, illustrating the trade-off between sensitivity and false positive rates.

These metrics, when applied together, weave a detailed narrative of the model's classification capabilities, shedding light on both its strengths and weaknesses.

2.4.2 Generative Metrics

Generative models introduce a unique challenge: evaluating outputs that may not have a singular ground truth. In molecule generation tasks, metrics are tailored to assess syntactic validity, chemical plausibility, and diversity.

A fundamental metric is **token accuracy**, which measures the percentage of correctly reconstructed non-padding tokens in generated SMILES strings. On a broader scale, **sequence accuracy** gauges how often entire SMILES strings are perfectly reconstructed:

$$\text{Sequence Accuracy} = \frac{\text{Correctly Reconstructed Sequences}}{\text{Total Generated Sequences}}. \tag{2.6}$$

Validity is a critical consideration, quantified by the **percentage invalid**, which reflects the fraction of molecules violating chemical valency rules or producing syntactically invalid SMILES.

Diversity is equally important, with the **percentage unique** capturing the proportion of distinct molecules generated in a single run. For retrosynthesis, **reactant prediction accuracy** evaluates the correctness of predicted reactants from a given product. Additionally, the **percentage solved** assesses success in multi-step synthesis, determining the fraction of target molecules that can be traced back to purchasable starting materials.

These metrics collectively narrate the journey of generative models, from reconstructing molecular structures to designing innovative synthetic routes.

2.4.3 XAI Metrics

Explainable AI (XAI) metrics illuminate the decision-making processes of models, validating their predictions and providing insight into their internal workings. One powerful tool is **cosine distance**, which compares explanations from different models or input segments, such as SMILES strings encoding the same molecule:

$$\text{Cosine Distance} = 1 - \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|}, \tag{2.7}$$

where $\mathbf{A}$ and $\mathbf{B}$ are vectors representing feature importances.

Another metric, **entropy**, quantifies the concentration of attention on specific features:

$$H = -\sum_{i=1}^{n} p_i \log(p_i), \tag{2.8}$$

where p_i denotes the normalized importance of the i-th feature. Lower entropy suggests focused attention on critical features, while higher entropy indicates more distributed importance.

Lastly, **relative structural alert importance** evaluates the emphasis placed on toxicologically relevant substructures compared to the overall molecule. Together, these metrics provide a rich tapestry of interpretability, enhancing trust and understanding in AI-driven molecular predictions.

2.5 Training Paradigms in ML

Training machine learning models is a process of optimizing parameters to minimize error on specific tasks. Training paradigms vary depending on the goals and data availability, ranging from pre-training large models on extensive datasets to fine-tuning them for specialized tasks. Below, the primary approaches employed in this thesis are detailed.

2.5.1 Pre-training

Pre-training serves as the foundation for many deep learning models, particularly in domains where labeled data is limited. It involves training models on broad, often unlabeled datasets to learn generalizable patterns, which are later fine-tuned for specific tasks. Strategies employed in this thesis were masked language modeling and auto-translation. In masked language modeling, portions of input data are deliberately masked, and the model is tasked with predicting the missing content. Auto-translation is the process to translate from one representation to another representation of the same entity. In the case of SMILES, this is often from a randomized SMILES representation to a canonical SMILES. By training the model to reconstruct the missing spans or translate one representation into another, it develops a broader understanding of syntactic and semantic relationships, enabling better generalization.

2.5.2 Fine-tuning

Fine-tuning adapts pre-trained models to specific tasks by continuing the training process on domain-specific datasets. Unlike pre-training, which focuses on general features, fine-tuning targets the nuances of a given task, such as predicting molecular properties or generating retrosynthetic pathways. This stage often involves smaller datasets and adjusted hyper-parameters to avoid overfitting while leveraging the learned representations of the pre-trained model.

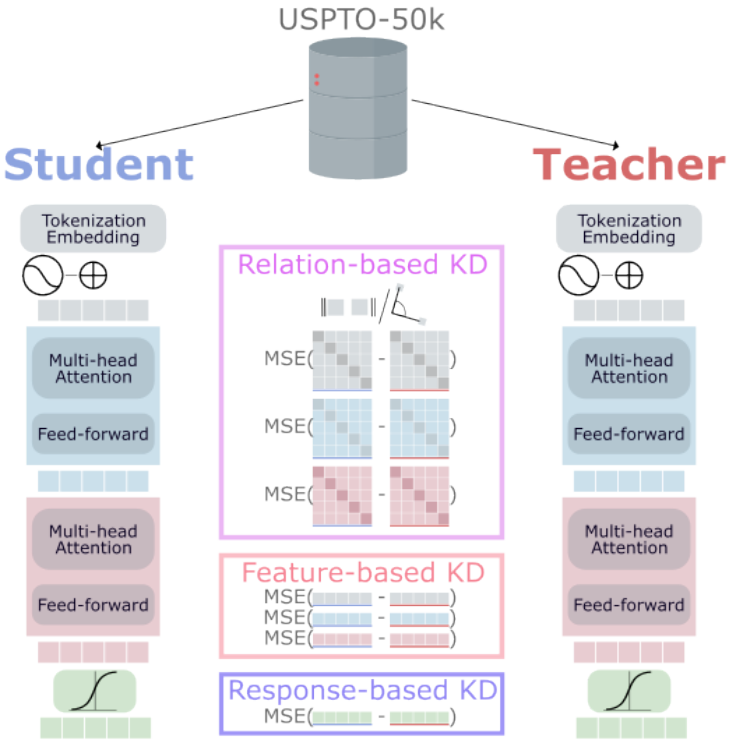


Figure 2.3 Illustrative example of different types of knowledge distillation. The information from a teacher model can be transferred to a student model in three ways. Response-based KD, which learns from the output of the teacher model. Feature-based KD, which learns from the internal representations. And Relation-based KD, which learns from the angles/distances between samples of the same models and correlates them between models.

2.5.3 Transfer Learning

Transfer learning facilitates knowledge reuse between related tasks by initializing a model with weights from pre-trained models. This approach is particularly advantageous in molecular machine learning, where the chemical space is vast, but labeled data for specific tasks may be scarce. By transferring learned representations from a general-purpose molecular model to a specialized task, transfer learning accelerates convergence and improves performance.

2.5.4 Knowledge distillation

Knowledge distillation (KD) [123] aims to compresses large, complex models into smaller, efficient ones while retaining their predictive power. The teacher model, typically a pre-trained and fine-tuned architecture, guides the training of a smaller student model by transferring knowledge in three ways: 1) response-based KD [123], 2) feature-based KD and 3) relation-based KD [124] (Figure 2.3). These approaches distill knowledge from the output, internal representations or the angles/distances in-between batches of internal representations, respectively.

3 Summary of contributed articles

This chapter summarizes my contributed articles in the context of the goals of this thesis.

3.1 Using test-time augmentation to investigate explainable AI: inconsistencies between method, model and human intuition

***Hartog, P. B.**, Krüger, F., Genheden, S., & Tetko, I. V. (2024). Using test-time augmentation to investigate explainable AI: inconsistencies between method, model and human intuition. Journal of Cheminformatics, 16(1), 39.*

(See also publication [1] in the bibliography and in Appendix A)

Synopsis

XAI is required to identify security and bias risks, enhance new discoveries by elucidating the reasoning behind predictions [125]. To this end, ML practitioners desire XAI to produce human-understandable and consistent explanations [126]. Notably, different XAI methods adopt different approximations to determine this importance in relation to the ML model [100, 127–130]. These distinct techniques come with their individual variances, underpinned by a foundation of internal model uncertainty that forms the basis for their explanations. As these XAI methods yield outcomes with varying degrees of uncertainty [131–133], it raises concerns about potential distrust among end-users.

In order to investigate faithfulness of current state-of-the-art XAI methods, we investigate the consistency of XAI methodology between methods, models and expert intuitions. We utilize test-time data augmentation from prior research conducted in domains including medical imaging, where data augmentation techniques, such as image rotation and contrast adjustment, have been used to measure model uncertainty [134, 135]. These strategies have demonstrated their efficacy in enhancing model performance, gauging model uncertainty, and increasing robustness. However, this methodology remains largely unexplored within the field of XAI concerning molecular representations, presenting an opportunity to assess the robustness of XAI methods.

We investigated predictions made in the field of computational toxicity, where faithfulness of predictions are crucial. Augmentation of SMILES-based molecular representations has been used successfully for transfer learning on downstream tasks. Using the predominant Transformer architecture and six different pre-training settings using either encoder-only or encoder-decoder training, we investigate XAI techniques in different settings comparable to how ML models are explored in the field. Additionally, using the curated Ames dataset [136], we compared XAI explanations not only for internal consistency and faithfulness, but also with expert-derived structural alerts. The Ames test [137] serves as an important screen for mutagenic compounds [138].

In this study, we performed an analysis of eight XAI methods, six pre-training settings and two transformer architectures. Using these settings in combination with test-time augmentation, we were able to test the robustness of XAI methods in-between model, method and samples using average cosine distances of multiple explanations of toxicity prediction representing the same ground-truth. After pre-training and transfer learning, model accuracy scores ranged from 0.5 in the random model to 0.74 for the highest trained model.

We show substantial disagreement between augmented SMILES, even when canonical SMILES show no more difference than randomized SMILES. Interestingly, by analyzing the explanation consistency over multiple representations of the same molecule, each XAI method gave very similar incon-

sistencies, regardless of the model. Furthermore, the random model showed no significant difference in XAI robustness when compared with the trained models, and testing on training data actually decreased robustness of XAI methods. Furthermore, the entropy, a measure of the information contained in an explanation, and relative importance given to expert-derived structural alerts were all conserved, regardless of whether the model was random or trained.

In general, our findings indicate a greater need to identify what XAI methods measure, and specifically to remove any confounding background information. In our case, all methods in this context seemed to rely not on learned gradients, but likely instead on the tokenization. While this could be the ground truth explanation, randomization experiments indicated that this is not the case. However, further comparisons, specifically with regards to randomization should help improve robustness in XAI methods. Although, as Adebayo et al. [131] discussed, randomization can indicate architecture-based priors of XAI, XAI should reflect learned parameters to explain model behavior. In the case of large ML models, neuron activation will differ even for identical models based on simple initializations. Having consistent XAI methods is crucial not just for reliability, but also to compare different ML models and gain insights on their internal function.

Test-time augmentation, in models that are not inherently invariant to data augmentation techniques, can be used to test XAI robustness. However, it requires a comparison to randomized and applicability domain to draw well-founded conclusions. Furthermore, test-time augmentation can be used as method to increase consistency between XAI methods and over different models, We identified that neither the amount of information nor the relative importance given to expert-derived structural alerts changes when using averaged values, whilst increasing overall consistency between similar models and XAI methods. This indicates that test-time augmentation can be used to make XAI more invariant, but not to improve XAI attribution overall. An interesting intuition from this result is that redundancy in the model likely leads different representations of the same ground-truth to take multiple routes from input to prediction. In conclusions, XAI methods should reflect learned parameters and test-time augmentation is therefore an important metric to use in testing text-based ML model consistency in future applications.

My contributions

1. I conceptualized the study together with Igor Tetko
2. I re-implemented the full methodology of Karpov et al. [78].
3. I produced a implementation of six XAI methods discussed in Qiang et al. [130] to be compatible with the workflow of my work.
4. I performed all data cleaning, processing and augmentation.
5. I performed all model training, validation and testing.
    a) To study the possible effect of pre-training, I trained six pre-training regimens for each of the two model types explored for pre-training.
    b) I fine-tuned each of these 12 models further for Ames toxicity.
    c) I gathered all eight XAI explanations for each molecule and enumeration in both test and training sets for all 12 models and baseline comparisons including untrained model architectures.
6. I performed all data analysis.
    a) I created a script to identify identical atoms in randomized SMILES of the same molecule using RDKit.
    b) I contrasted the cosine distances between explanations of different models, XAI methods and enumerations.
    c) I corrected SMARTS from Kazius et al. [116] to serve as expert-derived structural alerts.
    d) I evaluated the entropy and attribution breakdown of XAI explanations.

e) I performed statistical analysis between populations of randomized and canonical SMILES.

7. I interpreted the results, wrote and edited the manuscript and created all figures, with exception of supplementary Figure 11. All co-authors contributed to the discussion of the results and reviewing of the final manuscript.

3.2 Investigations into the efficiency of computer-aided synthesis planning

***Hartog, P. B. R.**, Westerlund, A. M., Tetko, I. V., & Genheden, S. (2025). Investigations into the Efficiency of Computer-Aided Synthesis Planning. Journal of Chemical Information and Modeling.*

(See also publication [2] in the bibliography and in Appendix B)

Synopsis

The efficiency of ML models is essential for their integration into larger systems, particularly those that rely on frequent model queries, such as reinforcement learning environments or recursive optimization frameworks. ML models, despite their wide applicability, face criticism for inefficiencies and high carbon footprints, making optimization crucial for their sustainable deployment [139]. In the context of drug discovery, ML models have become indispensable tools, particularly in computer-aided synthesis planning. Retrosynthesis analysis predicts synthetic pathways from a target molecule to purchasable starting materials, utilizing single-step models to identify reactants from product molecules [46]. These single-step predictions form the backbone of multi-step tree search algorithms, where their performance significantly influences the overall efficiency and feasibility of retrosynthesis. Optimizing single-step retrosynthesis models is therefore critical, as their frequent usage in multi-step searches exacerbates computational costs and environmental impacts [67].

Through three established strategies, we optimize single-step text-based retrosynthesis models to improve their efficiency using Chemformer as a baseline. First, we explored alternative Transformer architectures to identify designs that offered computational advantages, including the Switch Transformer and the Perceiver. Next, knowledge distillation played a pivotal role, where we employed feature-based and response-based strategies to transfer insights from a larger teacher model to smaller, faster student counterpart. Finally, hyper-parameter tuning was performed, aiming to strike a balance between reducing inference time and maintaining robust accuracy.

Performance was evaluated through metrics including inference time, top-1 accuracy and top-10 accuracy. Initial experiments highlighted limitations in alternative Transformer architectures, which underperformed when combined with KD. In contrast, the application of KD and hyper-parameter optimization to Chemformer demonstrated measurable improvements in inference times without compromising accuracy. Furthermore, the study analyzed the implications of single-step optimizations in multi-step retrosynthesis. While reducing model size and improving single-step inference speed yielded up to threefold gains, multi-step efficiency was more strongly influenced by the diversity and confidence of single-step predictions. Multi-step searches required between 400 and 1700 model calls per route, underscoring that the number of calls, rather than the speed of individual calls, was the primary determinant of computational cost. Top-10 accuracy and prediction diversity correlated positively with success rates, indicating that broader exploration improved solvability in multi-step tasks.

A supplementary analysis of route similarity revealed that optimized models produced synthesis pathways comparable to those generated by larger, slower models. This suggests that efficiency improvements do not degrade the quality of multi-step outputs.

The findings of this study emphasize the nuanced relationship between single-step and multi-step retrosynthesis efficiency. Although faster single-step models are beneficial, multi-step performance is dominated by factors such as call diversity and exploration-exploitation balance. Higher top-10 accuracy and increased model calls were positively associated with solvability, while an inverse relationship between solvability and top-1 accuracy highlighted the trade-offs between exploration and precision. These insights are critical for designing efficient synthesis planning frameworks.

The study underscores the importance of optimizing single-step retrosynthesis models using KD and related techniques to enhance their efficiency. However, factors such as top-10 accuracy and the number of model calls play a more critical role in determining multi-step efficiency. Future research should prioritize balancing these factors to improve the scalability of synthesis planning frameworks.

My contributions

1. I conceptualized the study together with Annie Westerlund and Samuel Genheden
2. I adapted code from Irwin et al. [44] to an upgraded environment.
3. I implemented knowledge distillation through Yang et al. [140].
4. I implemented the Perceiver model and adapted code from Varuna Jayasiri [141] to create a Switch Transformer implementation.
5. I performed all model training, validation and testing, with the exception of the pre-training which was gathered from the original Irwin et al. [44]. Including:
    a) Knowledge distillation strategies across teacher-student model pairs.
    b) Hyper-parameter optimization to balance speed and accuracy.
    c) Multi-step retrosynthesis predictions using each of the single-step models, and the small template-based model as baseline.
6. Route similarity analysis was performed by me working closely together with Samuel Genheden.
7. I interpreted the results, wrote and edited the manuscript and created all figures. All co-authors contributed to the discussion of the results and reviewing of the final manuscript.

4 Discussion and future outlook

> The scientific enterprise as a whole does from time to time prove useful, open up new territory, disclose new relationships, and suggest new insights. But these achievements are secondary to the main purpose of normal science, which is to extend the scope and precision of the scientific paradigm.
>
> Thomas Kuhn, *The Structure of Scientific Revolutions*

The application of ML in drug discovery has significantly transformed the field, offering innovative solutions for identifying and optimizing therapeutic compounds. However, for ML to continue driving advancements, its methodologies must be both efficient and interpretable. Representations such as SMILES have gained traction due to their compatibility with natural language processing (NLP) models and their ability to streamline molecular property predictions and reaction modeling. However, text-based molecular representations provide challenges for ML practitioners, including the need for robust explainability methods and efficient integration into larger workflows. Providing insights into the potential and limitations of text-based molecular representations for drug discovery will allow ML to continue to revolutionize the field of drug discovery.

4.1 Summary

This cumulative thesis investigates crucial aspects of text-based molecular representations in ML for drug discovery, emphasizing their interpretability and computational efficiency. Text-based representations, particularly SMILES, have revolutionized molecular modeling by offering a structured, linearized format that integrates seamlessly with established NLP architectures. These representations have facilitated advancements in molecular property prediction, retrosynthesis modeling, and reaction prediction, making them indispensable tools in cheminformatics. However, as ML adoption in drug discovery grows, understanding and addressing the limitations of these representations remain essential for their continued utility.

In section 3.1, I focused on explainability and the faithfulness of current XAI methods applied to text-based molecular representations. By employing augmentation during test-time, we were able to compare the explanations given to evaluations of the same ground-truth and thereby investigating the stability of explanations in computational toxicity prediction using SMILES. The findings reveal significant instability in XAI methods, as explanations often diverge for identical molecules encoded differently in SMILES. This work introduces SMILES augmentation as a diagnostic tool for assessing XAI consistency and highlights the limitations of text-based representations in ensuring faithful model interpretability. These contributions set the stage for future exploration of hybrid representations and improved validation methodologies for AI-driven drug discovery.

In section 3.2, I shift to the efficiency of text-based ML models, particularly for their use in larger systems. Using Chemformer as a baseline, the research explores alternative Transformer architectures, knowledge distillation, and hyper-parameter tuning to optimize computational efficiency. While single-step retrosynthesis models demonstrated improved speed and accuracy, I concluded that multi-step

retrosynthesis workflows are more heavily influenced by the diversity and confidence of single-step predictions than by their computational speed. These results underline the importance of balancing computational efficiency with the exploration required for complex synthetic planning. Additionally, the findings underscore the role of Monte Carlo-based methods in navigating exploration-exploitation trade-offs in multi-step retrosynthesis, offering valuable insights for future model development and integration.

4.2 Improvements of XAI for Text-Based Molecular Representations

Explainable AI (XAI) remains a crucial area for integrating ML models into high-stakes domains like drug discovery. For text-based molecular representations, advancing XAI methods is essential to ensure that predictions are interpretable and actionable.

4.2.1 Creating more Effective Evaluation of XAI

Effective evaluation methodologies are critical for XAI, as they allow researchers to assess whether XAI techniques provide reliable and meaningful insights into model behavior. Without robust evaluation frameworks, the integration of ML models into sensitive fields such as drug discovery remains problematic. To address these challenges, this section explores how to enhance the evaluation of XAI methods. Building upon findings from section 3.1, which highlighted the instability of XAI explanations for SMILES-based representations, I discuss strategies to increase the effectiveness of XAI evaluation and provide a nuanced understanding of faithfulness, validity, and data dependencies in XAI.

Faithfulness over Validity In general, XAI methods are often validated using criteria based on human intuition or expert knowledge about what predictions should focus on. While this approach provides a starting point, it assumes that ML models rely on similar mechanisms to make their predictions, which is not always the case [1]. Faithfulness, defined as the degree to which an explanation reflects the true reasoning process of the model, must take precedence over validity, which measures whether an explanation aligns with human expectations. section 3.1 demonstrated that expert-derived structural alert, a stand-in for human intuition, could not differentiate between stable and instable explanations. Augmentation techniques, such as SMILES augmentation or image rotations, could serve as a diagnostic tool to reveal inconsistencies in XAI explanations, and serve as a measure of faithfulness. SMILES augmentation as a measure for faithfulness was inspired by similar techniques validating XAI for image-based models, where rotation invariance was used as a measure for uncertainty [134, 135]. Models might correctly predict molecular properties but base their predictions on features unrelated to established chemical principles. Hence, evaluation methodologies should incorporate techniques that directly measure the alignment between explanations and the internal reasoning rather than relying solely on external domain expertise, which will also enable prospective data analysis of fields without substantial expert knowledge. Failing to provide an explanation for a prediction is bad, but providing false reasoning is worse.

Explainability Cliff Another important aspect that need further evaluation is the possibility of an "explainability cliff". This concept suggests that the validity of an explanation might strongly correlate with the accuracy of the underlying model. For high-performing models, explanations are often perceived as valid because they align with expected outcomes. However, as model accuracy decreases, it remains the question if explanation faithfulness deteriorates similarly. Previous research has shown that XAI evaluations do change depending on model parameters [142, 143]. This raises critical questions about whether explanations from lower-accuracy models are genuinely interpretable or merely artifacts of random noise. To address this, comparisons with baseline methods, such as random predictions and explanations from randomly initialized models, are essential. These comparisons help distinguish meaningful explanations from background noise and highlight whether XAI methods are

producing insights that genuinely reflect learned patterns rather than spurious correlations. In section 3.1, I evaluated models ranging from random accuracy to medium-range accuracy and found no significant difference between them. This may point to an explainability cliff, where a model must reach a certain accuracy, before XAI becomes faithful.

Data-Dependent Explanations Explanations generated by XAI methods are inherently dependent on the data for which they provide the explanation. In essence, explanations generated for two points of data may explain internal model reasoning differently. This data dependency arises from the phenomenon of neuronal polysemanticity, where individual neurons in ML models can encode different concepts depending on the input data [144]. For example, in SMILES-based molecular representations, a neuron might capture chemical properties like aromaticity for one molecule while encoding toxicity-related features for another. Such flexibility can complicate XAI evaluation, as the relevance of an explanation may vary significantly across inputs. Effective XAI methodologies must account for this context dependence, ensuring that explanations remain consistent and meaningful across diverse data scenarios. Additionally, visualization techniques and clustering methods can be used to analyze how neurons encode different features, providing deeper insights into the model's decision-making processes.

By addressing these challenges, the field can move toward more robust and faithful XAI methods, ultimately enhancing the integration of ML models into drug discovery workflows. The insights gained from improved evaluation frameworks will not only increase confidence in model predictions but also pave the way for the development of hybrid representations and more faithful XAI approaches.

4.2.2 Promising Avenues for Robust XAI

As the field of XAI continues to evolve, there are several promising approaches bing investigated currently. Text-based molecular representations face unique challenges in achieving robust explanations due to their inherent limitations. However, emerging methodologies such as multi-modal representations, sparse autoencoders, and leveraging training data for explanations offer exciting opportunities to overcome these challenges.

Multi-modal Representations for Robust XAI In recent research, Bertolini et al. [145] showed that you can increase the interpretability of your model by using an alternative modality of the input. In this approach, SMILES-based models were complemented with image representations, enabling the transfer of learned features through a shared internal representation. SMILES, despite its widespread adoption, suffers from an inherent instability in representing molecular structures. This lack of order-invariance complicates internal representations and therefore XAI analyses, as explanations for similar molecules can vary drastically. In contrast, alternative modalities like molecular graphs or image-based representations maintain structural similarity, offering greater consistency in feature extraction and model explanations. XAI methods applied to these hybrid models exhibited improved stability, suggesting that integrating modalities might mitigate the shortcomings of SMILES [145].

Multi-modal representations provide a compelling avenue for improving the robustness of XAI methods in molecular modeling [146]. By combining text-based representations, such as SMILES, with complementary modalities like 3D molecular graphs or image-based encodings, models can leverage diverse data sources to produce more reliable explanations. However, multi-modal approaches come with significant challenges. One major issue is modal collapse, where a single modality dominates the decision-making process, potentially undermining the contributions of other data types [147]. In drug discovery, this imbalance could lead to models that overly rely on graph-based data while neglecting the rich semantic information in text-based encodings. Exploring diverse representations can bolster

both model performance and interpretability, but future applications will need to keep modal collapse in account.

Explaining predictions through Mechanistic Interpretability Mechanistic interpretability aims to elucidate the internal workings of machine learning models by identifying how specific components contribute to their predictions. Sparse autoencoders (SAEs) and concept embedding models (CEMs) offer significant potential for enhancing the interpretability of text-based molecular representations. These methods function by transforming input data into an intermediate, often higher-dimensional space, enforcing sparsity to highlight key features relevant to a given task. This sparsity constraint compels the model to focus on the most informative aspects of the data, potentially aligning with chemically meaningful concepts such as functional groups or bonding patterns. For instance, in toxicity prediction, an SAE might emphasize toxicophores, directly linking specific substructures to adverse effects. This approach contrasts with post-hoc explanation methods, as SAEs aim to build interpretability directly into the model's architecture.

However, mechanistic interpretability may suffer significantly in the space of faithfulness. Sparsity may be used by the ML model to encode concepts that are still in a higher dimension [144]. Additionally, a significant challenge in mechanistic interpretability is the manual interpretation of the learned sparse features. In domains like natural language processing, the vast number of potential concepts can make this process prohibitively complex. Fortunately, molecular data is inherently more structured, with a finite set of well-defined chemical features. This constraint reduces the complexity of interpreting the sparse features learned by the model, making it more feasible to map them to known chemical concepts. Finally, while SAEs can highlight important features, they do not inherently provide causal explanations, necessitating further analysis to understand the relationships between identified features and model predictions. Ongoing research is focused on developing methods to automate the interpretation of sparse features and integrate causal inference techniques to enhance the explanatory power of these models.

Explaining Predictions with Data Attribution One promising approach to improving XAI involves linking predictions back to specific training data, using the training examples as a proxy for understanding model behavior. This method helps users understand which examples in the training set most influenced a particular prediction. In the realm of text-based molecular representations, this is particularly advantageous, as it allows for the assessment of whether predictions align with known chemical patterns or structure-activity relationships. More importantly, This approach also facilitates the incorporation of auxiliary information associated with the training data, such as experimental conditions or previously observed biological activities, which may not have been explicitly included during model training. Additionally, use of counterfactuals can aid users in understanding the context in which a prediction was made [97, 98].

However, implementing data attribution methods requires careful consideration of applicability and potential artifacts. One concern is that models might rely on spurious correlations present in the training data, leading to misleading attributions. Additionally, the computational complexity of tracing predictions back to training examples can be substantial, especially in large datasets. This complexity can also lead to misleading attributions, where one sparse data point is attributed with more influence than a data point from a dense training population. To mitigate these issues, researchers are exploring efficient algorithms for data attribution and developing techniques to discern genuine patterns from artifacts. Ensuring the robustness of these methods is crucial for their effective application in drug discovery and other high-stakes fields.

As XAI methodologies advance, it is clear that robust explanations require a combination of strategies. Multi-modal representations provide additional structural context, sparse autoencoders and concept embedding models offer structured interpretability, and data attribution techniques connect pre-

dictions back to their origins. However, each of these approaches presents its own challenges, from balancing multi-modal contributions to ensuring faithful interpretations of learned features. Future work must continue to refine these methods, ensuring that XAI remains both reliable and applicable to real-world drug discovery challenges. The ongoing development of these techniques will be instrumental in bridging the gap between model transparency and practical utility, ultimately fostering greater trust and adoption of ML-driven solutions in cheminformatics.

4.3 Utilizing Individual ML models in Larger Systems: A case study of Computer-Aided Synthesis Planning and ML Models in Drug Discovery

ML-driven synthesis planning has emerged as a cornerstone of modern drug discovery, enabling researchers to streamline the complex processes involved in retrosynthesis and reaction optimization. Retrosynthesis planning involves predicting the series of reactions needed to synthesize a target molecule from purchasable starting materials. Traditional approaches to synthesis planning have relied heavily on expert intuition and rule-based systems, but ML models have introduced new opportunities for data-driven and automated workflows. Current retrosynthesis tools often utilize ML models as oracles embedded within Monte Carlo Tree Search (MCTS) frameworks [46, 47]. These models predict possible reactants for a given target molecule, guiding the MCTS algorithm toward feasible synthetic routes. While effective, this approach often treats individual ML models as isolated components, focusing primarily on top-1 or top-10 prediction accuracy without leveraging additional aspects of the model's output, such as uncertainty estimates or reaction diversity. This results in a search where the exploration of possible full routes is performed on metrics unlinked to preferred route properties.

In section 3.2, I outline the limitations of this approach, particularly in multi-step retrosynthesis workflows. Single-step models are typically optimized independently, leading to potential inefficiencies in multi-step frameworks where exploration-exploitation trade-offs dominate. Incorporating uncertainty and diversity into model predictions could significantly improve the robustness of synthetic route planning. One option is to integrate ensemble-based approaches or probabilistic methods which could allow the MCTS framework to explore alternative reaction pathways, thereby increasing the likelihood of identifying efficient and feasible synthetic routes. However, I believe that my findings underscore the need for more holistic integration of ML models, where their full predictive potential is utilized to enhance both accuracy and adaptability within a larger multi-step synthesis workflow. Further adaptation, after training, can significantly aid the use of larger ML models in such workflows, where computational capacity will be a limiting factor in current applications [2].

Foundation Models Foundation models represent a transformative opportunity for small-molecule drug discovery. By pre-training on massive, diverse datasets, these models capture generalizable patterns applicable across a range of tasks[85]. For synthesis planning, a foundation model could unify multiple sub tasks within a single framework, reducing the fragmentation of current approaches and improving overall efficiency. The potential of foundation models lies in their ability to serve as an integrated platform for predicting reactants, selecting reagents, and optimizing pathways, all while leveraging their broad knowledge base. However, realizing this vision requires addressing key challenges. Scalability remains a significant hurdle, as training foundation models with sufficient chemical diversity demands vast computational resources. Fine-tuning these models for domain-specific applications also presents difficulties, particularly in ensuring that their predictions remain relevant and actionable in practical scenarios. Moreover, while foundation models excel in general tasks, their performance in highly specialized applications may be limited compared to possible alternative approaches.

Reinforcement Learning with Human Feedback Reinforcement Learning with Human Feedback (RLHF) has emerged as a technique for aligning machine learning models with expert knowledge and

user-defined objectives. RLHF offers a promising avenue for tailoring ML models to better meet the specific needs of synthesis planning [53, 54]. Unlike static models, RLHF dynamically refines its predictions by incorporating user-defined reward functions, enabling optimization for objectives such as speed, accuracy, cost efficiency, or environmental impact. In the context of retrosynthesis, RLHF could also guide the overall search to prefer basic reactions and smaller synthetic routes, with less overall single model calls during inference. Another significant advantage of RLHF is its flexibility in defining custom reward functions that reflect real-world priorities. For example, integrating physics-based models as auxiliary reward functions could guide the ML model toward synthesizable reactions, even if data for these reactions are sparse. This hybrid approach combines the generalizability of data-driven ML models with the specificity of physics-based simulations. However, RLHF also presents challenges, including the computational cost of iterative optimization and the risk of overfitting to narrow objectives. Striking a balance between generalization and task-specific adaptation will be crucial for RLHF's success in synthesis planning.

Agentic Workflows Agentic workflows represent an emerging paradigm in AI-driven synthesis planning, where multiple autonomous models collaborate to design, evaluate, and refine predictions [148, 149]. Instead of relying on a single ML model to perform retrosynthesis, agentic workflows could distribute tasks across specialized AI agents that operate independently but communicate within a shared framework. This approach leverages modularity, allowing different models to contribute to the synthesis process based on their strengths. For example, in an agentic workflow, one model could generate candidate synthetic pathways, another could assess their feasibility based on reaction constraints, and yet another could optimize conditions for yield and efficiency. By integrating reinforcement learning and foundation models within this framework, these agents can iteratively refine their decisions, ensuring that the final synthetic plan is both practical and optimized for real-world execution. One advantage of agentic workflows is their ability to dynamically adapt to new data, making them well-suited for continuously evolving chemical datasets. Additionally, they provide a scalable framework for incorporating new AI techniques as they emerge, ensuring that synthesis planning systems remain state-of-the-art. However, the success of agentic workflows depends on effective coordination between agents, requiring advances in communication protocols and multi-agent reinforcement learning strategies.

As synthesis planning continues to evolve, the integration of foundation models, RLHF, and agentic workflows offers a pathway toward more robust and efficient retrosynthetic strategies. Foundation models provide broad knowledge and scalability, but their computational costs and lack of task-specific optimization limit their immediate applicability. RLHF introduces adaptability by aligning models with human-defined objectives, but its iterative nature makes it computationally expensive. Agentic workflows present a compelling solution by distributing tasks across specialized agents, enabling a balance between generalization and specialization. Future research should focus on optimizing interactions between these components, ensuring that synthesis planning models can leverage the strengths of each approach while mitigating their respective limitations.

4.4 Conclusion

Overall, this thesis thoroughly investigates the current use of text-based molecular representations for ML models in drug discovery. Through analysis of XAI methods applied to these text-based representations, it highlights the need to fundamentally re-imagining their lack of invariance and explainability. Multi-modal models present a compelling opportunity to unify information from disparate sources. By integrating text-based, graph-based, and physics-based representations, these models can leverage the strengths of each modality. Such an approach not only improves molecular modeling but also fosters collaboration between data-driven and physics-based methodologies, bridging gaps in prediction accuracy and interpretability. Furthermore, to further the use of larger ML models in overarching systems,

these systems will need to be optimized to use the internal models rather then using them passively as oracles. In conclusion, this thesis underscores the importance of balancing interpretability and efficiency in ML for drug discovery. By addressing the limitations of SMILES, optimizing retrosynthesis workflows, and exploring multi-modality integration, future research can pave the way for more robust and scalable solutions in drug discovery.

Bibliography

[1] Peter BR Hartog, Fabian Krüger, Samuel Genheden, and Igor V Tetko. Using test-time augmentation to investigate explainable ai: inconsistencies between method, model and human intuition. *Journal of Cheminformatics*, 16(1):39, 2024.

[2] Peter BR Hartog, Annie M Westerlund, Igor V Tetko, and Samuel Genheden. Investigations into the efficiency of computer-aided synthesis planning. *Journal of Chemical Information and Modeling*, 2025.

[3] Peter BR Hartog, Emma Svensson, Lewis Mervin, Samuel Genheden, Ola Engkvist, and Igor V Tetko. Registries in machine learning-based drug discovery: A shortcut to code reuse. In *International Workshop on AI in Drug Discovery*, pages 98–115. Springer, 2024.

[4] Andrea Hunklinger, Peter Hartog, Martin Šícho, Guillaume Godin, and Igor V Tetko. The openochem consensus model is the best-performing open-source predictive model in the first euos/slas joint compound solubility challenge. *SLAS Discovery*, 29(2):100144, 2024.

[5] Daniel Lowe. Chemical reactions from US patents (1976-Sep2016), 6 2017. URL https://figshare.com/articles/dataset/Chemical_reactions_from_US_patents_1976-Sep2016_/5104873.

[6] Fabio Pammolli, Laura Magazzini, and Massimo Riccaboni. The productivity crisis in pharmaceutical r&d. *Nature reviews Drug discovery*, 10(6):428–438, 2011.

[7] Elina Petrova. Innovation in the pharmaceutical industry: The process of drug discovery and development. In *Innovation and Marketing in the Pharmaceutical Industry: Emerging Practices, Research, and Policies*, pages 19–81. Springer, 2013.

[8] Geoffrey Kabue Kiriiri, Peter Mbugua Njogu, and Alex Njoroge Mwangi. Exploring different approaches to improve the success of drug discovery and development projects: a review. *Future Journal of Pharmaceutical Sciences*, 6:1–12, 2020.

[9] European Parliament and Council of the European Union. Regulation (eu) 2019/933 of the european parliament and of the council of 20 may 2019 amending regulation (ec) no 469/2009 concerning the supplementary protection certificate for medicinal products. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32019R0933, 2019. Official Journal of the European Union, L153, 1–10.

[10] Natesh Singh, Philippe Vayer, Shivalika Tanwar, Jean-Luc Poyet, Katya Tsaioun, and Bruno O Villoutreix. Drug discovery and development: introduction to the general public and patient groups. *Frontiers in Drug Discovery*, 3:1201419, 2023.

[11] PharmaSource. Cdmo explained: An overview of contract development and manufacturing organisations in pharma, 2024. URL https://pharmasource.global/content/cdmo-explained-an-overview-of-contract-development-and-manufacturing-organisat Accessed: 2024-10-10.

[12] Mark E Nuttall. Drug discovery and target validation. *Cells Tissues Organs*, 169(3):265–271, 2001.

[13] EA Martis, R Radhakrishnan, and RR Badve. High-throughput screening: the hits and leads of drug discovery-an overview. *Journal of Applied Pharmaceutical Science*, (Issue):02–10, 2011.

[14] Eduardo Habib Bechelane Maia, Letícia Cristina Assis, Tiago Alves De Oliveira, Alisson Marques Da Silva, and Alex Gutterres Taranto. Structure-based virtual screening: from classical to artificial intelligence. *Frontiers in chemistry*, 8:343, 2020.

[15] Attila A Seyhan. Lost in translation: the valley of death across preclinical and clinical divide–identification of problems and overcoming obstacles. *Translational Medicine Communications*, 4 (1):1–19, 2019.

[16] Jack W Scannell, Alex Blanckley, Helen Boldon, and Brian Warrington. Diagnosing the decline in pharmaceutical r&d efficiency. *Nature reviews Drug discovery*, 11(3):191–200, 2012.

[17] Vivek Kaul, Sarah Enslin, and Seth A Gross. History of artificial intelligence in medicine. *Gastrointestinal endoscopy*, 92(4):807–812, 2020.

[18] Anita Ioana Visan and Irina Negut. Integrating artificial intelligence for drug discovery in the context of revolutionizing drug delivery. *Life*, 14(2):233, 2024.

[19] Alan Talevi, Juan Francisco Morales, Gregory Hather, Jagdeep T Podichetty, Sarah Kim, Peter C Bloomingdale, Samuel Kim, Jackson Burton, Joshua D Brown, Almut G Winterstein, et al. Machine learning in drug discovery and development part 1: a primer. *CPT: pharmacometrics & systems pharmacology*, 9(3):129–142, 2020.

[20] Arkadiusz Z Dudek, Tomasz Arodz, and Jorge Gálvez. Computational methods in developing quantitative structure-activity relationships (qsar): a review. *Combinatorial chemistry & high throughput screening*, 9(3):213–228, 2006.

[21] Ravichandran Veerasamy, Harish Rajak, Abhishek Jain, Shalini Sivadasan, Christapher P Varghese, Ram Kishore Agrawal, et al. Validation of qsar models-strategies and importance. *Int. J. Drug Des. Discov*, 3:511–519, 2011.

[22] Jessica Vamathevan, Dominic Clark, Paul Czodrowski, Ian Dunham, Edgardo Ferran, George Lee, Bin Li, Anant Madabhushi, Parantu Shah, Michaela Spitzer, et al. Applications of machine learning in drug discovery and development. *Nature reviews Drug discovery*, 18(6):463–477, 2019.

[23] Suresh Dara, Swetha Dhamercherla, Surender Singh Jadav, CH Madhu Babu, and Mohamed Jawed Ahsan. Machine learning in drug discovery: a review. *Artificial intelligence review*, 55(3):1947–1999, 2022.

[24] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

[25] Tom Ceska, Chun-Wa Chung, Rob Cooke, Chris Phillips, and Pamela A Williams. Cryo-em in drug discovery. *Biochemical Society Transactions*, 47(1):281–293, 2019.

[26] Genki Terashi, Xiao Wang, Devashish Prasad, Tsukasa Nakamura, and Daisuke Kihara. Deepmainmast: integrated protocol of protein structure modeling for cryo-em with deep learning and structure prediction. *Nature Methods*, 21(1):122–131, 2024.

[27] Jason Chen, Ayisha Zia, Albert Luo, Hanze Meng, Fengbin Wang, Jie Hou, Renzhi Cao, and Dong Si. Enhancing cryo-em structure prediction with deeptracer and alphafold2 integration. *Briefings in Bioinformatics*, 25(3):bbae118, 2024.

[28] John G Moffat, Fabien Vincent, Jonathan A Lee, Jörg Eder, and Marco Prunotto. Opportunities and challenges in phenotypic drug discovery: an industry perspective. *Nature reviews Drug discovery*, 16(8):531–543, 2017.

[29] Yuge Ji, Mohammad Lotfollahi, F Alexander Wolf, and Fabian J Theis. Machine learning for perturbational single-cell omics. *Cell Systems*, 12(6):522–537, 2021.

[30] Sophie Tritschler, Moritz Thomas, Anika Böttcher, Barbara Ludwig, Janine Schmid, Undine Schubert, Elisabeth Kemter, Eckhard Wolf, Heiko Lickert, and Fabian J Theis. A transcriptional cross species map of pancreatic islet cells. *Molecular Metabolism*, 66:101595, 2022.

[31] James Cantley, Decio Laks Eizirik, Esther Latres, and Colin M Dayan. Islet cells in human type 1 diabetes: from recent advances to novel therapies–a symposium-based roadmap for future research. *Journal of Endocrinology*, 259(1), 2023.

[32] Daniel Oropeza and Pedro Luis Herrera. Glucagon-producing α-cell transcriptional identity and reprogramming towards insulin production. *Trends in cell biology*, 34(3):180–197, 2024.

[33] Sydney Brenner and Richard A Lerner. Encoded combinatorial chemistry. *Proceedings of the National Academy of Sciences*, 89(12):5381–5383, 1992.

[34] Adrián Gironda-Martínez, Etienne J Donckele, Florent Samain, and Dario Neri. Dna-encoded chemical libraries: a comprehensive review with succesful stories and future challenges. *ACS Pharmacology & Translational Science*, 4(4):1265–1279, 2021.

[35] Kevin McCloskey, Eric A Sigel, Steven Kearnes, Ling Xue, Xia Tian, Dennis Moccia, Diana Gikunju, Sana Bazzaz, Betty Chan, Matthew A Clark, et al. Machine learning on dna-encoded libraries: a new paradigm for hit finding. *Journal of Medicinal Chemistry*, 63(16):8857–8866, 2020.

[36] Alfredo Martín, Christos A Nicolaou, and Miguel A Toledo. Navigating the dna encoded libraries chemical space. *Communications Chemistry*, 3(1):127, 2020.

[37] Chak Ming Leung, Pim De Haan, Kacey Ronaldson-Bouchard, Ge-Ah Kim, Jihoon Ko, Hoon Suk Rho, Zhu Chen, Pamela Habibovic, Noo Li Jeon, Shuichi Takayama, et al. A guide to the organ-on-a-chip. *Nature Reviews Methods Primers*, 2(1):33, 2022.

[38] Manna Dai, Gao Xiao, Ming Shao, and Yu Shrike Zhang. The synergy between deep learning and organs-on-chips for high-throughput drug screening: a review. *Biosensors*, 13(3):389, 2023.

[39] Jinseok Park, Yang Woo Kim, and Hee-Jae Jeon. Machine learning-driven innovations in microfluidics. *Biosensors*, 14(12):613, 2024.

[40] Zipeng Zhong, Jie Song, Zunlei Feng, Tiantao Liu, Lingxiang Jia, Shaolun Yao, Tingjun Hou, and Mingli Song. Recent advances in deep learning for retrosynthesis. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 14(1):e1694, 2024.

[41] Pavel Karpov, Guillaume Godin, and Igor V Tetko. A transformer model for retrosynthesis. In *International Conference on Artificial Neural Networks*, pages 817–830. Springer, 2019.

[42] Igor V Tetko, Pavel Karpov, Ruud Van Deursen, and Guillaume Godin. State-of-the-art augmented nlp transformer models for direct and single-step retrosynthesis. *Nature communications*, 11(1):5575, 2020.

[43] Philippe Schwaller, Teodoro Laino, Théophile Gaudin, Peter Bolgar, Christopher A Hunter, Costas Bekas, and Alpha A Lee. Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. *ACS central science*, 5(9):1572–1583, 2019.

[44] Ross Irwin, Spyridon Dimitriadis, Jiazhen He, and Esben Jannik Bjerrum. Chemformer: a pre-trained transformer for computational chemistry. *Machine Learning: Science and Technology*, 3 (1):015022, 2022.

[45] Marwin HS Segler and Mark P Waller. Neural-symbolic machine learning for retrosynthesis and reaction prediction. *Chemistry–A European Journal*, 23(25):5966–5971, 2017.

[46] Samuel Genheden, Amol Thakkar, Veronika Chadimová, Jean-Louis Reymond, Ola Engkvist, and Esben Bjerrum. Aizynthfinder: a fast, robust and flexible open-source software for retrosynthetic planning. *Journal of cheminformatics*, 12(1):70, 2020.

[47] Lakshidaa Saigiridharan, Alan Kai Hassen, Helen Lai, Paula Torren-Peraire, Ola Engkvist, and Samuel Genheden. Aizynthfinder 4.0: developments based on learnings from 3 years of industrial application. *Journal of Cheminformatics*, 16(1):57, 2024.

[48] Annie M Westerlund, Siva Manohar Koki, Supriya Kancharla, Alessandro Tibo, Lakshidaa Saigiridharan, Mikhail Kabeshov, Rocío Mercado, and Samuel Genheden. Do chemformers dream of organic matter? evaluating a transformer model for multistep retrosynthesis. *Journal of Chemical Information and Modeling*, 2024.

[49] Marwin HS Segler, Mike Preuss, and Mark P Waller. Planning chemical syntheses with deep neural networks and symbolic ai. *Nature*, 555(7698):604–610, 2018.

[50] Connor W Coley, Dale A Thomas III, Justin AM Lummiss, Jonathan N Jaworski, Christopher P Breen, Victor Schultz, Travis Hart, Joshua S Fishman, Luke Rogers, Hanyu Gao, et al. A robotic platform for flow synthesis of organic compounds informed by ai planning. *Science*, 365(6453): eaax1566, 2019.

[51] Dario Caramelli, Jarosław M Granda, S Hessam M Mehr, Dario Cambié, Alon B Henson, and Leroy Cronin. Discovering new chemistry with an autonomous robotic platform driven by a reactivity-seeking neural network. *ACS Central Science*, 7(11):1821–1830, 2021.

[52] Artem I Leonov, Alexander JS Hammer, Slawomir Lach, S Hessam M Mehr, Dario Caramelli, Davide Angelone, Aamir Khan, Steven O'Sullivan, Matthew Craven, Liam Wilbraham, et al. An integrated self-optimizing programmable chemical synthesis and reaction engine. *Nature Communications*, 15(1):1240, 2024.

[53] Thomas Blaschke, Josep Arús-Pous, Hongming Chen, Christian Margreitter, Christian Tyrchan, Ola Engkvist, Kostas Papadopoulos, and Atanas Patronov. Reinvent 2.0: an ai tool for de novo drug design. *Journal of chemical information and modeling*, 60(12):5918–5922, 2020.

[54] Hannes H Loeffler, Jiazhen He, Alessandro Tibo, Jon Paul Janet, Alexey Voronov, Lewis H Mervin, and Ola Engkvist. Reinvent 4: Modern ai–driven generative molecule design. *Journal of Cheminformatics*, 16(1):20, 2024.

[55] Michael Moor, Oishi Banerjee, Zahra Shakeri Hossein Abad, Harlan M Krumholz, Jure Leskovec, Eric J Topol, and Pranav Rajpurkar. Foundation models for generalist medical artificial intelligence. *Nature*, 616(7956):259–265, 2023.

[56] Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Nan Duan, and Bo Wang. scgpt: toward building a foundation model for single-cell multi-omics using generative ai. *Nature Methods*, pages 1–11, 2024.

[57] Mark Davies, Michał Nowotka, George Papadatos, Nathan Dedman, Anna Gaulton, Francis Atkinson, Louisa Bellis, and John P Overington. Chembl web services: streamlining access to drug discovery data and utilities. *Nucleic acids research*, 43(W1):W612–W620, 2015.

[58] Evan E Bolton, Yanli Wang, Paul A Thiessen, and Stephen H Bryant. Pubchem: integrated platform of small molecules and biological activities. In *Annual reports in computational chemistry*, volume 4, pages 217–241. Elsevier, 2008.

[59] Mark A Johnson, Gerald M Maggiora, et al. Concepts and applications of molecular similarity. *(No Title)*, 1990.

[60] Harry L Morgan. The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service. *Journal of chemical documentation*, 5(2):107–113, 1965.

[61] David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.

[62] Mario Krenn, Florian Häse, AkshatKumar Nigam, Pascal Friederich, and Alan Aspuru-Guzik. Self-referencing embedded strings (selfies): A 100% robust molecular string representation. *Machine Learning: Science and Technology*, 1(4):045024, 2020.

[63] Kenneth Atz, Francesca Grisoni, and Gisbert Schneider. Geometric deep learning on molecular representations. *Nature Machine Intelligence*, 3(12):1023–1032, 2021.

[64] Frank Noé, Alexandre Tkatchenko, Klaus-Robert Müller, and Cecilia Clementi. Machine learning for molecular simulation. *Annual review of physical chemistry*, 71(1):361–390, 2020.

[65] Dan Ofer, Nadav Brandes, and Michal Linial. The language of proteins: Nlp, machine learning & protein sequences. *Computational and Structural Biotechnology Journal*, 19:1750–1758, 2021.

[66] Jerret Ross, Brian Belgodere, Vijil Chenthamarakshan, Inkit Padhi, Youssef Mroueh, and Payel Das. Large-scale chemical language representations capture molecular structure and properties. *Nature Machine Intelligence*, 4(12):1256–1264, 2022.

[67] Jason D Shields, Rachel Howells, Gillian Lamont, Yin Leilei, Andrew Madin, Christopher E Reimann, Hadi Rezaei, Tristan Reuillon, Bryony Smith, Clare Thomson, et al. Aizynth impact on medicinal chemistry practice at astrazeneca. *RSC Medicinal Chemistry*, 15(4):1085–1095, 2024.

[68] Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

[69] S Hochreiter. Long short-term memory. *Neural Computation MIT-Press*, 1997.

[70] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[71] Sam Wiseman and Alexander M Rush. Sequence-to-sequence learning as beam-search optimization. *arXiv preprint arXiv:1606.02960*, 2016.

[72] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.

[73] Adam C Mater and Michelle L Coote. Deep learning in chemistry. *Journal of chemical information and modeling*, 59(6):2545–2559, 2019.

[74] Medard Edmund Mswahili and Young-Seob Jeong. Transformer-based models for chemical smiles representation: A comprehensive literature review. *Heliyon*, 2024.

[75] Robin Winter, Floriane Montanari, Frank Noé, and Djork-Arné Clevert. Learning continuous and data-driven molecular descriptors by translating equivalent chemical representations. *Chemical science*, 10(6):1692–1701, 2019.

[76] David Weininger, Arthur Weininger, and Joseph L Weininger. Smiles. 2. algorithm for generation of unique smiles notation. *Journal of chemical information and computer sciences*, 29(2):97–101, 1989.

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

[78] Pavel Karpov, Guillaume Godin, and Igor V Tetko. Transformer-cnn: Swiss knife for qsar modeling and interpretation. *Journal of cheminformatics*, 12(1):1–12, 2020.

[79] Shengchao Liu, Weili Nie, Chengpeng Wang, Jiarui Lu, Zhuoran Qiao, Ling Liu, Jian Tang, Chaowei Xiao, and Animashree Anandkumar. Multi-modal molecule structure–text model for text-based retrieval and editing. *Nature Machine Intelligence*, 5(12):1447–1457, 2023.

[80] Ana Sanchez-Fernandez, Elisabeth Rumetshofer, Sepp Hochreiter, and Günter Klambauer. Cloome: contrastive learning unlocks bioimaging databases for queries with chemical structures. *Nature Communications*, 14(1):7339, 2023.

[81] Ce Zhou, Qian Li, Chen Li, Jun Yu, Yixin Liu, Guangjing Wang, Kai Zhang, Cheng Ji, Qiben Yan, Lifang He, et al. A comprehensive survey on pretrained foundation models: A history from bert to chatgpt. *International Journal of Machine Learning and Cybernetics*, pages 1–65, 2024.

[82] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[83] Danny Hernandez, Jared Kaplan, Tom Henighan, and Sam McCandlish. Scaling laws for transfer. *arXiv preprint arXiv:2102.01293*, 2021.

[84] Shashank Subramanian, Peter Harrington, Kurt Keutzer, Wahid Bhimji, Dmitriy Morozov, Michael W Mahoney, and Amir Gholami. Towards foundation models for scientific machine learning: Characterizing scaling and transfer behavior. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 71242–71262. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/e15790966a4a9d85d688635c88ee6d8a-Paper-Conference.pdf.

[85] Devon Myers, Rami Mohawesh, Venkata Ishwarya Chellaboina, Anantha Lakshmi Sathvik, Praveen Venkatesh, Yi-Hui Ho, Hanna Henshaw, Muna Alhawawreh, David Berdik, and Yaser Jararweh. Foundation and large language models: fundamentals, challenges, opportunities, and social impacts. *Cluster Computing*, 27(1):1–26, 2024.

[86] Muhammad Awais, Muzammal Naseer, Salman Khan, Rao Muhammad Anwer, Hisham Cholakkal, Mubarak Shah, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Foundational models defining a new era in vision: A survey and outlook. *arXiv preprint arXiv:2307.13721*, 2023.

[87] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[88] Andreas Holzinger, Peter Kieseberg, Edgar Weippl, and A Min Tjoa. Current advances, trends and challenges of machine learning and knowledge extraction: from machine learning to explainable ai. In *Machine Learning and Knowledge Extraction: Second IFIP TC 5, TC 8/WG 8.4, 8.9,*

TC 12/WG 12.9 International Cross-Domain Conference, CD-MAKE 2018, Hamburg, Germany, August 27–30, 2018, Proceedings 2, pages 1–8. Springer, 2018.

[89] Bryce Goodman and Seth Flaxman. European union regulations on algorithmic decision-making and a "right to explanation". *AI magazine*, 38(3):50–57, 2017.

[90] Kate Vredenburgh. The right to explanation. *Journal of Political Philosophy*, 30(2):209–229, 2022.

[91] Arun Das and Paul Rad. Opportunities and challenges in explainable artificial intelligence (xai): A survey. *arXiv preprint arXiv:2006.11371*, 2020.

[92] Rabia Saleem, Bo Yuan, Fatih Kurugollu, Ashiq Anjum, and Lu Liu. Explaining deep neural networks: A survey on the global interpretation methods. *Neurocomputing*, 513:165–180, 2022. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2022.09.129. URL https://www.sciencedirect.com/science/article/pii/S0925231222012218.

[93] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

[94] Mateo Espinosa Zarlenga, Pietro Barbiero, Gabriele Ciravegna, Giuseppe Marra, Francesco Giannini, Michelangelo Diligenti, Frederic Precioso, Stefano Melacci, Adrian Weller, Pietro Lio, et al. Concept embedding models. In *NeurIPS 2022-36th Conference on Neural Information Processing Systems*, 2022.

[95] Scott Lundberg. A unified approach to interpreting model predictions. *arXiv preprint arXiv:1705.07874*, 2017.

[96] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[97] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

[98] Riccardo Guidotti. Counterfactual explanations and how to find them: literature review and benchmarking. *Data Mining and Knowledge Discovery*, 38(5):2770–2824, 2024.

[99] Leo Breiman. *Classification and regression trees*. Routledge, 2017.

[100] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks, 2017.

[101] Kelvin Xu. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*, 2015.

[102] Karen Simonyan. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[103] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[104] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[105] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 5(3):e00024–001, 2020.

[106] Robert Huben, Hoagy Cunningham, Logan Riggs Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations*, 2023.

[107] Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*, 2023.

[108] Phillip Huang Guo, Aaquib Syed, Abhay Sheshadri, Aidan Ewart, and Gintare Karolina Dziugaite. Robust unlearning via mechanistic localizations. In *ICML 2024 Workshop on Mechanistic Interpretability*, 2024.

[109] Gabriëlle Ras, Marcel van Gerven, and Pim Haselager. Explanation methods in deep learning: Users, values, concerns and challenges. *Explainable and interpretable models in computer vision and machine learning*, pages 19–36, 2018.

[110] Gintare Karolina Dziugaite, Shai Ben-David, and Daniel M Roy. Enforcing interpretability and its statistical impacts: Trade-offs between accuracy and interpretability. *arXiv preprint arXiv:2010.13764*, 2020.

[111] Will Fleisher. Understanding, idealization, and explainable ai. *Episteme*, 19(4):534–560, 2022.

[112] Greg Adamson. Explainable artificial intelligence (xai): A reason to believe? *Law Context: A Socio-Legal J.*, 37:23, 2020.

[113] David Mendez, Anna Gaulton, A Patrícia Bento, Jon Chambers, Marleen De Veij, Eloy Félix, María Paula Magariños, Juan F Mosquera, Prudence Mutowo, Michał Nowotka, et al. Chembl: towards direct deposition of bioassay data. *Nucleic acids research*, 47(D1):D930–D940, 2019.

[114] Teague Sterling and John J Irwin. Zinc 15–ligand discovery for everyone. *Journal of chemical information and modeling*, 55(11):2324–2337, 2015.

[115] Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor W Coley, Cao Xiao, Jimeng Sun, and Marinka Zitnik. Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development. *arXiv preprint arXiv:2102.09548*, 2021.

[116] Jeroen Kazius, Ross McGuire, and Roberta Bursi. Derivation and validation of toxicophores for mutagenicity prediction. *Journal of medicinal chemistry*, 48(1):312–320, 2005.

[117] Alan D McNaught, Andrew Wilkinson, et al. *Compendium of chemical terminology*, volume 1669. Blackwell Science Oxford, 1997.

[118] William J Wiswesser. Historic development of chemical notations. *Journal of chemical information and computer sciences*, 25(3):258–263, 1985.

[119] Paul G Dittmar, Nick A Farmer, William Fisanick, Reginald C Haines, and Joseph Mockus. The cas online search system. 1. general system design and selection, generation, and use of search screens. *Journal of Chemical Information and Computer Sciences*, 23(3):93–102, 1983.

[120] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[121] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International conference on machine learning*, pages 4651–4664. PMLR, 2021.

[122] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120): 1–39, 2022.

[123] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[124] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3967–3976, 2019.

[125] Sajid Ali, Tamer Abuhmed, Shaker El-Sappagh, Khan Muhammad, Jose M Alonso-Moral, Roberto Confalonieri, Riccardo Guidotti, Javier Del Ser, Natalia Díaz-Rodríguez, and Francisco Herrera. Explainable artificial intelligence (xai): What we know and what is left to attain trustworthy artificial intelligence. *Information fusion*, 99:101805, 2023.

[126] Markus Langer, Daniel Oster, Timo Speith, Holger Hermanns, Lena Kästner, Eva Schmidt, Andreas Sesing, and Kevin Baum. What do we want from explainable artificial intelligence (xai)?–a stakeholder perspective on xai and a conceptual model guiding interdisciplinary xai research. *Artificial Intelligence*, 296:103473, 2021.

[127] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf.

[128] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[129] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[130] Yao Qiang, Deng Pan, Chengyin Li, Xin Li, Rhongho Jang, and Dongxiao Zhu. Attcat: Explaining transformers via attentive class activation tokens. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 5052–5064. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/20e45668fefa793bd9f2edf19be12c4b-Paper-Conference.pdf.

[131] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31, 2018.

[132] Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. The (un) reliability of saliency methods. *Explainable AI: Interpreting, explaining and visualizing deep learning*, pages 267–280, 2019.

[133] Patrick Schwab and Walter Karlen. Cxplain: Causal explanations for model interpretation under uncertainty. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/3ab6be46e1d6b21d59a3c3a0b9d0f6ef-Paper.pdf.

[134] Guotai Wang, Wenqi Li, Michael Aertsen, Jan Deprest, Sébastien Ourselin, and Tom Vercauteren. Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks. *Neurocomputing*, 338:34–45, 2019.

[135] Murat Seckin Ayhan and Philipp Berens. Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks. In *Medical Imaging with Deep Learning*, 2022.

[136] Congying Xu, Feixiong Cheng, Lei Chen, Zheng Du, Weihua Li, Guixia Liu, Philip W Lee, and Yun Tang. In silico prediction of chemical ames mutagenicity. *Journal of chemical information and modeling*, 52(11):2840–2847, 2012.

[137] Pauline Gee, Dorothy M Maron, and Bruce N Ames. Detection and classification of mutagens: a set of base-specific salmonella tester strains. *Proceedings of the National Academy of Sciences*, 91(24):11606–11610, 1994.

[138] Markus Kamber, Sini Flückiger-Isler, Günter Engelhardt, Rudolf Jaeckh, and Errol Zeiger. Comparison of the ames ii and traditional ames test responses with respect to mutagenicity, strain specificities, need for metabolism and correlation with rodent carcinogenicity. *Mutagenesis*, 24 (4):359–366, 2009.

[139] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in nlp. *arXiv preprint arXiv:1906.02243*, 2019.

[140] Ziqing Yang, Yiming Cui, Zhipeng Chen, Wanxiang Che, Ting Liu, Shijin Wang, and Guoping Hu. TextBrewer: An Open-Source Knowledge Distillation Toolkit for Natural Language Processing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 9–16. Association for Computational Linguistics, 2020. URL https://www.aclweb.org/anthology/2020.acl-demos.2.

[141] Nipun Wijerathne Varuna Jayasiri. labml.ai annotated paper implementations, 2020. URL https://nn.labml.ai/.

[142] Amir-Hossein Karimi, Krikamol Muandet, Simon Kornblith, Bernhard Schölkopf, and Been Kim. On the relationship between explanation and prediction: A causal view. *arXiv preprint arXiv:2212.06925*, 2022.

[143] Pratinav Seth and Vinay Kumar Sankarapu. Bridging the gap in xai-why reliable metrics matter for explainability and compliance. *arXiv preprint arXiv:2502.04695*, 2025.

[144] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2, 2023.

[145] Marco Bertolini, Linlin Zhao, Floriane Montanari, and Djork-Arné Clevert. Enhancing interpretability in molecular property prediction with contextual explanations of molecular graphical depictions. In *International Workshop on AI in Drug Discovery*, pages 1–12. Springer, 2024.

[146] Dong Huk Park, Lisa Anne Hendricks, Zeynep Akata, Anna Rohrbach, Bernt Schiele, Trevor Darrell, and Marcus Rohrbach. Multimodal explanations: Justifying decisions and pointing to the evidence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8779–8788, 2018.

[147] Letitia Parcalabescu and Anette Frank. Mm-shap: A performance-agnostic metric for measuring multimodal contributions in vision and language models & tasks. *arXiv preprint arXiv:2212.08158*, 2022.

[148] Tamer Abuelsaad, Deepak Akkil, Prasenjit Dey, Ashish Jagmohan, Aditya Vempaty, and Ravi Kokku. Agent-e: From autonomous web navigation to foundational design principles in agentic systems. *arXiv preprint arXiv:2407.13032*, 2024.

[149] Shanghua Gao, Ada Fang, Yepeng Huang, Valentina Giunchiglia, Ayush Noori, Jonathan Richard Schwarz, Yasha Ektefaie, Jovana Kondic, and Marinka Zitnik. Empowering biomedical discovery with ai agents. *Cell*, 187(22):6125–6151, 2024.

A Using test-time augmentation to investigate explainable AI: inconsistencies between method, model and human intuition

Journal of Cheminformatics

RESEARCH Open Access

Using test-time augmentation to investigate explainable AI: inconsistencies between method, model and human intuition

Peter B. R. Hartog[1,2*], Fabian Krüger[2], Samuel Genheden[1] and Igor V. Tetko[2]

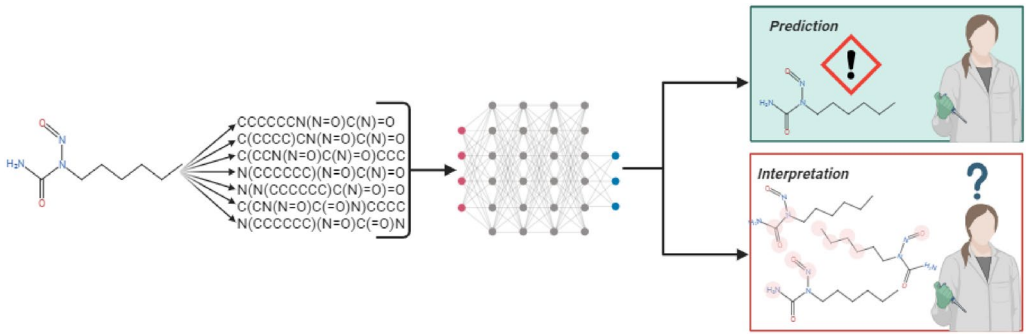
Abstract Stakeholders of machine learning models desire explainable artificial intelligence (XAI) to produce human-understandable and consistent interpretations. In computational toxicity, augmentation of text-based molecular representations has been used successfully for transfer learning on downstream tasks. Augmentations of molecular representations can also be used at inference to compare differences between multiple representations of the same ground-truth. In this study, we investigate the robustness of eight XAI methods using test-time augmentation for a molecular-representation model in the field of computational toxicity prediction. We report significant differences between explanations for different representations of the same ground-truth, and show that randomized models have similar variance. We hypothesize that text-based molecular representations in this and past research reflect tokenization more than learned parameters. Furthermore, we see a greater variance between in-domain predictions than out-of-domain predictions, indicating XAI measures something other than learned parameters. Finally, we investigate the relative importance given to expert-derived structural alerts and find similar importance given irregardless of applicability domain, randomization and varying training procedures. We therefore caution future research to validate their methods using a similar comparison to human intuition without further investigation.

Scientific contribution In this research we critically investigate XAI through test-time augmentation, contrasting previous assumptions about using expert validation and showing inconsistencies within models for identical representations. SMILES augmentation has been used to increase model accuracy, but was here adapted from the field of image test-time augmentation to be used as an independent indication of the consistency within SMILES-based molecular representation models.

Keywords ML, XAI, Robustness, Test-time augmentation, Interpretation, Representation learning, Explainability

*Correspondence:
Peter B. R. Hartog
peter.hartog@astrazeneca.com
Full list of author information is available at the end of the article

Graphical Abstract



Introduction

Explainable artificial intelligence (XAI) is used to elucidate how predictions from machine learning (ML) models are generated [1]. XAI is required to identify security and bias risks, enhance new discoveries by elucidating the reasoning behind predictions and enforce *Right of Explanation* policies [2]. Explanations are reported by assigning relative importance to the input variables. Notably, different XAI methods adopt different approximations to determine this importance in relation to the ML model [3–7]. These distinct techniques come with their individual variances, underpinned by a foundation of internal model uncertainty that forms the basis for their explanations. As these XAI methods yield outcomes with varying degrees of uncertainty [8–10], it raises concerns about potential distrust among end-users. Thus, in this study, we delve into how the internal model representations affect the stability of these explanations, notably in the field of toxicity prediction.

Historically, computational toxicity prediction relied on statistical methods, structural alerts, and quantitative structure-activity relationship (QSAR) models [11], often based on chemical descriptors or fingerprint representations. However, a recent surge in popularity revolves around natural language processing (NLP) used for text-based representation learning in molecular modeling, exemplified by the high accuracy of the transformer CNN [12] for Ames mutagenicity prediction [13]. The Ames test [14] serves as a screen for mutagenic compounds, and had important advantages when it was first introduced, including less test chemical, time, plastic and can be automated [15]. In NLP-based molecular representation for prediction of Ames, Simplified Molecular Input Line Entry System (SMILES) [16] are used to represent molecules as a text string. These SMILES are then used within an NLP-driven machine learning (ML) model, such as sequence-to-sequence (seq2seq) [17, 18]. Later state-of-the-art NLP models used augmentation strategies including masking [19, 20], which also spread to NLP models for cheminformatics [12, 21, 22]. Furthermore, the high accuracy of the transformer CNN was partly achieved with data enumeration. Comparable to vision-based ML, where images are rotated to augment the training data, researchers used multiple SMILES strings representing the same molecule [23]. This was then used to optimize internal model representations in an auto-translation task to capture intricate structural nuances [24].

When model input can be directly translated to structural parts of a molecule, explanations from XAI methods more naturally correspond to human intuition. These XAI techniques extend their importance assessment capabilities across global and local dimensions. On the global context, the evaluation centers on the role of variables in shaping the model's overall construction [25]. In the local context, the focus shifts to the influence of variables on specific predictions, a facet that aligns with our current investigation. Local prediction elucidation strategies have emerged in the literature, including approximating complex models with simpler ML counterparts [5], or leveraging game theory to pinpoint influential input variables by comparing a selection of subsets [4]. Alternatively, researchers have harnessed the model's intrinsic architecture to unveil its decision-making processes. In deep learning, integrated gradients [3] use accumulate local gradients to assess variable effects on output. Alternatively, XAI can use inherently interpretive parts of a model. The latter has proven especially influential by using the attention mechanism of transformers, exemplified by recent work from Qiang et al. [7] that utilize the transformer attention mechanism thought to be inherently interpretable [18].

Contrasting multiple input parameters that represent the same underlying ground truth is a method to assess

the stability of a model or method. This test-time data augmentation has been used in prior research conducted in domains including medical imaging, where data augmentation techniques, such as image rotation and contrast adjustment, have been used to measure model uncertainty [26, 27]. These strategies have demonstrated their efficacy in enhancing model performance, gauging model uncertainty, and increasing robustness. However, this methodology remains largely unexplored within the field of XAI concerning molecular representations, presenting an opportunity to assess the robustness of XAI methods.

In this study, we delve into the challenges presented by Langer et al. [28] who investigated that stakeholders of ML models desire XAI methods to produce human-understandable and consistent interpretations. We focus on the robustness of local XAI methods for molecular representations, specifically by comparing importance assigned to equivalent SMILES representations. Our study has broad implications for the XAI field, given the potential impact of varying importance assignments in toxicity prediction. Our contributions can be summarized as follows:

- We investigate the influence of molecular pre-training settings on the transfer learning on Ames mutagenicity.
- We assess the robustness of eight XAI methods given different representations of the same underlying molecule, using test-time augmentation.
- The robustness is assessed using different data distributions, training variations and from the perspective of model randomization.
- Additionally, test-time augmentation is investigated as a means of increasing consistency between XAI methods for the same model and interpretations of the same XAI method from different learned representations.
- Finally, we provide a global-level comparison of model attributions with expert-based structural alerts.

Background

Here, we describe the mathematics behind the transformer-based, deep learning and model agnostic methods investigated in this paper. The methods utilize some common parameters and variables including the sequence length S, embedding dimension E, number of attention heads h, and head-specific embedding dimension $d = E/h$, number of layers L. Furthermore, we define $\mathbf{x}$ as the tokenized input sequence, Ames prediction model $f(\mathbf{x}) = y, y \in \mathbb{R}^2$. XAI attributions are given as $\phi \in \mathbb{R}^{S\times E}$, where E can equal S depending on the XAI method. Because the dimensions of the embedding of ϕ can be different based on method, we average over the embedding space to make $\phi_i = \frac{1}{n}\sum_{j=0}^{n}\phi_{i,j}$, which means that $\phi_i, i \in (1, ..., S)$ is consistent between methods.

Transformer-based interpretation

Methods that depend on the transformer architecture primarily make use of the attention mechanism. We here redefine the methods proposed by Qiang et al. [7]. Vaswani et al. [18] defined the multi-head attention $\alpha_h^l \in \mathbb{R}^{h\times S\times dh}$ of layer $l \in (0, ..., L)$ (Eq. 1). Later usage of $\alpha_{i,j}$ corresponds to the values averaged over all heads. Furthermore, the output of each layer in the encoder is denoted as $o^l \in \mathbb{R}^{S\times E}$ is defined in Eq. 2.

$$\alpha_{i,j}^l = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}\right) \tag{1}$$

$$h_{i,j}^l = \text{concat}\left(\boldsymbol{\alpha}_{i,j,1}\mathbf{V}, ..., \boldsymbol{\alpha}_{i,j,h}\mathbf{V}\right) W^O \tag{2}$$

In Eqs. 1, 2 $(\mathbf{Q}, \mathbf{K}, \mathbf{V}) \in \mathbb{R}^{h\times S\times d}$ represent the projection matrices of query, key and values respectively, and $\mathbf{W}^O \in \mathbb{R}^{E\times E}$ represent model weight matrix of the projection in multi-head attention. Bias is left out in the definitions for simplicity.

Firstly, [29] proposed to use the raw attention directly as an explanation of the entire model (Eq. 3). Usually the last layer of the encoder is used in this approach. Alternatively, the raw attention by multiplying all raw attentions over all layers of the model can be aggregated to form one explanation, as defined in Eq. 4.

$$\phi_{i,AttentionMaps} = \frac{1}{n}\sum_{j=0}^{n}\alpha_{i,j}^L \tag{3}$$

$$\phi_{i,Rollout} = \frac{1}{n}\sum_{j=0}^{n}\prod_{l=0}^{L}(\alpha^l \mathbf{I}_S)_{i,j} \tag{4}$$

In Eq. 4, $\mathbf{I}_S \in \mathbb{R}^{S\times S}$ is the identity matrix.

Qiang et al. [7] further expanded research by Selvaraju et al. [30] by using the class-activated gradients. Here, class-activated gradients are denoted using $\nabla_\omega y_c$ with ω being the position with respect to which the gradients are calculated and y_c is the index of the output corresponding to the class of interest c (in our case toxic or non-toxic). This was then used in the interpretation method of the attention layers (Eq. 5). Firstly, similarly to the attention maps, by using the last layer or a specific layer to explain the entire model. Furthermore, [7] expanded this by summing over all layers and combining the gradients with the raw attention (Eq. 6).

$$\phi_{i,Grad} = \frac{1}{n}\sum_{j=0}^{n}(\nabla_{[\alpha,L]}y_c)_{i,j} \tag{5}$$

$$\phi_{i,AttGrad} = \frac{1}{n}\sum_{j=0}^{n}\left(\sum_{l=0}^{L}(\nabla_{[\alpha,l]}y_c)_{i,j} \odot \alpha_{i,j}^{l}\right) \tag{6}$$

In Eq. 6, $\odot$ represents the Hadamard product.

Finally, [7] have defined rules to use the outputs of the attention layers instead of the attention itself. Here, they combine the outputs o with the class-activated gradients with respect to o, $\nabla_{o,l}y_c$ (Eq. 7) and together with the attention (Eq. 8).

$$\phi_{i,CAT} = \frac{1}{n}\sum_{j=0}^{n}\left(\sum_{l=0}^{L}(\nabla_{h,l}y_c)_{i,j} \odot h_{i,j}^{l}\right) \tag{7}$$

$$\phi_{i,AttCAT} = \frac{1}{n}\sum_{j=0}^{n}\left(\sum_{l=0}^{L}\alpha_{i,j}^{l} \odot (\nabla_{h,l}y_c]_{i,j}) \odot h_{i,j}^{l}\right) \tag{8}$$

Deep learning-based interpretation

Methods that are dependent on deep learning, but not specifically transformers-based, usually depend on the calculation of the gradients for their interpretation. The most straightforward way is to use the basic full gradients over the entire model. Integrated gradients (IG) contrasts gradients of a prediction iteratively with the gradients of a background sample by integrating over the differences between input parameters of the sample $\mathbf{x}$ and the input parameters of the empty background $\bar{\mathbf{x}}$ (Eq. 9).

$$\phi_{i,IG} = \frac{1}{n}\sum_{j=0}^{n}\left((x_{i,j} - \bar{x}_{i,j})\int_{\alpha=0}^{1}\frac{\delta f(\bar{x} + \alpha(x - \bar{x}))}{\delta x_{i,j}}d\alpha\right) \tag{9}$$

Model agnostic interpretation

The model agnostic methods use either inherently interpretable approximations of models [5], or other methods that perturb the input and evaluate model output variation. One perturbation technique is using Shapely additive values (Eq. 10) [4] where game theory of parameter subsets are used to identify the relative importance of parameters.

$$\phi_{i,SHAP} = \sum_{\mathbf{S}\subseteq\mathbf{F}-\{k\}}\frac{|\mathbf{S}|!(|\mathbf{F}| - |\mathbf{S}| - 1)!}{|\mathbf{F}|!}\left(f_{\mathbf{S}\cup\{k\}}(x_{\mathbf{S}\cup\{k\}}) - f_{\mathbf{S}}(x_{\mathbf{S}})\right), \tag{10}$$

In Eq. 10 $\mathbf{F}$ is the set of all subsets, $\{k\}$ the variable of interest, $\mathbf{S}$ the sets of all subsets without $\{k\}$, and $x_{\mathbf{S}}$ and $x_{\mathbf{S}\cup\{k\}}$ are the input with the parameters of subset $\mathbf{S}$ and $\mathbf{S}\cup\{k\}$ respectively.

Table 1 Model training details

	Training	
Parameter	**Pre-training**	**Transfer learning**
Batch size	128	128
Learning rate	10^{-4}	5×10^{-5}
Weight decay	0.01	0.01
Dropout	0.1	0.3
Initialization	Xavier	Xavier
Optimizer	AdamW	AdamW
Scheduler	None	None
Frozen encoder	No	Yes
Max sequence length	175	175
Token space	68	68
Embedding dimension	512	512

Methods

Data collection and processing

All data was gathered from Therapeutic Data Commons (TDC) (version 0.4.0) [31] which provided standardized output for the ChEMBL database (version 29) [32, 33] and the Ames dataset [13] including standardized splitting. Both datasets were cleaned using RDKit (version 22.9.3) [34]. Datasets were processed to remove stereochemistry and salts; to correct invalid hybridization, conjugation, chirality, and valency; and to set correct chirality, aromaticity and chemical property flags. Corresponding canonical SMILES were then generated and duplicates were removed. Finally, datapoints that had overlap between Ames and ChEMBL were removed from the ChEMBL dataset. SMILES were tokenized based on the character representations of the string format and given both a beginning-of-sequence (BOS) and end-of-sequence (EOS) token together with optional padding (PAD) tokens to make sequences of equal length, according to the original paper from Karpov et al. [12]. Structural alerts were gathered from Kazius et al. [35] and generated using RDKit SMARTS (SMILES arbitrary target specification) representations. Identification of alerts in molecules was performed using RDKit substructure search.

Model architecture and training

Model training was performed using PyTorch (version 2.0.1) [36] and Lightning (version 2.0.5) [37]. Early stopping was implemented based on the validation cross entropy loss. Full model parameters for the training are described in Table 1 and a general overview of the methods used are visualized in Fig. 1.

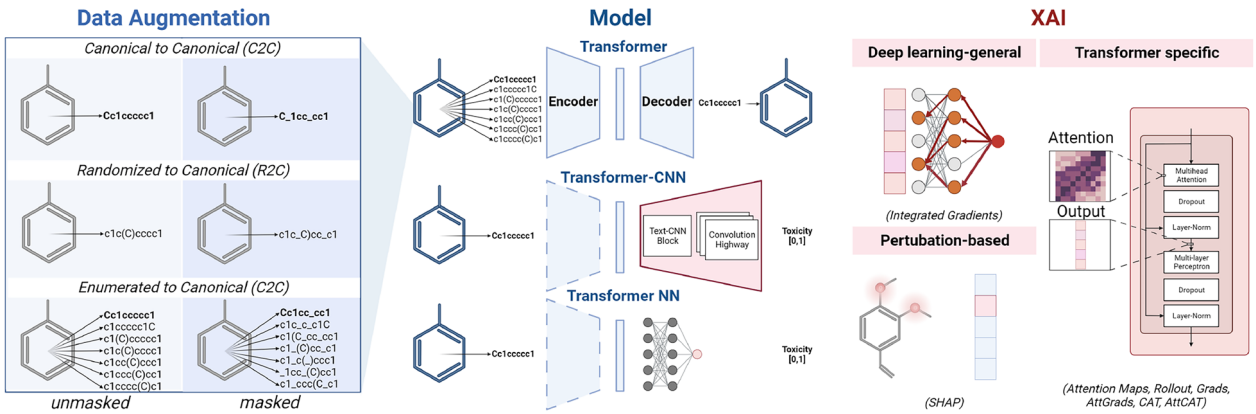


Fig. 1 Overview of the methods used throughout the research. Data augmentation is used during pre-training. Transfer learning uses the pre-trained transformer encoder together with a small neural network or CNN. Thereafter, eight XAI methods subdivided into three groups were used for interpretation

Table 2 ChEMBL data breakdown

	Size	Training	Validation	Test
ChEMBL 29	1.895.373	1.291.223	197.497	407.014
ChEMBL 29 (no Ames)	1.892.281	1.289.081	197.119	406.081

Representation learning

In this study, two transformer-based architectures were explored to learn the molecular representation of SMILES. The first is the encoder-decoder architecture that forms the basis of seq2seq [17] and BART [20] models. The second is the encoder-only architecture that formed the basis for the BERT model [19]. The sole difference between these architectures is that the decoder architecture in the former is a transformer block with cross attention and the latter is a simple multilayer perceptron.

The pre-training was based on the ChEMBL database (Table 2), where the task was auto translation. The architecture was trained to produce the canonical SMILES representation from either a canonical SMILES, randomized SMILES or an enumerated SMILES (up to ten randomized SMILES and one canonical SMILES). Additionally, masking was performed as another way to augment the training process, where 15% of the tokens in a SMILES string were replaced by 80% MASK tokens, 10% random character tokens and 10% unchanged tokens. Furthermore, alternative embedding dimensions and pruning options were also explored.

Ames transfer learning

The architectures were used for transfer learning by using the pre-trained encoders from the pre-trained models coupled with a small neural network, the predictor. Both the original implementation of the transformer CNN where the predictor was a TextCNN layer with a highway unit layer, as well as a simple max pooling layer with a multilayer perceptron were used as model variants for the transfer learning stage. Transfer learning was performed using canonical to a binary classification (toxic or non-toxic) (Table 3). During training, the pre-trained encoder were frozen to keep generalization capabilities, and trained on the scaffold split as provided by TDC. In order to give an indication of variance in performance whilst still only using the scaffold split, test-time bootstrapping was performed where the test set was sampled with replacement until full test length, and prediction statistics calculated 1000-fold to indicate variance (Tables 10, 11).

Additionally, we evaluated training variations, including transfer learning using enumerated SMILES to a

Table 3 Ames data breakdown

	Size	Scaffold split			Random split		
		Training	Validation	Test	Training	Validation	Test
Toxic		2511	373	753	2470	391	776
Non-toxic		2248	297	535	2139	297	644
Total	6717	4759	670	1288	4609	688	1420

Table 4 Descriptions and equations of metrics used during the analysis of the interpretations

Method	Implementation	Equations
IG	Captum	Eq. 9
SHAP	Captum	Eq. 10
Attention Maps	Re-implemented	Eq. 3
Rollout	Re-implemented	Eq. 4
Grads	Re-implemented	Eq. 5
AttGrads	Re-implemented	Eq. 6
CAT	Re-implemented	Eq. 7
AttCAT	Re-implemented	Eq. 8

Described are three metrics, what they measure and their equations. ϕ_i is the attribution of token *i*. Entropy and cosine similarity were calculated only using the atom components

Table 5 Names of components, description of components and example of token strings

Name	Analysed components	Example
Full	Full token string	*BOS* O=C=O *EOS PAD PAD*
SMILES	Full SMILES string	O=C=O
Atom	Atom tokens ordered canonically	COO
Alerts	Atom tokens of structural alerts	CO

Described are four names with corresponding sections of the tokens analysed in statistical analyses. Most analyses use atom tokens or alerts but are relative to the full tokenized string

binary classification (enumerated), training using a randomly initialized frozen encoder, a completely randomized model without further training and training using the random split instead of the scaffold split (Fig. 7). Details regarding data distribution differences between the data splits are found in Appendix A.

XAI methods

The XAI methods of IG and SHAP were used using captum [38]. All other methods were re-implemented. Methods are described in Table 4. For XAI methods that use a specific layer for their interpretation (Attention Maps, Grads), we used the last layer averaged over all attention heads. Furthermore, the original implementations of Grads and AttGrads, and CAT and AttCAT were re-implemented using the PyTorch [36] autograd system instead of hooks as in the original implementation [7].

Statistical analysis

All attributions were obtained based on the SMILES sequences of molecules and represented each token attribution as ϕ_i with i corresponding to each token in the original full string. Analyses of token attributions were analysed in the normalized form were the original ϕ attribution vector was divided by the absolute sum of the total string $\hat{\phi}_i = \frac{\phi_i}{||\phi||}$. We understand ϕ to correspond the attributions of the full tokenized string, including original SMILES string, BOS, EOS and PAD tokens. Other analyses include analyses of components of SMILES, atom and alerts (Table 5). As mentioned, all analysed components were first normalized with respect to the full tokenized string.

We investigated the normalized attributions by comparing distances between attributions, entropy within the attributions and the relative importance given to relative components of the distributions (Table 6). Cosine similarity was chosen as a distance measure to analyse the variation of importance over attributions using the implementation from SciPy [39]. Cosine similarity was chosen to measure the agreement of importance given to each of the tokens. Entropy was calculated as a measure of information, similar to Dabkowski and Gal [40], and relative importance is the fraction of the attribution given to a specific component of the input (Table 5), mostly to investigate the overlap between XAI information and human-derived structural alerts [35].

Computational efficiency

To avoid unnecessary computational overhead, all models were kept small 16.5 M parameters or lower, trained on the smaller ChEMBL dataset rather than the more standard PubChem dataset and using 16-bit precision. All models can be trained using a single GPU. The longest pre-training time in our experiments was 24 h (masked enumerated encoder-decoder) using two GPUs with distributed training for faster and more efficient computing.

Reproducibility

The implementation code are publicly available on GitHub at https://github.com/PeterHartog/augmented-xai to ensure the reproducibility of the experiments. Cleaned data and trained model weights are publicly available from Figshare: https://doi.org/10.6084/m9.figshare.24866091.

Results

Model training

ChEMBL representation learning

To generate our internal representations, we first trained attention-based encoder and attention-based decoder (encoder-decoder) and attention-based encoder with an MLP (encoder-only) transformer models on the ChEMBL small molecule dataset (Table 8). Training regimens were to translate canonical to canonical (C2C), randomized to canonical (R2C) and enumerated

Table 6 Descriptions and equations of metrics used during the analysis of the interpretations

Metric	Analyses	Equation
Cosine distance	Distance between attributions	$cos(\theta) = 1 - \frac{u \cdot v}{\|\|u\|\|_2 \|\|v\|\|_2}$.
Entropy	Information contained in the attribution	$H(\phi) = -\sum_{i=0}^{n} \phi_i \times log_2(\phi_i)$
Relative importance	Relative importance given to component	$\sum \phi_i, \phi_i \in$ component

Described are three metrics, what they measure and their equations. ϕ_i is the attribution of token *i*. Entropy and cosine similarity were calculated only using the atom components

Table 7 AUROC, accuracy, F1, MCC precision and recall scores of MLP models transfer learned on Ames data

	Training	AUROC↑	Accuracy↑	F1↑	MCC↑	Precision↑	Recall↑
No training	Untrained	0.652	0.516	0.143	0.063	0.081	0.619
	Native	0.676	0.634	0.624	0.269	0.607	0.642
Variations	Random split	0.856	0.788	0.788	0.576	0.789	0.787
	Train set	0.873	0.792	0.792	0.584	0.792	0.792
	CNN	0.709	0.650	0.657	0.300	0.671	0.644
	Enumerated	0.810	0.739	0.739	0.478	0.738	0.739
Encoder only	C2C	0.734	0.666	0.666	0.332	0.665	0.666
	R2C	0.738	0.670	0.670	0.339	0.670	0.670
	E2C	0.731	0.665	0.665	0.331	0.665	0.665
	MC2C	0.754	0.682	0.682	0.364	0.683	0.682
	MR2C	0.694	0.653	0.652	0.305	0.651	0.653
	ME2C	0.804	0.719	0.719	0.438	0.719	0.719
Encoder-decoder	C2C	0.716	0.662	0.662	0.324	0.663	0.662
	R2C	0.698	0.634	0.634	0.269	0.634	0.634
	E2C	0.748	0.677	0.678	0.354	0.679	0.676
	MC2C	0.751	0.682	0.682	0.365	0.681	0.683
	MR2C	0.698	0.647	0.647	0.294	0.647	0.647
	ME2C	0.772	0.696	0.697	0.393	0.699	0.695

Values are based on the scaffold split

to canonical (E2C) as well as the masked versions (e.g., ME2C). Additionally, the encoder-decoder models character and sequence accuracy scores for the canonical to canonical versions were high in all versions except for the R2C models. The encoder-decoder R2C models were more predictive regarding the sequence accuracy without context (greedy-search), but still significantly worse than other training regimens.

Ames transfer learning

To create our final prediction models, we used transfer learning of the pre-trained models to the Ames training set by freezing the encoders and replacing the decoders by either a textCNN or MLP. The MLP results on the scaffold split (Table 7) outperformed the transformerCNN model (Table 9). Because of this, we decided to continue our interpretation analysis with the MLP model. Pre-trained models of C2C and E2C outperformed R2C models, whilst masked models increased model statistics over unmasked models. Finally, encoder-only pre-trained models generally outperformed encoder-decoder models on the Ames transfer learning task.

Interpretation analysis

The robustness of XAI methods was examined from three different perspectives. The first was to analyse XAI methods in the same circumstance, namely the same model and the same input, to see how different the explanation given is over all methods. Figure 2a shows the different attributions from each XAI method for one randomly chosen canonical SMILES of the encoder-decoder ME2C model. Secondly, we want to investigate the variance that internal representation gives. To further illustrate this, the heat map shows the inner variance of the method, and the robustness score is the mean value of this heat map. Figure 2b shows how the different training regimens for an encoder-decoder architecture influence the explanations of IG for on randomly chosen canonical SMILES. Again, the heat map shows the internal consistency for

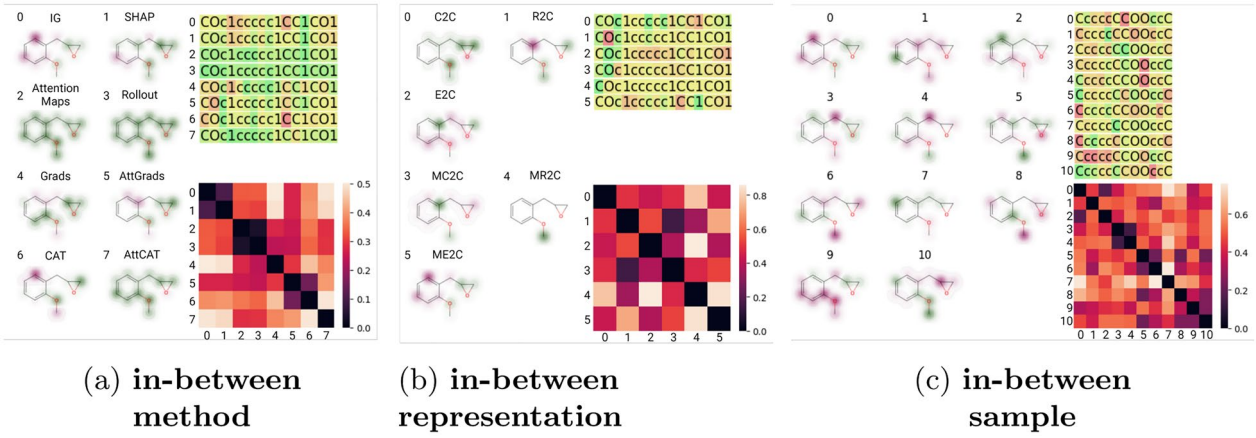


Fig. 2 Single instance robustness analysis of XAI methods from three different angles: in-between method, in-between representation and in-between input. **a** Importance given to each input parameter varied over XAI methods, given a constant canonical SMILES input and encoder-decoder masked enumerated to canonical (ME2C) model. **b** Importance given to each input parameter varied over representations of the encoder-decoder model, given a constant canonical SMILES input and IG XAI method. **c** Importance given to each input parameter varied over different SMILES representations, given a constant canonical SMILES input, IG XAI method and encoder-decoder masked enumerated to canonical (ME2C) model

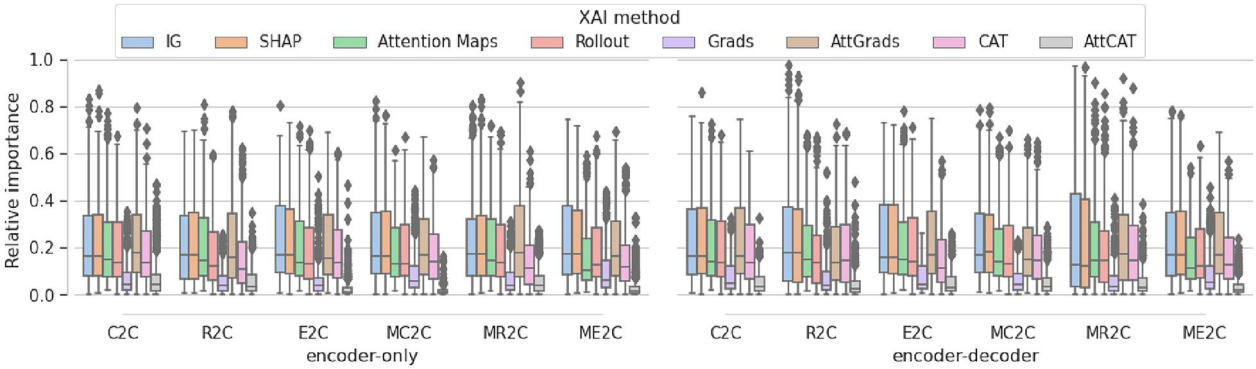


Fig. 3 In-between sample distance robustness analysis of XAI methods. Cosine distances of attributions given to different SMILES representations of the same molecule over different XAI methods with different pre-trained representation models

in-between model robustness. Finally, we investigate how input variation all representing the same underlying ground truth, can change the given importance of IG for the encoder-decoder ME2C model. In Fig. 2c, we investigate the canonical SMILES and ten randomized SMILES and compare the internal robustness between the importance given to the atom indices. Importantly, we reshuffle the respective SMILES strings according to RDKit canonical atom order so that each character represents the same atom when compared.

Test-time augmentation as a measure for robustness

Firstly, we analyse the difference between SMILES of the same molecules for each method and model training (Fig. 3). Overall, the cosine distance between these samples is greatest in the IG, SHAP and AttGrads and AttCat. Additionally, the distances of the attention maps, rollout attention are lowest and most variable. The Grads value seem to have the highest variability in most models. All other methods are relatively similar in all models, with the exception of the R2C and MR2C pre-trained models, where all methods have increased cosine distances in the encoder-decoder models, but less pronounced differences in the encoder-only models.

Additionally, we investigate whether the distances can be explained by the difference between canonical and random SMILES. No significant differences are found between distances of canonical and a randomly chosen random SMILES in most models. Some exceptions were found, as determined by the Mann Whitney U test (Table 12): CAT and AttCAT for encoder-only ME2C; encoder-decoder C2C AttGrads; encoder-decoder R2C Rollout and AttGrads; encoder-decoder E2C Attention Maps; encoder-decoder MC2C IG and SHAP. In

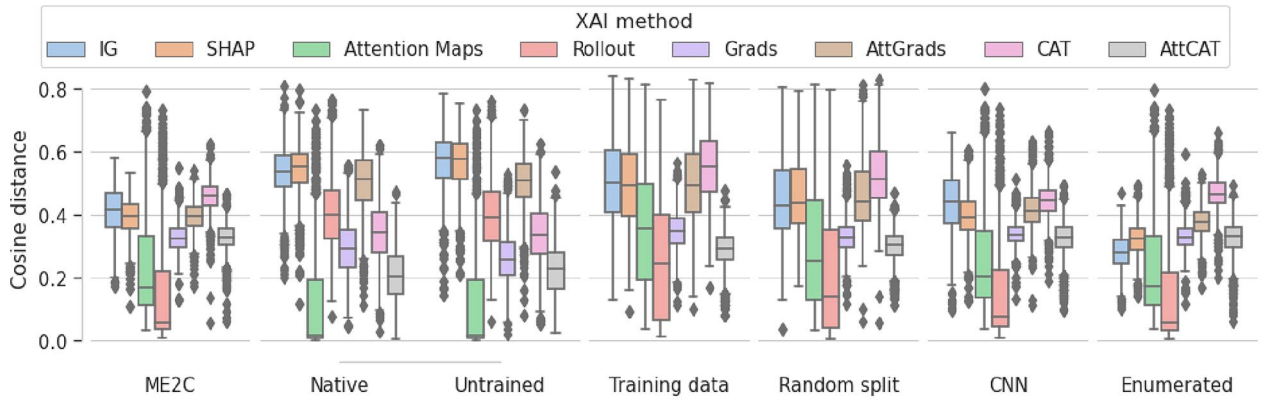


Fig. 4 In-between sample cosine distance of different training settings. Cosine distances of different attributions given to different SMILES representations per molecule of different XAI methods of different training settings. Training settings include baseline ME2C of the encoder-decoder pre-training architecture, untrained, frozen encoder (native), completely random model (untrained), distances of the training data (train) of baseline, distances of the test set on a random split model (random), enumerated training (enumerated) and the statistics of the CNN model (CNN). All variations had the ME2C encoder-decoder model as the initial encoder with the exception of native and untrained

these models the populations do look similar (Fig. 8), but results of these method-model combinations could possibly be explained by randomization attribution differences.

Finally, we utilize entropy as a measure for information to examine if the in-between sample analysis was due to overall small differences between values (Fig. 9). Entropy scores of IG, SHAP, Attention Maps, Rollout attention, AttGrad and CAT were similarly high throughout all models, but the values for Grads and AttCAT were significantly lower, indicating that the test-time augmentation robustness is partially dependent on the entropy the method itself.

Influence of ML model on XAI methods

To investigate the dependence of XAI on models, we investigate the difference of both our test-time augmentation robustness score and the overall entropy of the XAI methods between different models. Firstly, differences between the pre-trained model with AUROC of 0.798, native model with AUROC of 0.763 and untrained model with AUROC of 0.474 in both entropy and in-between sample variation are minimal (Fig. 4). This indicates that the in-between model analysis of these models is dependent on something other than learned parameters.

Secondly, we identify the difference between the in-domain training data and the out-of-domain test data, as well as in-domain test data from a model trained using a random split ((Fig. 4). The in-domain data shows a larger difference in in-between sample distance, indicating that the XAI methods depend more on learned parameters than the out-of-domain samples. It also indicates that larger cosine distances are not necessarily less consistent with model predictions. Finally, we also analyse if different training techniques (enumerated) or architecture (CNN) affects cosine distances. No obvious distance changes were found (Fig. 4).

Interestingly, investigations into differences in entropy (Fig. 10) based on these variations were minor. We also further investigated if the number of tokens explained the consistency in in-between sample cosine distances (Fig. 11), where it seemed to be carbon-dependent and explains the variation.

Using test-time augmentation to improve XAI robustness

To further assess the use of test-time augmentation, we analysed the in-between model and in-between method distances (Fig. 5). Figure 5 indicates that both the in-between model distance and the in-between method distances are reduced when values are averaged using test-time augmentation. This means that values become more consistent when using the average over test-time augmentation between both methods and models. We further investigated whether this effect was because of consistency or overall reduction in information by investigating the entropy of averaged values and canonical values (Fig. 10). There was no difference found between averaged values or canonical values, indicating only consistency between methods change, not specific attributions.

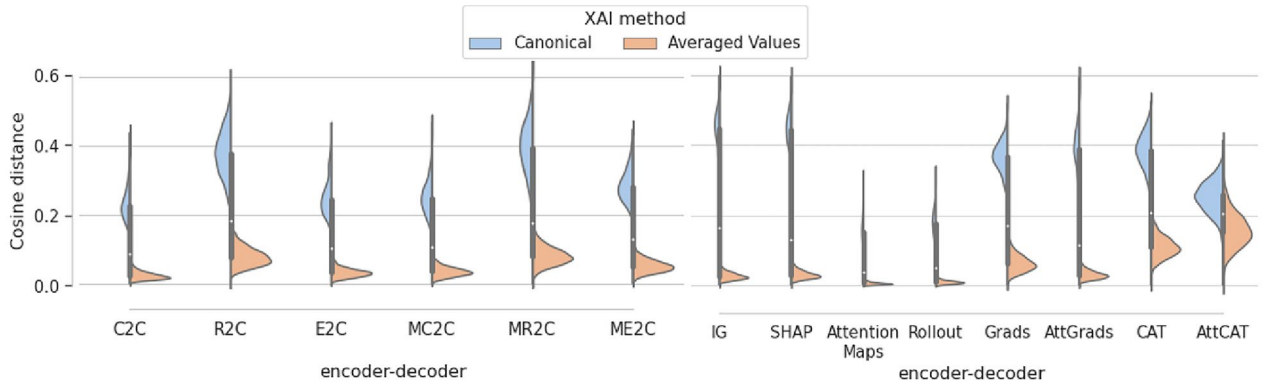


Fig. 5 Comparison of the canonical and averaged values of in-between model and in-between method cosine distances. Cosine distances between different encoder-decoder models for the same method (in-between model) and different methods for the same encoder-decoder model (in-between method) of canonical and averaged atom attributions

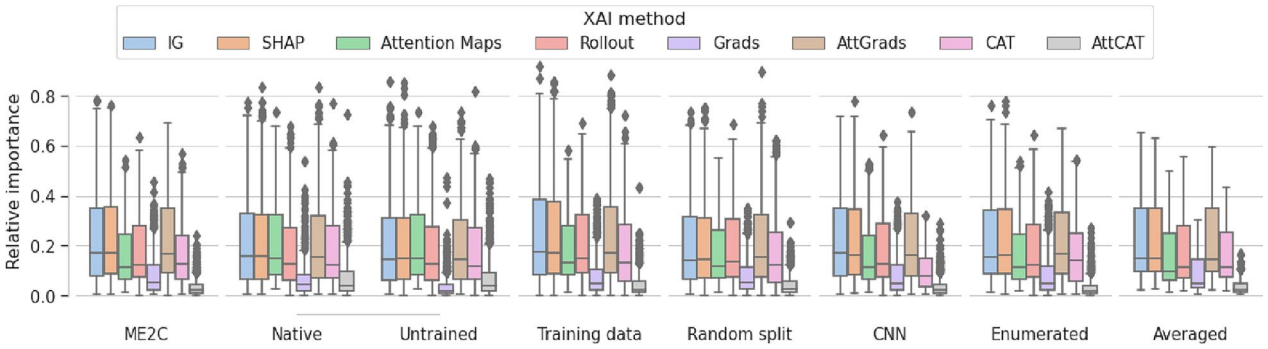


Fig. 6 Relative importance given to expert-derived structural alerts. Relative token importance given to atoms corresponding to expert-derived structural alerts. Relative importance is given for canonical representations of different XAI methods of different training settings. Training settings include untrained, frozen encoder (native), completely random model (untrained), distances of the training data (train), distances of the test set on a random split model (random), enumerated training (enumerated), the statistics of the CNN model (CNN) and the averaged values of all SMILES enumerations of the ME2C model

Comparison with expert-based structural alerts

Finally, we analyse the robustness of model explanation by examining the overall attribution of all tokens as opposed to tokens corresponding to expert-derived structural alerts (Fig. 6). Values of IG, SHAP and AttGrads all consistently had the highest values with mean values around 0.2, whereas a Attention Maps, Rollout and CAT gave consistently similar or slightly lower relative importance. Relative importance of Grads and AttCAT consistently gave the lowest relative attribution to the structural alerts. Relative importance is generally consistent between randomized models, in-domain samples, training variations and test-time augmentation averaging. Similar consistent results were observed in model variations (Fig. 12).

We further analysed the overall attribution of all tokens as opposed to the SMILES tokens, atom tokens and tokens corresponding to expert-derived structural alerts of all methods for each model (Fig. 14) and all methods for each variation (Fig. 13), where similar attributions were found irregardless of model or variation.

Discussion

In this research, we performed an analysis of eight XAI methods and used (1) test-time augmentation, (2) in-between model, method and sample cosine distances, (3) entropy, and (4) relative importance given to structural alerts to assess the validity of these XAI methods and XAI robustness analyses in the context of NLP-based molecular-representation models for toxicity.

The importance of tokenization in NLP-based research

Notably, we show that overall cosine distances between samples of a trained model and progressively randomly initiated models is minimal. This indicates that model interpretation in these cases is dependent on an aspect

other than the learned model parameters. We analyse the variation with respect to the amount of carbon atoms in Fig. 11, which shows that consistency is dependent on the amount of carbon atoms. This indicates that tokenization is what is measured mostly by these interpretation methods. In fact, we find this aligns with the previous research of Zafar et al. [41] who found that interpretation of untrained models is not random in NLP-based models.

We hypothesise that the questions about the intuitive attributions of random models posed by Zafar et al. [41] can be answered by assessing the influence of the initial tokenization or architecture artifacts. This is further supported by our findings that a native encoder sometimes even outperformed pre-trained encoders during transfer learning, as well as research from Ucak et al. [42] who showed that different tokenizations can have significant effects on final performance in NLP-based molecular representation models.

Usage of human intuition to validate XAI

In this study, we also used expert-derived structural alerts for Ames mutagenicity to investigate if the XAI overlapped with human intuition. A recent study investigated the mean attention weights given to toxicophores in Tox21 prediction and found significantly higher attention weights to toxicophore atoms than non-toxicophore atoms [43]. In this study, we analyse the overall importance given to structural alerts and don't divide by the number of atoms. Regardless, in this study we also show that the relative importance given to structural alerts is mostly model-independent and, crucially, no different from completely randomized models. Structural alerts and case studies are often used to investigate the inner workings of ML research, but this finding urges caution using these tactics without further investigation.

Test-time augmentation to improve XAI

Recently, [44] independently analysed the invariance and equivariance of interpretability methods. They used the bag-of-words [45] NLP model in combination with text permutation to assess the robustness of a number of feature importance methods, including IG and gradient SHAP [46]. Bag-of-words models differ from our models in that text data in bag-of-words is inherently invariant to text permutation, whereas different text representations in our approach are learned to be invariant given the same underlying molecule. Crabbé and van der Schaar postulated that "...Any interpretability method can be made invariant..., one should increase the number of samples N_{inv} until the desired invariance is achieved. In this way, the method is made robust without increasing the number of calls more than necessary" [44]. This is consistent with our findings using test-time augmentation to improve XAI robustness, where we see greater in-between model and in-between method consistency when using values averaged over multiple N_{inv} samples (i.e., test-time augmentations). This was even true in a model that was not inherently invariant.

However, we also identified that neither the amount of information analysed through entropy, nor the relative importance given to expert-derived structural alerts changes when using averaged values. This indicates that test-time augmentation can be used to make XAI more invariant, but not to improve XAI attribution overall.

XAI methods and test-time augmentation for out-of-domain identification

Wang et al. [26] analysed test-time augmentation as a measure of aleatoric (data-based) uncertainty in the task of image segmentation and found it improved over baseline methods. Later, [27] used test-time augmentation to measure epistemic (model-based) uncertainty in the field of image classification where it again found to improve uncertainty classification. Uncertainty can be useful to identify out-of-domain predictions, but, to our knowledge, test-time augmentation has not yet been applied to that area of uncertainty prediction.

In this research, we hypothesize that test-time augmentation has a difference between in-domain and out-of-domain predictions. It remains to be seen if these differences can be used to determine out-of-domain predictions, especially in the field of NLP-based molecular representations. This is because we find less variation in out-of-domain in-between sample interpretations than the variation of in-domain interpretations. This indicates that if the epistemic uncertainty is measures the same model effects as XAI methods described here, it will show decreased values in uncertainty for out-of-domain samples. Additionally, this result identifies the need to combine XAI explanations with applicability domain assessments, to verify the explanation.

Test-time augmentation as a measure for robustness

In this study, we analysed test-time augmentation as a measure for XAI robustness. We show substantial disagreement between augmented SMILES, even when canonical SMILES show no more difference than randomized SMILES. However, we also note a number of cautionary findings, including similar disagreements in randomized models, higher disagreement in in-domain distributions and relatively consistent distribution rankings. We did observe higher variation in in-between sample distances of untrained and worse-performing models (R2C and MR2C). This finding was subsequently diminished by the finding that in-domain training set values and random split values had similarly increased variation

and higher overall values of in-between sample distances. We therefore conclude that using test-time augmentation as a measure for XAI robustness is inherently valid, but requires a comparison to randomized and applicability domain to draw well-founded conclusions.

XAI implementation differences

Overall, we have investigated eight XAI methods, of which six are transformer-specific, one more general to neural networks and one perturbation XAI method. Firstly, we note that all methods from Qiang et al. [7] were re-implemented, and crucially, the methods of Grads, AttGrads, CAT and AttCAT were re-implemented with changes to the original implementation. Namely, for the gradients of the attention ($\nabla\alpha_{i,j}$) and the attention output ($\nabla h_{i,j}$), we implemented the gradients with respect to the full outcome. We believe this to be in line with the original publication, but have not tested differences due to implementation difficulties. Due to this, investigations into Grads, AttGrads, CAT and AttCAT were can be subject to change based on implementation. We did find the methods of Grads and AttCAT to have significantly lower entropy and lower relative importance to the structural alerts. However, the difference with respect to gradient calculations should not impact our final conclusions, as the methods still use the same underlying principles.

Furthermore, the methods of attention maps and Grads were performed using the last layer and averaged over all attention heads. Some researchers have suggested to use specific layer and head combinations to explain the model based on heuristic approaches, such as Schwaller et al. [47]. We leave such analyses to future papers and stay consistent here with the methods from Qiang et al. [7]. The methods of IG and SHAP were implemented using the standard Captum [38] library and can therefore serve as proper baseline implementations.

Future investigation of XAI methods for SMILES-based representation models

In general, our findings indicate a greater need to identify what XAI methods measure, and specifically to remove any confounding background information. In our case, all methods in this context seemed to rely not on learned gradients, but likely instead on the tokenization. While this could be the ground truth explanation, randomization experiments indicated that this is not the case. We therefore advocate for approaches that properly take background into account. Notably, local XAI methods are often further refined through comparative analysis, contrasting their findings with those obtained from background or empty samples. Two of our methods included such a comparison to a background sample, namely SHAP and IG. The approach contrasted the input with padding tokens. This did increase their in-between sample distances in randomization studies, but this effect did not translate to the in-domain predictions, which saw similar distances albeit with higher variation. Interestingly, these methods were the only ones affected in the training variations of enumerated training on Ames and the CNN architecture, where other methods stayed consistent.

However, further comparisons, specifically with regards to randomization should help improve robustness in XAI methods. Although, as [8] discussed, randomization can indicate architecture-based priors of XAI, XAI should reflect learned parameters to explain model behaviour. Methods to increase XAI robustness to randomized models can include but are not limited to creating models to investigate ML models, such as [10], contrasting findings to randomized models predictions as background instead of empty samples, and counterfactual explanations, such as the recent study from Fradkin et al. [48].

Finally, we hypothesize that test-time augmentation can improve XAI methods and identify robust XAI methods, but only when XAI methods are both in-domain and measure decision-dependent parameters, not confounding information.

Conclusion

In this research, we performed an analysis of eight XAI methods and used several analyses to assess the validity of these XAI methods and XAI robustness in the context of NLP-based molecular-representation models for toxicity. We report significant differences between explanations for different representations of the same ground truth. Additionally, we show that randomized models are similarly different, indicating that the XAI methods applied to NLP-based molecular representations in this and past research reflect tokenization more than learned parameters. Interestingly, we see a greater variance between in-domain predictions than out-of-domain predictions, further supporting this hypothesis. Furthermore, we investigated the relative importance given to expert-derived structural alerts and find similar importance given irregardless of applicability domain, randomization and training variation. We therefore caution future research to validate their methods using a similar comparison to human intuition without further investigation into the validity and robustness of the XAI method used. Finally, we note that test-time augmentation can be used as a measure of robustness, only if used in conjunction with other XAI method analyses, and note a greater need to identify what XAI precisely measure before drawing conclusions.

Appendix A
Ames data analysis
See Fig. 7.

Appendix B
Model statistics
See Tables 8, 9.

Appendix C
Bootstrapping statistics
See Tables 10, 11

Appendix D
Differences between canonical and randomized smiles distances
See Fig. 8 and Table 12

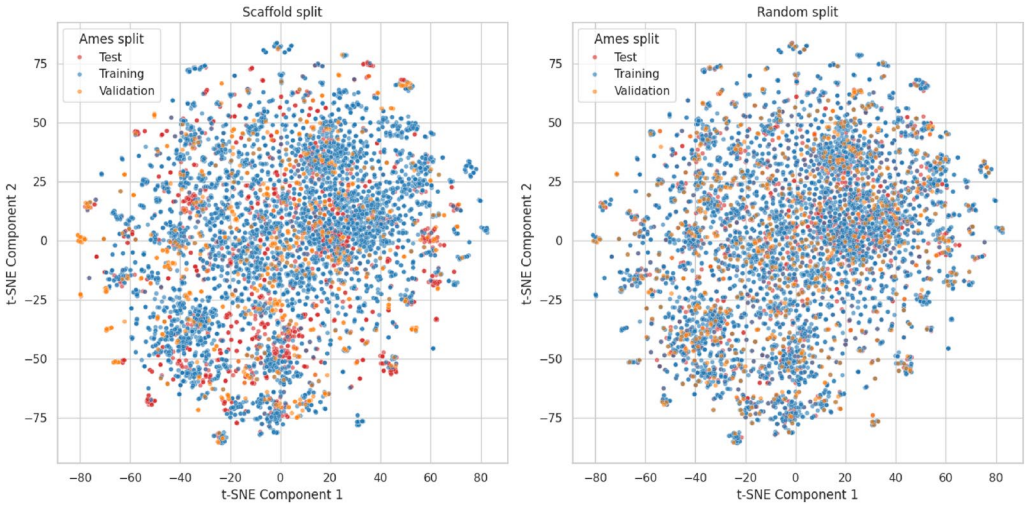


Fig. 7 t-SNE plots of the distributions of training, validation and test for different splits of the Ames data set

Appendix E
Entropy analyses

Table 8 Model Statistics of pretrained transformer models

Architecture	Training	Canonical				Enumerated				Time (hh:mm)	Size
		Character accuracy		Sequence accuracy		Character accuracy		Sequence accuracy			
Encoder only	C2C	**1.000**		**1.000**		**1.000**		**1.000**		01:16	6.7M
	R2C	**0.202**		**0.000**		**0.195**		**0.000**		00:42	6.7M
	E2C	**1.000**		**1.000**		**1.000**		**1.000**		03:33	6.7M
	MC2C	**0.999**		**0.993**		**0.994**		**0.819**		01:25	6.7M
	MR2C	**0.202**		**0.000**		**0.196**		**0.000**		00:46	6.7M
	ME2C	**1.000**		**0.991**		**0.999**		**0.962**		10:31	6.7M
Encoder-decoder	C2C	0.998	**0.995**	0.994	**0.994**	0.998	**0.996**	0.994	**0.994**	01:45	15.9M
	R2C	0.813	**0.089**	0.000	**0.000**	0.554	**0.082**	0.000	**0.000**	02:44	15.9M
	E2C	0.998	**0.995**	0.995	**0.995**	0.998	**0.995**	0.994	**0.994**	08:43	15.9M
	MC2C	0.998	**0.996**	0.993	**0.987**	0.976	**0.965**	0.463	**0.452**	02:53	15.9M
	MR2C	0.812	**0.040**	0.000	**0.000**	0.549	**0.040**	0.000	**0.000**	05:56	15.9M
	ME2C	0.998	**0.994**	0.983	**0.958**	0.998	**0.994**	0.979	**0.936**	24:04	15.9M

Bold values are accuracies calculated without previous token information. Models were tested on canonical SMILES to canonical SMILES (canonical) and enumerated SMILES to canonical SMILES (enumerated)

Table 9 AUROC, accuracy, F1, MCC precision and recall scores of the TransformerCNN models transfer learned on Ames data

Architecture	Training	AUROC↑	Accuracy↑	F1↑	MCC↑	Precision↑	Recall↑
No training	Untrained	0.564	0.507	0.668	0.055	0.991	0.504
	Native	0.698	0.653	0.665	0.308	0.689	0.643
Encoder only	C2C	0.687	0.639	0.638	0.279	0.635	0.641
	R2C	0.706	0.643	0.652	0.287	0.668	0.636
	E2C	0.685	0.644	0.656	0.288	0.679	0.634
	MC2C	0.711	0.656	0.658	0.311	0.663	0.653
	MR2C	0.693	0.641	0.647	0.283	0.656	0.637
	ME2C	0.738	0.676	0.681	0.352	0.692	0.670
Encoder-decoder	C2C	0.711	0.663	0.662	0.327	0.658	0.665
	R2C	0.680	0.625	0.623	0.249	0.621	0.626
	E2C	0.719	0.650	0.650	0.301	0.649	0.651
	MC2C	0.715	0.643	0.647	0.287	0.655	0.640
	MR2C	0.652	0.583	0.583	0.167	0.582	0.583
	ME2C	0.709	0.650	0.657	0.300	0.671	0.644

Values are based on the scaffold split

See Figs. 9, 10

Table 10 AUROC, accuracy, F1, MCC precision and recall scores with bootstrap variability of MLP models transfer learned on Ames data

Architecture	Training	AUROC↑	Accuracy↑	F1↑	MCC↑	Precision↑	Recall↑
No training	Untrained	0.640 ± 0.004	0.518 ± 0.003	0.172 ± 0.008	0.066 ± 0.011	0.100 ± 0.005	0.610 ± 0.019
	Native	0.652 ± 0.005	0.611 ± 0.006	0.601 ± 0.006	0.221 ± 0.012	0.587 ± 0.006	0.616 ± 0.006
Variations	Random split	0.853 ± 0.004	0.776 ± 0.006	0.776 ± 0.006	0.551 ± 0.012	0.776 ± 0.006	0.775 ± 0.006
	Training set	0.872 ± 0.002	0.789 ± 0.003	0.789 ± 0.003	0.577 ± 0.007	0.789 ± 0.003	0.788 ± 0.003
	CNN	0.721 ± 0.006	0.663 ± 0.008	0.668 ± 0.008	0.326 ± 0.016	0.678 ± 0.008	0.658 ± 0.008
	Enumerated	0.812 ± 0.003	0.740 ± 0.005	0.740 ± 0.005	0.480 ± 0.011	0.739 ± 0.005	0.741 ± 0.005
Encoder only	C2C	0.735 ± 0.004	0.666 ± 0.006	0.665 ± 0.006	0.332 ± 0.011	0.665 ± 0.006	0.666 ± 0.006
	R2C	0.735 ± 0.004	0.672 ± 0.006	0.672 ± 0.005	0.344 ± 0.011	0.672 ± 0.006	0.672 ± 0.006
	E2C	0.733 ± 0.004	0.657 ± 0.005	0.657 ± 0.005	0.315 ± 0.011	0.657 ± 0.005	0.658 ± 0.005
	MC2C	0.764 ± 0.004	0.695 ± 0.006	0.696 ± 0.006	0.389 ± 0.012	0.698 ± 0.006	0.693 ± 0.006
	MR2C	0.694 ± 0.003	0.648 ± 0.004	0.647 ± 0.004	0.296 ± 0.009	0.644 ± 0.004	0.649 ± 0.004
	ME2C	0.808 ± 0.003	0.734 ± 0.006	0.734 ± 0.006	0.468 ± 0.012	0.733 ± 0.006	0.735 ± 0.006
Encoder-decoder	C2C	0.715 ± 0.002	0.660 ± 0.005	0.661 ± 0.005	0.321 ± 0.009	0.663 ± 0.005	0.660 ± 0.005
	R2C	0.699 ± 0.005	0.640 ± 0.007	0.640 ± 0.007	0.281 ± 0.013	0.640 ± 0.007	0.640 ± 0.007
	E2C	0.748 ± 0.003	0.677 ± 0.006	0.678 ± 0.005	0.355 ± 0.011	0.680 ± 0.006	0.677 ± 0.006
	MC2C	0.748 ± 0.004	0.683 ± 0.006	0.682 ± 0.006	0.366 ± 0.012	0.681 ± 0.006	0.684 ± 0.006
	MR2C	0.687 ± 0.006	0.633 ± 0.007	0.633 ± 0.007	0.267 ± 0.013	0.633 ± 0.007	0.634 ± 0.007
	ME2C	0.782 ± 0.005	0.710 ± 0.007	0.710 ± 0.007	0.420 ± 0.014	0.711 ± 0.007	0.709 ± 0.007

Values are based on the scaffold split. ± values have been determined using 1000 fold test-time bootstrapping

Table 11 AUROC, accuracy, F1, MCC precision and recall scores with bootstrap variability of the TransformerCNN models transfer learned on Ames data

Architecture	Training	AUROC↑	Accuracy↑	F1↑	MCC↑	Precision↑	Recall↑
No training	Untrained	0.565 ± 0.006	0.502 ± 0.002	0.665 ± 0.001	0.019 ± 0.016	0.990 ± 0.002	0.501 ± 0.001
	Native	0.702 ± 0.005	0.659 ± 0.006	0.669 ± 0.006	0.318 ± 0.012	0.689 ± 0.007	0.650 ± 0.006
Encoder only	C2C	0.697 ± 0.005	0.654 ± 0.007	0.654 ± 0.007	0.309 ± 0.013	0.652 ± 0.007	0.655 ± 0.007
	R2C	0.714 ± 0.005	0.652 ± 0.007	0.660 ± 0.007	0.304 ± 0.013	0.676 ± 0.007	0.645 ± 0.006
	E2C	0.692 ± 0.005	0.649 ± 0.007	0.660 ± 0.007	0.298 ± 0.014	0.683 ± 0.007	0.639 ± 0.007
	MC2C	0.725 ± 0.006	0.660 ± 0.008	0.663 ± 0.008	0.320 ± 0.015	0.668 ± 0.008	0.657 ± 0.008
	MR2C	0.695 ± 0.005	0.643 ± 0.007	0.649 ± 0.007	0.287 ± 0.014	0.660 ± 0.007	0.638 ± 0.007
	ME2C	0.740 ± 0.006	0.680 ± 0.008	0.686 ± 0.007	0.360 ± 0.015	0.701 ± 0.008	0.673 ± 0.007
Encoder-decoder	C2C	0.711 ± 0.004	0.659 ± 0.006	0.657 ± 0.006	0.319 ± 0.012	0.652 ± 0.007	0.662 ± 0.006
	R2C	0.680 ± 0.006	0.634 ± 0.007	0.633 ± 0.008	0.269 ± 0.015	0.631 ± 0.008	0.635 ± 0.008
	E2C	0.713 ± 0.004	0.653 ± 0.006	0.652 ± 0.006	0.306 ± 0.012	0.649 ± 0.006	0.655 ± 0.006
	MC2C	0.726 ± 0.006	0.663 ± 0.008	0.667 ± 0.007	0.326 ± 0.015	0.676 ± 0.008	0.659 ± 0.007
	MR2C	0.644 ± 0.004	0.584 ± 0.001	0.583 ± 0.001	0.167 ± 0.001	0.583 ± 0.001	0.584 ± 0.001
	ME2C	0.721 ± 0.006	0.663 ± 0.008	0.668 ± 0.008	0.326 ± 0.016	0.678 ± 0.008	0.658 ± 0.008

Values are based on the scaffold split. ± values have been determined using 1000 fold test-time bootstrapping

Appendix F
Carbon breakdown

See Fig. 11

Appendix G

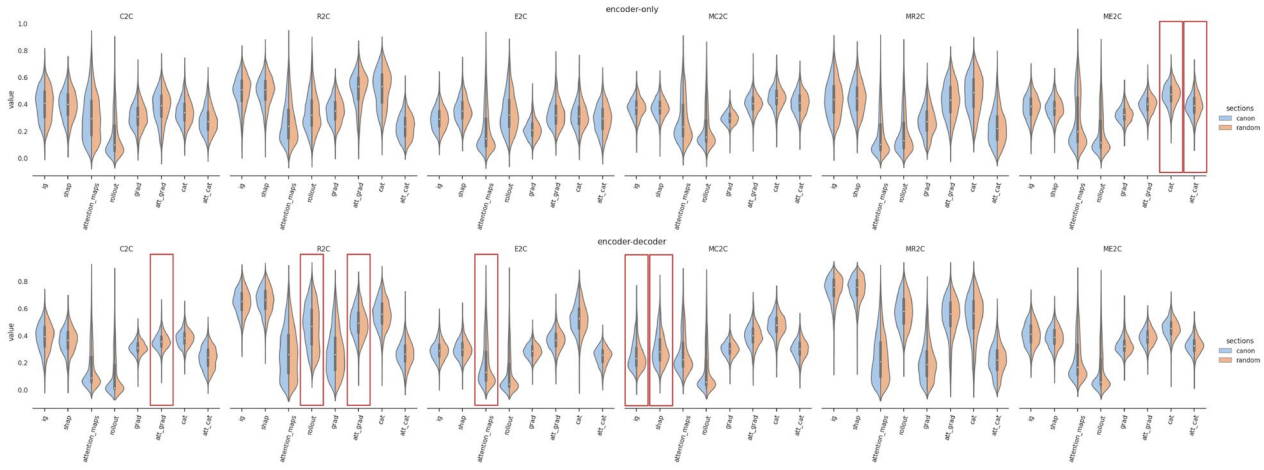


Fig. 8 Violin plot of in-sample cosine distance populations per model and XAI method. Red boxes indicate p > 0.005 and notes that the populations are not significantly similar

Table 12 Mann Whitney U test statistics of the difference in in-sample distance populations for each model and interpretation method

Architecture	Training	IG	SHAP	Att. Maps	Rollout	Grads	AttGrads	CAT	AttCAT
Encoder only	C2C	0.761	0.867	0.767	0.722	0.831	0.032	0.930	0.673
	R2C	0.121	0.114	0.507	0.339	0.316	0.036	0.376	0.469
	E2C	0.375	0.796	0.871	0.299	0.667	0.908	0.900	0.081
	MC2C	0.848	0.978	0.370	0.548	0.281	0.342	0.921	0.409
	MR2C	0.711	0.760	0.925	0.793	0.903	0.278	0.457	0.855
	ME2C	0.216	0.373	0.626	0.955	0.326	0.060	**0.004**	**0.000**
Encoder-decoder	C2C	0.414	0.202	0.794	0.794	0.537	**0.001**	0.060	0.261
	R2C	0.563	0.324	0.765	**0.000**	0.657	**0.000**	0.237	0.376
	E2C	0.066	0.348	**0.000**	0.046	0.422	0.167	0.814	0.536
	MC2C	**0.004**	**0.002**	0.651	0.932	0.447	0.521	0.625	0.362
	MR2C	0.055	0.398	0.960	0.861	0.668	0.798	0.154	0.356
	ME2C	0.590	0.381	0.955	0.971	0.498	0.186	0.377	0.376

Values are compared between canonical SMILES representation attributions or random SMILES representation as compared to all other enumerated values. Bold values are p > 0.005

Relative importance

See Figs. 12, 13, 14

Acknowledgements
The author would like to thank the doctoral candidates and supervisors from

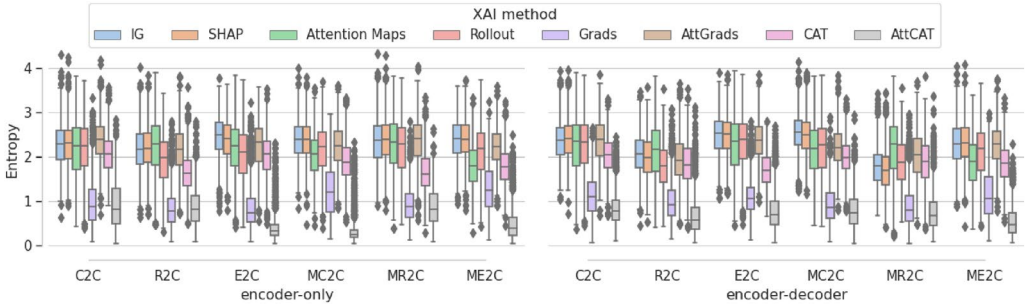


Fig. 9 Entropy values of each XAI method per model. Entropy values of XAI methods of canonical representations indicating relative information in the attributions

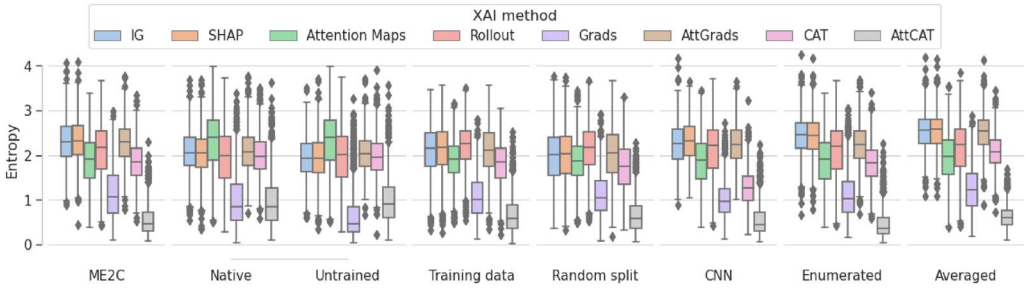


Fig. 10 Entropy values of each XAI method per experiment. Entropy values of XAI methods of canonical representations or indicating relative information in the attributions. Averaged experiment variation used the average over all enumerations instead of the canonical representation for it's entropy analysis

the Marie Skłodowska-Curie Innovative AIDD consortium for their support. In particular, the authors thank Paula Torren Peraire for the beamsearch implementation, and Emma Svensson for review and feedback for the manuscript Alessandro Tibo for troubleshooting. Additionally, the authors thank BioRender for the publication of the table of contents figure generated with BioRender authorized under the subscription plan of Peter Hartog.

Author contributions

Conceptualization, P.H., I.T., S.G.; methodology, P.H., I.T.; validation, P.H., F.K.; writing original draft preparation, P.H.; writing, review and editing, P.H., F.K., E.S., I.T.;

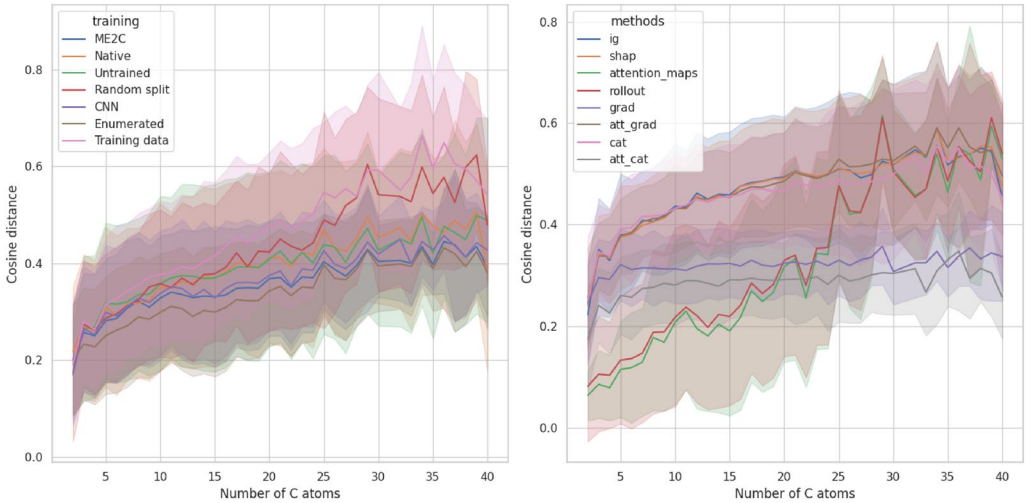


Fig. 11 Cosine distance as a measure of number of carbon atoms. Variations of in-between sample cosine distances and carbon atoms, broken down to show training variations and XAI methods

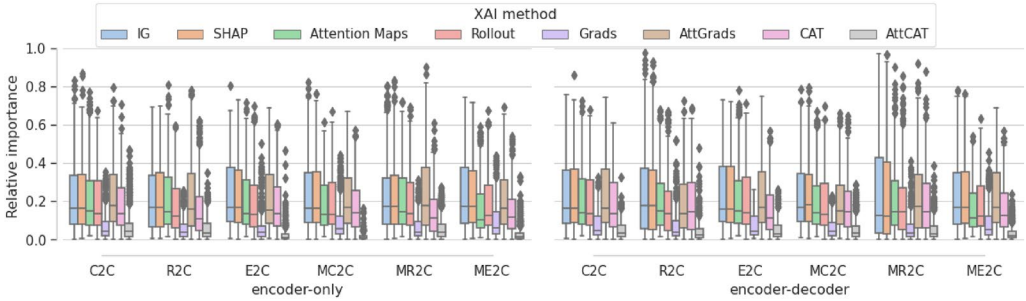


Fig. 12 Relative importance given to expert-derived structural alerts for models. Relative token importance given to atoms corresponding to expert-derived structural alerts. Relative importance is given for canonical representations of different XAI methods of different pre-trained representation models

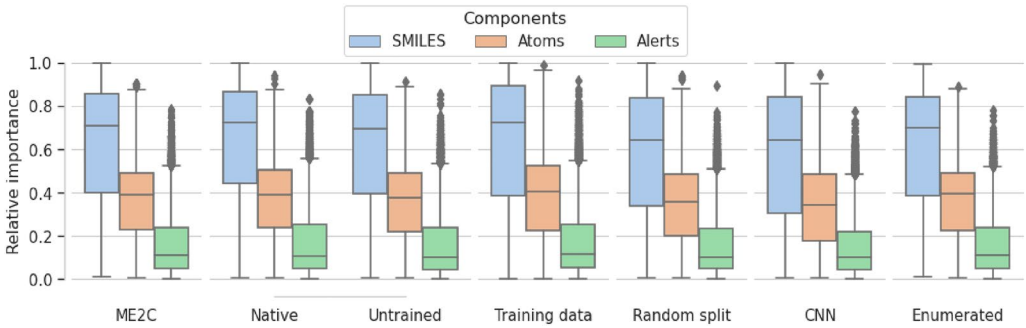


Fig. 13 Relative importance given to different components for experiments. Relative token importance given to smiles, atoms and atoms corresponding to expert-derived structural alerts. Relative importance is given for canonical representations and aggregated over all XAI methods of different training settings

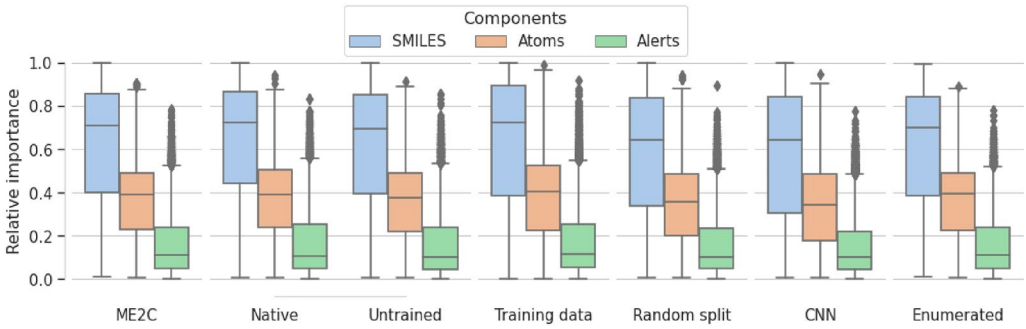


Fig. 14 Relative importance given to different components for experiments. Relative token importance given to smiles, atoms and atoms corresponding to expert-derived structural alerts. Relative importance is given for canonical representations and aggregated over all XAI methods of different pre-trained representation models

figures: P.H., F.K.; visualization, P.H.; supervision, S.G., I.T.; project administration, S.G., I.T.; funding acquisition, S.G., I.T..

Funding
This study has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie Actions Innovative Training Network European Industrial Doctorate grant agreement "Advanced machine learning for Innovative Drug Discovery (AIDD)" No. 956832.

Availability of data and materials
The cleaned ChEMBL and Ames files, as well as model parameters are available from Figshare: https://doi.org/10.6084/m9.figshare.24866091.

Code availability
Project home page: https://github.com/PeterHartog/augmented-xai. Operating system(s): Platform independent Programming language: Python 3 Other requirements: several open source python packages License: MIT. Any restrictions to use by non-academics: none.

Declarations

Ethics approval and consent to participate
Not applicable

Consent for publication
All authors have read and agreed to the published version of the manuscript.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Molecular AI, Discovery Sciences, R &D, AstraZeneca, 431 83 Mölndal, Sweden. [2]Institute of Structural Biology, Helmholtz Munich, Munich 85764, Germany.

Received: 20 December 2023 Accepted: 9 March 2024
Published online: 04 April 2024

References

1. Vellido A, Martín-Guerrero JD, Lisboa PJ (2012) Making machine learning models interpretable. In: ESANN, vol. 12, pp 163–172. Citeseer
2. Ali S, Abuhmed T, El-Sappagh S, Muhammad K, Alonso-Moral JM, Confalonieri R, Guidotti R, Del Ser J, Díaz-Rodríguez N, Herrera F (2023) Explainable artificial intelligence (XAI): What we know and what is left to attain trustworthy artificial intelligence. Inf Fus 99:101805
3. Sundararajan M, Taly A, Yan Q (2017) Axiomatic attribution for deep networks
4. Lundberg SM, Lee S-I (2017) A unified approach to interpreting model predictions. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R (eds) Advances in neural information processing systems, vol 30. Curran Associates Inc, New York
5. Ribeiro MT, Singh S, Guestrin C (2016) "why should i trust you?" explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 1135–1144
6. Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A (2016) Learning deep features for discriminative localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 2921–2929
7. Qiang Y, Pan D, Li C, Li X, Jang R, Zhu D (2022) AttCAT: explaining transformers via attentive class activation tokens. In: Koyejo S, Mohamed S, Agarwal A, Belgrave D, Cho K, Oh A (eds) Advances in neural information processing systems, vol 35. Curran Associates, Inc., New York, pp 5052–5064
8. Adebayo J, Gilmer J, Muelly M, Goodfellow I, Hardt M, Kim B (2018) Sanity checks for saliency maps. Adv Neural Inf Process Syst 31
9. Kindermans P-J, Hooker S, Adebayo J, Alber M, Schütt KT, Dähne S, Erhan D, Kim B (2019) The (un) reliability of saliency methods. Explainable AI: interpreting, explaining and visualizing deep learning, pp 267–280
10. Schwab P, Karlen W (2019) CXplain: causal explanations for model interpretation under uncertainty. In: Wallach H, Larochelle H, Beygelzimer A, Alché-Buc F, Fox E, Garnett R (eds) Advances in neural information processing systems, vol 32. Curran Associates Inc, New York
11. Hansch C, Fujita T (1964) p-σ-π analysis: a method for the correlation of biological activity and chemical structure. J Am Chem Soc 86(8):1616–1626
12. Karpov P, Godin G, Tetko IV (2020) Transformer-CNN: Swiss knife for QSAR modeling and interpretation. J Cheminf 12(1):1–12
13. Xu C, Cheng F, Chen L, Du Z, Li W, Liu G, Lee PW, Tang Y (2012) In silico prediction of chemical Ames mutagenicity. J Chem Inf Model 52(11):2840–2847
14. Gee P, Maron DM, Ames BN (1994) Detection and classification of mutagens: a set of base-specific salmonella tester strains. Proc Natl Acad Sci 91(24):11606–11610
15. Kamber M, Flückiger-Isler S, Engelhardt G, Jaeckh R, Zeiger E (2009) Comparison of the Ames II and traditional Ames test responses with respect to mutagenicity, strain specificities, need for metabolism and correlation with rodent carcinogenicity. Mutagenesis 24(4):359–366
16. Weininger D (1988) Smiles, a chemical language and information system. 1. Introduction to methodology and encoding rules. J Chem Inf Computer Sci 28(1):31–36
17. Wiseman S, Rush AM (2016) Sequence-to-sequence learning as beam-search optimization. arXiv preprint arXiv:1606.02960
18. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R (eds)

Advances in Neural Information Processing Systems, vol 30. Curran Associates Inc, New York
19. Devlin J, Chang M-W, Lee K, Toutanova K (2018) Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805
20. Lewis M, Liu Y, Goyal N, Ghazvininejad M, Mohamed A, Levy O, Stoyanov V, Zettlemoyer L (2019) Bart: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461
21. Öztürk H, Özgür A, Schwaller P, Laino T, Ozkirimli E (2020) Exploring chemical space using natural language processing methodologies for drug discovery. Drug Discov Today 25(4):689–705
22. David L, Thakkar A, Mercado R, Engkvist O (2020) Molecular representations in ai-driven drug discovery: a review and practical guide. J Cheminf 12(1):1–22
23. Xu M, Yoon S, Fuentes A, Park DS (2023) A comprehensive survey of image augmentation techniques for deep learning. Pattern Recognit 137:109347
24. Bjerrum EJ (2017) Smiles enumeration as data augmentation for neural network modeling of molecules. arXiv preprint arXiv:1703.07076
25. Tetko IV, Villa AE, Livingstone DJ (1996) Neural network studies. 2. Variable selection. J Chem Inf Computer Sci 36(4):794–803
26. Wang G, Li W, Aertsen M, Deprest J, Ourselin S, Vercauteren T (2019) Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks. Neurocomputing 338:34–45
27. Ayhan MS, Berens P (2022) Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks. In: Medical Imaging with Deep Learning
28. Langer M, Oster D, Speith T, Hermanns H, Kästner L, Schmidt E, Sesing A, Baum K (2021) What do we want from explainable artificial intelligence (XAI)?-a stakeholder perspective on XAI and a conceptual model guiding interdisciplinary XAI research. Artif Intell 296:103473
29. Kovaleva O, Romanov A, Rogers A, Rumshisky A (2019) Revealing the dark secrets of BERT. arXiv preprint arXiv:1908.08593
30. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2017) Grad-cam: visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision, pp 618–626
31. Huang K, Fu T, Gao W, Zhao Y, Roohani Y, Leskovec J, Coley CW, Xiao C, Sun J, Zitnik M (2021) Therapeutics data commons: machine learning datasets and tasks for drug discovery and development. arXiv preprint arXiv:2102.09548
32. Davies M, Nowotka M, Papadatos G, Dedman N, Gaulton A, Atkinson F, Bellis L, Overington JP (2015) ChEMBL web services: streamlining access to drug discovery data and utilities. Nucl Acids Res 43(W1):612–620
33. Mendez D, Gaulton A, Bento AP, Chambers J, De Veij M, Félix E, Magariños MP, Mosquera JF, Mutowo P, Nowotka M (2019) ChEMBL: towards direct deposition of bioassay data. Nucl Acids Res 47(D1):930–940
34. Landrum G (2006) RDKit: open-source cheminformatics software https://doi.org/10.5281/zenodo.7415128, https://www.rdkit.org. Accessed 9 Oct 2023
35. Kazius J, McGuire R, Bursi R (2005) Derivation and validation of toxicophores for mutagenicity prediction. J Med Chem 48(1):312–320
36. ...Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Kopf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, Chintala S (2019) PyTorch: an imperative style, high-performance deep learning library. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, Garnett R (eds) Advances in neural information processing systems, vol 32. Curran Associates Inc, New York, pp 8024–8035
37. Falcon W (2019). The PyTorch Lightning team: PyTorch Lightning https://doi.org/10.5281/zenodo.3828935. https://github.com/Lightning-AI/lightning. Accessed 19 Oct 2023
38. Kokhlikyan N, Miglani V, Martin M, Wang E, Alsallakh B, Reynolds J, Melnikov A, Kliushkina N, Araya C, Yan S, Reblitz-Richardson O (2020) Captum: a unified and generic model interpretability library for PyTorch
39. ...Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, Burovski E, Peterson P, Weckesser W, Bright J, van der Walt SJ, Brett M, Wilson J, Millman KJ, Mayorov N, Nelson ARJ, Jones E, Kern R, Larson E, Carey CJ, Polat İ, Feng Y, Moore EW, VanderPlas J, Laxalde D, Perktold J, Cimrman R, Henriksen I, Quintero EA, Harris CR, Archibald AM, Ribeiro AH, Pedregosa F, van Mulbregt P (2020) SciPy 1.0 contributors: SciPy 1.0—fundamental algorithms for scientific computing in python. Nat Methods 17:261–272. https://doi.org/10.1038/s41592-019-0686-2
40. Dabkowski P, Gal Y (2017) Real time image saliency for black box classifiers. Adv Neural Inf Process Syst 30
41. Zafar MB, Donini M, Slack D, Archambeau C, Das S, Kenthapadi K (2021) On the lack of robust interpretability of neural text classifiers. arXiv preprint arXiv:2106.04631
42. Ucak UV, Ashyrmamatov I, Lee J (2023) Improving the quality of chemical language model outcomes with atom-in-smiles tokenization. J Cheminf 15(1):55
43. Born J, Markert G, Janakarajan N, Kimber TB, Volkamer A, Martínez MR, Manica M (2023) Chemical representation learning for toxicity prediction. Digit Discov. https://doi.org/10.1039/D2DD00099G
44. Crabbé J, Schaar M (2023) Evaluating the robustness of interpretability methods through explanation invariance and equivariance. arXiv preprint arXiv:2304.06715
45. Lan M, Tan CL, Su J, Lu Y (2008) Supervised and traditional term weighting methods for automatic text categorization. IEEE Trans Pattern Anal Mach Intell 31(4):721–735
46. Erion G, Janizek JD, Sturmfels P, Lundberg SM, Lee S-I (2021) Improving performance of deep learning models with axiomatic attribution priors and expected gradients. Nat Mach Intell 3(7):620–631
47. Schwaller P, Hoover B, Reymond J-L, Strobelt H, Laino T (2021) Transformer-based neural networks capture organic chemistry grammar from unsupervised learning of chemical reactions. In: American Chemical Society (ACS) Spring Meeting
48. Fradkin P, Young A, Atanackovic L, Frey B, Lee LJ, Wang B (2022) A graph neural network approach for molecule carcinogenicity prediction. Bioinformatics 38(Supplement_1), 84–91

Publisher's Note

B Investigations into the Efficiency of Computer-Aided Synthesis Planning

JCIM JOURNAL OF CHEMICAL INFORMATION AND MODELING

pubs.acs.org/jcim Article

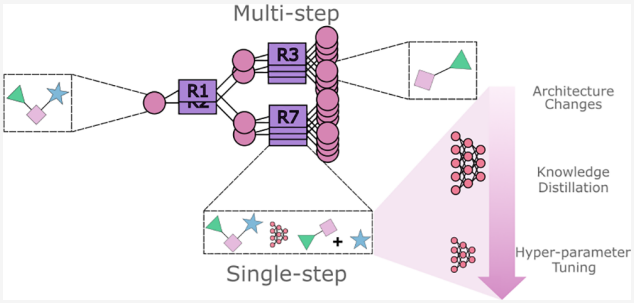
Investigations into the Efficiency of Computer-Aided Synthesis Planning

Peter B.R. Hartog,* Annie M. Westerlund, Igor V. Tetko, and Samuel Genheden*

Cite This: https://doi.org/10.1021/acs.jcim.4c01821 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: The efficiency of machine learning (ML) models is crucial to minimize inference times and reduce the carbon footprints of models deployed in production environments. Current models employed in retrosynthesis to generate a synthesis route from a target molecule to purchasable compounds are prohibitively slow. The model operates in a single-step fashion in a tree search algorithm by predicting reactant molecules given a product molecule as input. In this study, we investigate the ability of alternative transformer architectures, knowledge distillation (KD), and simple hyper-parameter optimization to decrease inference times of the Chemformer model. Initially, we assess the ability of closely related transformer architectures and conclude that these models under-performed when using KD. Additionally, we investigate the effects of feature-based and response-based KD together with hyper-parameters optimized based on inference sample time and model accuracy. We find that although reducing model size and improving single-step speed are important, our results indicate that multi-step search efficiency is more significantly influenced by the diversity and confidence of single-step models. Based on this work, further research should use KD in combination with other techniques, as multi-step speed continues to prevent proper integration of synthesis planning. However, in Monte Carlo-based (MC) multi-step retrosynthesis, other factors play a crucial role in balancing exploration and exploitation during the search process, often outweighing the direct impact of single-step model speed and carbon footprints.



INTRODUCTION

The efficiency of machine learning (ML) models is crucial to minimize inference times and reduce the carbon footprints of models.[1] This is especially pronounced for ML deployed in production environments, where model queries are more frequent. ML models have become standard in most areas of drug design, including computer-aided synthesis planning tools. Retrosynthesis analysis predicts the synthesis route from a target molecule to a purchasable material. This route is generated using a single-step ML model that predicts reactant molecules for a product molecule. The single-step model is then employed in a multi-step tree search to generate full synthesis routes.[2] Such algorithms typically only produce the reactants needed for the synthesis and need to be combined with other algorithms to predict reagents such as solvents and catalysts, as well as other reaction conditions.[3] Optimizing single-step retrosynthesis prediction becomes especially relevant as retrosynthesis tools are routinely used by chemists and the computational complexity of the multi-step search is heavily dependent on single-step performance.[4]

Single-step retrosynthesis models are either template-based methods, template-free methods, or a combination of both.[5] Transformer encoder-decoder architectures[6] have been used in retrosynthesis research[7−10] for their ability to scale to large data set and the promise of extrapolation to novel chemistry. However, no specific architecture has been recognized to outperform, as depending on training data and architectures cover different aspects of the chemical field.[11] Additionally, although single-step accuracy and multi-step success rates are high with template-free methods, inference and search times are too slow to be readily used in production.[12] Furthermore, recent work by Torren-Peraire et al.,[13] Maziarz et al.[14] highlights the difficulty of translating single-step performance to multi-step retrosynthesis success.

Knowledge distillation (KD)[15] exploits previously trained teacher models to train smaller and more efficient student models. The information from a teacher model can be transferred to a student model in three ways: (1) response-based KD,[15] (2) feature-based KD and (3) relation-based KD.[16] These approaches distill knowledge from the output, internal representations, or the angles/distances in between batches of internal representations, respectively (Figure 1).

Received: October 4, 2024
Revised: December 13, 2024
Accepted: December 18, 2024

Additionally, more efficient Transformer architectures have been pursued to increase the inference speed of encoder-decoder architectures. Efficient alternative Transformer architectures are a large area of research[17] and include models with induced sparsity for faster compute times,[18−20] and memory-efficient transformers with lower complexity.[21−24] Previous work specific to retrosynthesis also included increasing the efficiency of the sampling algorithms using speculative decoding.[25]

Here, we build on previous work on the AiZynthFinder tool[12,26] that has been used successfully in drug discovery projects.[4] We address the efficiency issue of a retrosynthesis Transformer model by employing alternative architectures, KD, and hyper-parameter optimization. We then analyze the impact these optimizations have on the fine-tuned models in terms of accuracy, speed, and carbon footprint. Importantly, we investigate whether single-step speed-up translates into multi-step retrosynthesis search times. This work thus demonstrates the promise of scaling up a retrosynthesis transformer model to reduce carbon footprints while retaining prediction accuracy and multi-step success rates.

METHODS

Data Collection and Processing. We adapted the Chemformer model[10] into different KD variants. The baseline Chemformer was originally pretrained on a masked auto-translation task on approximately 100 million randomly selected SMILES strings of molecules from the around 1.5 billion molecules in the ZINC-15 data set.[27] It was then fine-tuned on the USPTO-50k for the retrosynthesis task.[10] We collected the corresponding data set, original model weights, and parameters from Irwin et al.[10] The USPTO-50k data set contains around 50,000 reactions and was here used to benchmark the model variants. Each Chemformer variant was trained on UPSTO-full, gathered from Genheden et al.[28] Both USPTO-50k and UPSTO-full originated from the USPTO data set from Lowe.[29] To evaluate the multi-step search, we used 5000 ChEMBL structures from Genheden et al.[28] The stock was taken as the building block set of 24.7 million molecules from eMolecules.[30]

The original Chemformer vocabulary was constructed by applying regular expression matching, similar to Schwaller et al.,[9] to the canonical SMILES[31] of the molecules in the ChEMBL 27.[32] The total 523 tokens include 250 chemical tokens, 200 tokens that are unused during the pretraining stage but filled during fine-tuning, and 73 token meta-tokens, such as masking, padding, beginning-of-sentence, and end-of-sentence tokens, or tokens for the originally implemented molecular optimization task. Tokenization and augmentation of SMILES was performed by the ChemformerTokenizer which extends the routines in PySMILESUtils.[33]

Model Architectures. We experimented with three different model architectures in Chemformer, so-called model variants, including the original Chemformer architecture,[10] the Perceiver architecture,[23] and the Switch Transoformer.[20] The original Chemformer architecture is a BART (Bidirectional and Auto-Regressive Transformer) model with encoder-decoder architecture,[6,34,35] using sinusoidal positional embedding, prenorm layer-normalization, and GELU activation functions. A Perceiver variation was used,[23] keeping the original Chemformer architecture with the exception of an initial cross-attention layer in the Transformer encoder. This cross attention is combined with a learnable projection tensor, which was used for the query, and the embedded source sequence was used as the key-value pair, projecting the internal representation to a smaller dimension. Finally, to experiment with induced sparsity, we used the Switch Transformer[20] layer implementation of Varuna Jayasiri.[36] In the Switch transformer, the traditional feed-forward was replaced with eight experts with feed-forward layers of one-eight times the size of the original model.

Model Training and Hyper-Parameter Optimization. Model training was performed using PyTorch (version 2.2.2)[37] and Lightning (version 2.2.1)[38] on a single GPU. Most of the original training parameters from the Chemformer paper[10] were used to allow for direct comparisons (Table 1). Using Optuna,[39] hyper-parameter optimization was carried out using the Bayesian Tree-structured Parzen estimator (TPE).[40,41] Inspired by earlier work on hyper-parameter optimization for multi-step retrosynthesis,[42] we set the objectives for the optimization to the validation greedy search accuracy, and an adapted version of the rate-correct score (RCS).[43] It was obtained by dividing the correct samples by the cumulative time ($\frac{\text{Correct Samples}}{\text{Total Time}}$). The choices of the model parameters are given in Table 1.

Table 1. Training Hyper-parameters for the Baseline Implementations[a]

Parameter	Training		
	UPSTO-50k	UPSTO-full	Optimization
Batch size	128	128	128
Epochs	500	50	100
Scheduler	cycle	cycle	cycle
Learning rate	10^{-3}	10^{-3}	(10^{-2},10^{-3},10^{-4})
Model Dim	512	512	(64,128,256,512)
Feed-forward	2048	2048	(64,128,256,1024,2048)
Encoder Layer	6	6	(1,2,3,6)
Decoder Layer	6	6	(1,2,3,6)
Attention Head	8	8	8
Switch Experts	8	8	N/A
Perceiver Dim	32	32	N/A
Max seq length	512	512	512
Vocab Size	523	523	523
Dropout	0.1	0.1	0.1
Optimizer	Adam	Adam	Adam

[a]Multiple values represent choices for hyper-parameter optimization. Feed forward dimension was divided by the number of experts in Switch Transformer architectures.

Distillation was performed using implementations from Textbrewer (version 0.2.1 - post1),[44] which was adapted to Chemformer. Cross entropy losses were computed on the output values (response-based KD) or internal representations using direct comparison (feature-based representations) with or without standard cross-entropy loss, also named hard-label loss. All losses were combined using a weighted average based on weight hyper-parameters, normally set to 1.0, which were subsequently normalized to sum to one. Feature-based representations were compared between the output of the embedding, encoder, or decoder modules.

Carbon footprints were approximated using eco2AI (version 0.3.9).[45] CO_2 emissions were approximated by multiplying the energy consumption by the average emissions coefficient of a country, specifically Sweden.

https://doi.org/10.1021/acs.jcim.4c01821
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

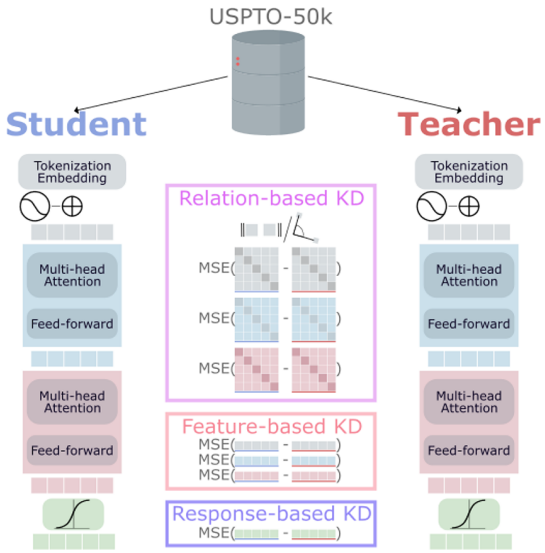


Figure 1. Illustrative example of different types of knowledge distillation. The information from a teacher model can be transferred to a student model in three ways. Response-based KD learns from the output of the teacher model. Feature-based KD, which learns from the internal representations. And relationship-based KD, which learns from the angles/distances between samples of the same models and correlates them between models.

Multi-step Search. The multi-step search was performed using AiZynthfinder (version 3.6.0).[2,26] To compare the performance of Chemformer to the current in-production model, we used the template model from Genheden et al.[28] The multi-step expansion policy was set to either template-based, Chemformer-based (including all models developed here) or multiexpansion. In the multiexpansion policy, the template-based was combined with the original Chemformer model trained on USPTO-50k. The maximum search time per search was set to 500 s, maximum iterations to 100, and the maximum depth to 6.

We evaluate the performance of the multi-step search in terms of search time (elapsed wall-clock time) and success rate. The success rate is defined as the percentage of the ChEMBL targets for which the search finds at least one synthesis route leading to starting material in the e-Molecules stock (also referred to as solvability). Although not a measure of route quality, it is a necessary condition to find the solved synthesis routes. The measure of route quality is a debated topic,[14,46] and herein we have chosen not to evaluate it because our primary objective is to compare the effect of different single-step models on multi-step search. To compare the routes produced by the different models, we compute the route similarity between the highest-ranked routes for each target.[47] The routes were ranked by the AiZynthFinder reward function, which takes into account the fraction of starting material in stock and the length of the route.[48]

RESULTS

Investigating Alternative Model Architectures for Retrosynthesis Prediction. In order to assess the general effects of using alternative architectures, we investigated three different architectures, the baseline BART transformer model, the mixture-of-experts Switch Transformer, and the linear scaling Perceiver model. Additionally, we studied the effects of halving the number of encoder and decoder layers and using KD on the various architectures. To isolate the effects of architecture and KD, all other hyper-parameters were kept identical to the baseline.

First, we look at the changes originating from the architecture changes alone. The Switch transformer architecture initially yielded similar effects as training from scratch with the baseline Transformer (Table 2), generating similar top-1 and top-10 accuracy. However, the Switch Transformer is more than twice as slow, 39.42 s over 17.20 s per batch, and has a significant increase in the carbon footprint of training similarly sized Transformers from scratch. The Perceiver architecture similarly yields a reduction in accuracy, 47.2% over 51.1% top-1 accuracy, but has the fastest inference times at 14.10 s per batch.

Next, we investigated the effects of halving the number of model weights from around 45 to around 23 M through the reduction of encoder and decoder layers. This architectural

Table 2. Effects of Alternative Model Architectures On Single-step Retrosynthesis Prediction[a]

model	size	sample time (s)	top 1 accuracy	top 10 accuracy	percentage invalid	percentage unique	CO_2 emissions (g)
			Full Size				
Chemformer	44.7 M	17.72	53.5	61.6	0.68	18.06	0.62
Transformer	44.7 M	17.20	51.1	64.2	0.89	22.03	0.48
Switch	47.9 M	39.42	50.9	64.3	1.04	24.54	0.78
Perceiver	45.7 M	14.10	47.2	57.5	1.02	22.92	0.66
			Half Size				
Transformer	22.6 M	9.25	50.8	70.8	1.34	31.44	0.28
Switch	24.2 M	16.00	49.6	70.3	1.64	35.29	0.38
Perceiver	23.7 M	7.84	42.8	57.2	1.91	31.92	0.25
			Knowledge Distillation				
Transformer	22.6 M	11.57	50.6	68.3	4.40	78.34	0.51
Switch	24.2 M	18.76	32.6	62.9	5.54	81.04	0.99
Perceiver	23.7 M	8.49	29.0	58.4	6.67	85.24	0.26

[a]All models are trained from scratch with different architectures, including an encoder-decoder transformer, an implementation of the Switch transformer with mixture-of-expert layers, and an implementation of the Perceiver model. CO_2 emissions are estimates based on kWh consumption, GPU type, and location of the compute cluster.

https://doi.org/10.1021/acs.jcim.4c01821
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

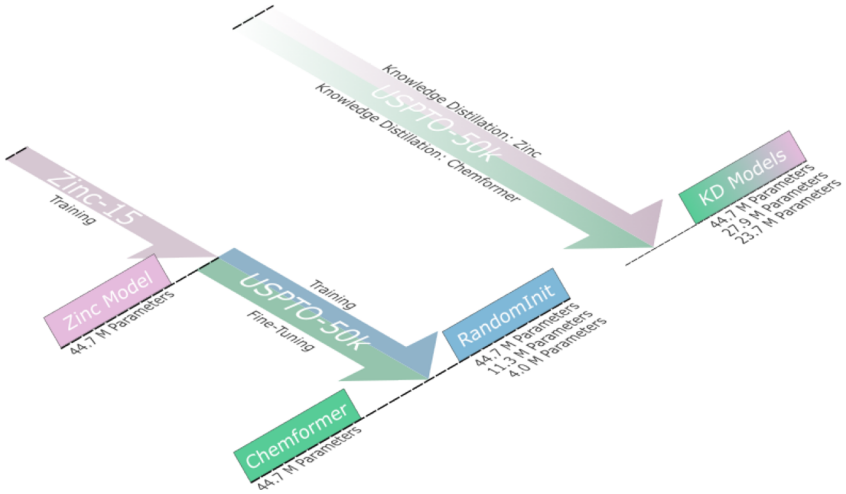


Figure 2. Illustrative example of model training types. Visualization of different types of model training employed, including standard training, fine-tuning, and knowledge distillation.

change yielded a small decrease in top-1 paired with an increase in top-10 accuracy for both the baseline Chemformer as well as the Switch Transformer (Table 2). Halving the amount of encoder and decoder layers resulted in slightly halved inference times. However, the Perceiver model experienced a marked decrease in top-1 accuracy when halving the model size compared to the BART and Switch models. Top-10 accuracy in the Perceiver also remained similar to the large-sized variant, while top-10 accuracy actually increased substantially in the other two architectures upon decreasing the model size. Carbon footprints also decreased in all half-size model variants, although the effect was less than the reductions in sample times.

Finally, we examined whether training the architecture variants with KD could reduce the inference time while preserving high accuracy. Response-based KD was used for all three variants, whereas feature-based KD on the Encoder and Decoder was used on the baseline and Switch transformer but was not used in the Perceiver model due to the changes of internal sequence length which are inherent to the model. The results of KD on the original Chemformer indicated that most of the reduction in top-1 accuracy from 53.5 to 50.6% could be explained by the significant increase of the invalid and unique molecules during inference (Table 2). When comparing the original Chemformer to the KD Transformer, we observed a reduction in sample time and carbon footprint, paired with slight reductions in model accuracy (50.6% vs 53.5%). The Switch transformer, however, displayed detrimental results when trained with KD, reaching at most 32.6% top-1 accuracy. The Perceiver KD had the worst results overall, with a top-1 accuracy of 29%, while the speed-up was not substantial compared to the BART Chemformer (8.49 s vs 11.57 s). Notably, this difference is likely due to the lack of a feature-based KD.

Overall, reducing the number of encoder and decoder layers decreases batch inference times and carbon footprints but impacts model accuracy when evaluated on similar training times. Additionally, changes in architecture are not directly compatible with the KD of the original architecture, as indicated by the low top-1 and top-10 accuracy scores when using KD on alternative architectures. In conclusion, for single-step prediction, decreasing the size of the BART Chemformer variant appears to be more efficient (in accuracy, sample time, and fraction of invalid) than employing knowledge distillation on any architecture variant.

Hyper-parameter Optimization Using Knowledge Distillation To Increase Inference Speed. To assess the impact that hyper-parameters have on single-step retrosynthesis for speed, Bayesian optimization was used to optimize the types and proportions of KD during training, model dimension, feed-forward dimension, number of attention heads, and number of encoder and decoder layers (Figure 3). We used the optimization objectives of validation accuracy and the validation rate-correct score (RCS) on one of three training variations: training from scratch (randomly initializing the weights with Xavier uniform distribution), response-based, and feature-based KD using the original Chemformer (KD: Chemformer) or the pretrained model trained on Zinc (KD: Zinc). The Pareto front indicates the highest validation accuracy in combination with the fastest average inference time, shown in red.

First, we observe the overall model quality generated by the hyper-parameter optimization. By optimization on either high validation accuracy or RCS, models were obtained that were both fast and accurate. However, the Pareto front shows that no model achieves a top-1 accuracy above or near the original Chemformer model of 53.5%. The highest validation accuracy was achieved using the optimal parameters from randomly initialized models, 46.7% (Table A1, RandomInit). Overall, most configurations were dominated by KD using the pretrained Zinc model (Figure 3), with the highest accuracy models on the Pareto front coming from the randomly initialized and KD Chemformer models.

Finally, specific hyper-parameters that contributed to the highest validation accuracy and RCS models are identified (Table A2). All models had the maximum number of six encoder layers and fewer than six decoder layers, with the exception of the slowest, best-performing randomly initialized model. Furthermore, all chosen models, with the exception of the randomly initialized models, selected the largest possible feed-forward and embedding dimensions of 2048 and 512, respectively. With respect to KD, models distilling from the Zinc models selected high values for all possible KD values, whereas models distilling from Chemformer prioritized

https://doi.org/10.1021/acs.jcim.4c01821
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

embedding and decoder feature-based KD over encoder feature-based KD and response-KD, with all preferring training knowledge distillation in combination with standard training. In conclusion, hyper-parameter optimization, in general, showed that on average KD based on the pretrained zinc model is closest to the Pareto front; however, all training methods have models on the Pareto front.

Effects of Hyper-Parameters on Fully Trained Models. Here, we train a variety of models based on the Transformer architecture to analyze the effects of hyper-parameter tuning on both the original data set and transferring these to the larger data set (Figure 2). Table 3 shows a full breakdown of beam search accuracies including and excluding valid and unique SMILES.

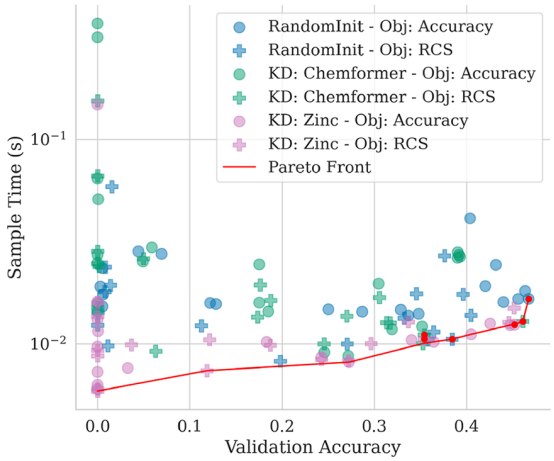


Figure 3. Results of hyper-parameter optimization with two optimization objectives and three training variations. Greedy search validation accuracy of hyper-parameter optimization with Pareto front of most accurate and fastest models. Three training variations include training from scratch, KD using the original Chemformer, and KD using the pretrained model trained on Zinc. Red points represent models chosen for further analysis for each variation of the highest rate correct score and highest accuracy.

When analyzing the randomly initialized transformer models trained on USPTO-50k, we compared each model with baseline parameters to the model with optimal hyper-parameters in terms of accuracy and RCS (Figure 4). When optimizing for accuracy, the resulting model had around 25% of the original size with only a 1.4% decrease in top-1 accuracy, coupled with a 5% increase in top-10 accuracy (Table 3). Optimizing on the RCS instead resulted in a model 10% of the original size with a 5% drop in the top-1 accuracy and a 16% increase in the top-10 accuracy. Notable, even though the sampling time dropped from 9.67 to 6.52 s between optimizing on accuracy instead of RCS, the carbon footprint was slightly higher for the RCS-optimized model (0.40 g instead of 0.34 g). However, both were substantially lower than the original implementation, indicating that carbon footprints decreased in these optimized models.

Subsequently, we analyze the difference between the original model and the newly retrained version to verify that changes in implementation did not result in significant changes in baselines. The results are shown in Table 3. First, we observed no significant difference between accuracy scores and only slight deviations in invalid and unique values. However, a significant difference was observed in the carbon footprints, which can be mostly attributed to the loading of model weights of the original model over inference after training. Additionally, models were also trained using the lowest validation loss, which showed lower top-1 accuracy but increased top-10 accuracy in all randomly initialized and fine-tuned models (Table A5). The highest change was the Chemformer (FT-Zinc) which went from 53.6 top-1 and 61.7 top-10 accuracy to 48.3 top-1 and 79.7 top-10 accuracy, for the last and optimal validation epoch, respectively. In this setting, the optimized models showed minimal differences or even improvements with respect to the randomly initialized smaller models, all reaching above 79% top-10 accuracy.

Furthermore, we compared the effects of KD on the full model predictions using the original Chemformer and the Zinc model with the original. Using KD, all models are significantly faster with respect to the randomly initialized models and Zinc fine-tuned models. However, this is paired with a decrease in the top-1 and top-10 accuracy scores. Furthermore, significant increases in the percentage of invalid smiles generated are observed in the KD models. Specifically, the best-performing Chemformer-based KD with optimal hyper-parameters for accuracy has a top-1 accuracy of 48.5% compared to 53.5% of the original Chemformer. A major explanation for this is the high percentage of invalid predictions, which range from 2.78% up to 23.32% for the KD models. These models are on average half the size of the original implementation but are as fast or faster than the randomly initialized model optimized for RCS, which is only 10% of the original model.

When looking at the ability of transferring optimal hyper-parameters across related training data, we observe similar effects found in the USPTO-50k models but more pronounced. The randomly initialized baseline model has a 6% higher top-1 and 4% higher top-10 accuracy over the smaller randomly initialized model with hyper-parameters optimized on accuracy from the UPSTO-50k data set (Table 3). The decrease in sample time and carbon footprint was also less pronounced between the accuracy model and the baseline model. The decrease in accuracy was larger in the randomly initialized model with parameters optimized for RCS, where accuracy dropped to 29.8 and 57.8%, for top-1 and top-10 respectively. This constitutes a 16.7 and 14.9% drop in accuracy from the baseline model for top-1 and top-10 accuracy, respectively.

Overall, all optimized models are faster but less accurate than the baseline and original implementations. Furthermore, the original implementations outperform all other models in top-1 accuracy but have lower top-10 accuracy and percentage of unique predictions. Additionally, the original implementations have carbon footprints higher than those of the newly trained models. Finally, we additionally investigated the influence of the invalid percentage and the unique percentage on top-n accuracies and observed minimal differences between top-n accuracy with or without valid and uniquely valid predictions (Table A3).

Translating Single-Step into Multistep. In order to assess whether single-step speed-ups translate to a multistep retrosynthesis search, a full multistep retrosynthesis search on 5000 ChEMBL molecules was analyzed for each of the single-step models. We compared model performances with a small template-based model which is currently used as the default in the AiZynthFinder framework. We focus our analysis on the speed, carbon footprint, and percentage of targets for which a route to starting material in stock is found (percentage solved).

https://doi.org/10.1021/acs.jcim.4c01821
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

Table 3. Single-step Model Full Training Model Statistics Based on the Transformer[a]

training	size	sample time (s)	top 1 accuracy	top 10 accuracy	percentage invalid	percentage unique	CO_2 emissions (g)
USPTO-50k				Original			
RandomInit	44.7 M	17.97	50.6	63.1	0.93	21.90	0.70
FT-Zinc	44.7 M	17.45	53.5	61.6	0.68	18.06	2.01
				Baselines			
RandomInit	44.7 M	16.20	51.3	63.7	0.87	21.75	0.76
FT-Zinc	44.7 M	16.73	53.6	61.7	0.82	18.61	0.61
				Optimized: Randomly Initiated			
Opt: Acc	11.3 M	9.67	50.9	70.2	1.14	30.96	0.34
Opt: RCS	4.0 M	6.52	45.8	79.5	2.52	62.04	0.40
				Optimized: KD Chemformer			
Opt: Acc	27.9 M	6.94	48.5	60.4	2.78	29.71	0.33
Opt: RCS	23.7 M	4.58	43.3	55.7	23.32	57.27	0.29
				Optimized: KD Zinc			
Opt: Acc	27.9 M	6.64	19.0	53.5	7.92	60.56	0.33
Opt: RCS	23.7 M	4.30	19.5	49.1	21.03	71.73	0.24
USPTO-full				Baselines			
RandomInit	44.7 M	29.69	46.5	72.7	1.10	66.46	12.30
FT-Zinc	44.7 M	24.22	47.9	72.6	0.98	60.53	6.42
				Optimized: Randomly Initiated			
Opt: Acc	11.3 M	21.85	40.2	68.6	1.23	73.21	9.40
Opt: RCS	4.0 M	7.55	29.8	57.8	1.64	75.59	2.25

[a]Models are trained either from scratch (RandomInit), fine-tuned from training on Zinc (FT-Zinc), using KD on the Chemformer model (KD-Chem) or KD on the zinc model (KD-Zinc). Hyper-parameters are either from the original publication (original), retrained models (baseline), or using hyper-parameters gathered from an Optuna search (opt: Acc/RCS) where specific parameters depend on the training set. Models are then trained and evaluated on USPTO-50k and USPTO-full. Reported statistics are based on beam-search results on random split test set values.

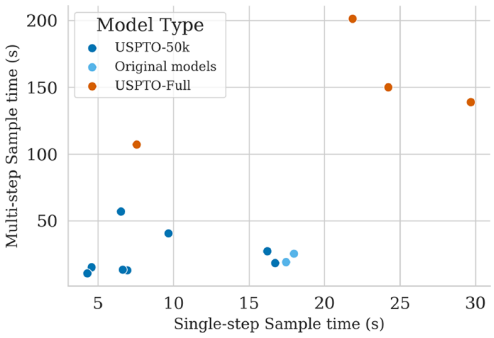


Figure 4. Median multi-step sample times set against single-step sample time. Median sample times of 5000 ChEMBL molecules set out against single-step average beam search sample times. Models include those trained on USPTO-50k, including the original models highlighted in light blue and models trained for USPTO-full.

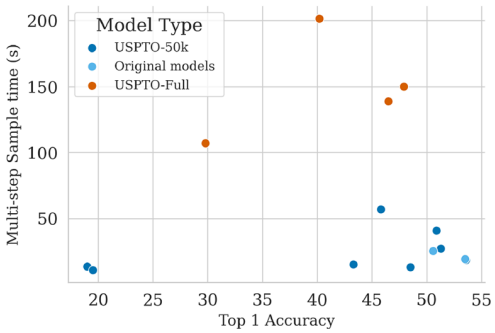


Figure 5. Median multi-step sample times set against top-1 single-step accuracy. Median search times of 5000 ChEMBL molecules set out against the accuracy of single-step models. Models include those trained on USPTO-50k, including the original models highlighted in light blue, and models trained for USPTO-full.

Carbon footprints are noted in grams and are mostly in agreement with search times.

First, we analyzed the differences between the original implementations and newly trained baselines (Table 4). When comparing the original implementations of the randomly initialized model and the Chemformer model and fine-tuning the Zinc model (FT-Zinc), we observed only small differences (Table 4). Most notable was the difference in the percentage solved between the original and the retrained baseline of the Chemformer model which decreased from 54.6 to 56.2% and the small decreases in carbon footprints from the original implementations, 196.34 and 123.46 g, to the baseline implementations, 168.15 and 110.17 g, for the randomly initialized and Chemformer models, respectively.

When investigating model behaviors on a single- and multistep basis, we could draw five key conclusions. First, an increased training data size, as exemplified by going from USPTO-50k to USPTO-full, showed significant increases in percentage solved but also markedly increased both search time and carbon footprints, with similar model calls and policy probabilities. Second, single-step top-1 is not predictive of the number of model calls (Figure 5). In contrast, multistep search times are more correlated with the average number of model calls per search. The number of model calls reflects how often the single-step model was queried over the retrosynthesis search. Third, the top-10 accuracy and policy probability are the major predictors of the average number of model calls, with lower policy probability correlating with a higher number of

https://doi.org/10.1021/acs.jcim.4c01821
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

Table 4. Multi-Step Retrosynthesis Search[a]

	median	average over search			average per molecule		
	search	search	CO_2	percentage	# solved	policy	# model
training	time (s)	time (h)	emissions (g)	solved (%)	routes	probability	calls
USPTO-50k			Including template-based model				
template-based	26.01	40	0.25	67.3	20.74	0.17	446
multiexpansion	382.08	495	2482.29	89.0	159.17	0.19	1832
			Original				
RandomInit	25.48	62	196.34	61.5	132.92	0.48	1384
FT-Zinc	19.16	47	123.46	56.2	125.21	0.53	1259
			Baselines				
RandomInit	27.29	69	168.15	60.4	134.19	0.48	1391
FT-Zinc	18.54	46	110.17	54.6	120.21	0.54	1277
			Optimized: Randomly Initiated				
Opt: Acc	40.96	94	237.94	69.7	136.59	0.39	1641
Opt: RCS	57.71	93	258.59	84.8	134.16	0.22	1733
			Optimized: KD Chemformer				
Opt: Acc	13.19	35	106.63	60.3	110.48	0.40	1053
Opt: RCS	15.37	34	97.60	62.4	65.99	0.28	773
			Optimized: KD Zinc				
Opt: Acc	13.59	29	101.10	63.9	37.77	0.15	442
Opt: RCS	10.82	25	61.89	63.5	35.50	0.16	365
USPTO-Full			Baselines				
RandomInit	138.87	259	967.68	87.5	119.53	0.22	1493
FT-Zinc	149.99	271	1340.05	86.8	116.00	0.22	1472
			Optimized: Randomly Initiated				
opt: acc	201.37	325	975.75	88.4	116.86	0.20	1500
Opt: RCS	107.15	180	552.17	91.1	122.95	0.17	1701

[a]Translating effects of single-step models to multi-step search. The models used here are the standard template-based model, a multi-expansion model combining template-based, and the original Chemformer. Furthermore, it includes the original implementations of the Chemformer paper, randomly initialized and fine-tuned from the pretrained Zinc model (RandomInit, FT-Zinc) and the retrained baselines. Additionally, optimized versions used either accuracy or rate correct score (OptAcc, OptRCS) of the randomly initialized (RandomInit), KD using Chemformer (KD-Chemformer), and KD using the pretrained Zinc model (KD-Zinc). Finally, it also includes models trained on USPTO-full.

model calls. The exceptions are the template-based and KD models, where low policy probability is more associated with fewer model calls, as well as the KD models with a high percentage of invalid molecule predictions where lower policy probability is also associated with fewer model calls. Fourth, the percentage of searches where a molecule has a solved route is also inversely correlated with single-step top-1 accuracy. Additionally, the percentage solved is associated with the number of model calls, where fewer calls correlate to a higher percentage solved. Finally, choosing the final epoch as done in previous research results in faster but fewer solved routes (Table A6). We observe that using the lowest validation loss checkpoints, with lower top-1 accuracies, but higher top-10 accuracies, results in significant increases in model calls. This is paired with increases in percentage solved, where the baseline FT-Zinc increases from 54.6 to 84.1% when the lowest validation weights are used.

Taken together, an uncertain diverse model may lead to more model calls, as it promotes exploration in the tree search. The multi-step efficiency of a single-step model thus depends on the overall feasibility and quality of the predicted top-10 reactions. Moreover, the ability to properly rank predictions becomes important, something that is difficult to assess in a single-step setting. While all predictions produced by the single-step model affect the multi-step performance, the top 10 only considers the best prediction.

Route Similarity. To investigate the similarity between the predicted routes from different models, the average route similarity was analyzed using a recently published metric.[47] In this analysis, we only compared the top-ranked routes of each target and only made comparisons where both methods produced a route to commercial starting material (Figure 6). Most methods show an average route similarity of approximately 0.7 or higher. This indicates that the methods on average produce routes that share a majority of the synthetic strategy. Thus, these methods would produce routes of similar quality. However, the two KD-Zinc models stand out with their low route similarities to those of the other model routes. This can be explained by these two models producing routes with fewer starting materials relative to the route length, which is not considerably different compared to routes generated by the other models (Table A4). In Figure A1, we show routes for an example target, CHEMBL1668049, that can be synthesized in only one step. The template-based and KD-Chemformer-OptRCS models suggest simple esterification. This is likely the suggestion of KD-Zinc-OptRCS as well, but only one of the reactants is produced. This causes the similarity metric to return zero, as it cannot determine that the bond-forming similarity is in fact identical in all of these three routes. This feature is consistent for the KD-Zinc models; 44.1% of the routes produced by the KD-Zinc-OptRCS model end up in a single starting material, compared to 11.8 and 1.8% for the

https://doi.org/10.1021/acs.jcim.4c01821
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

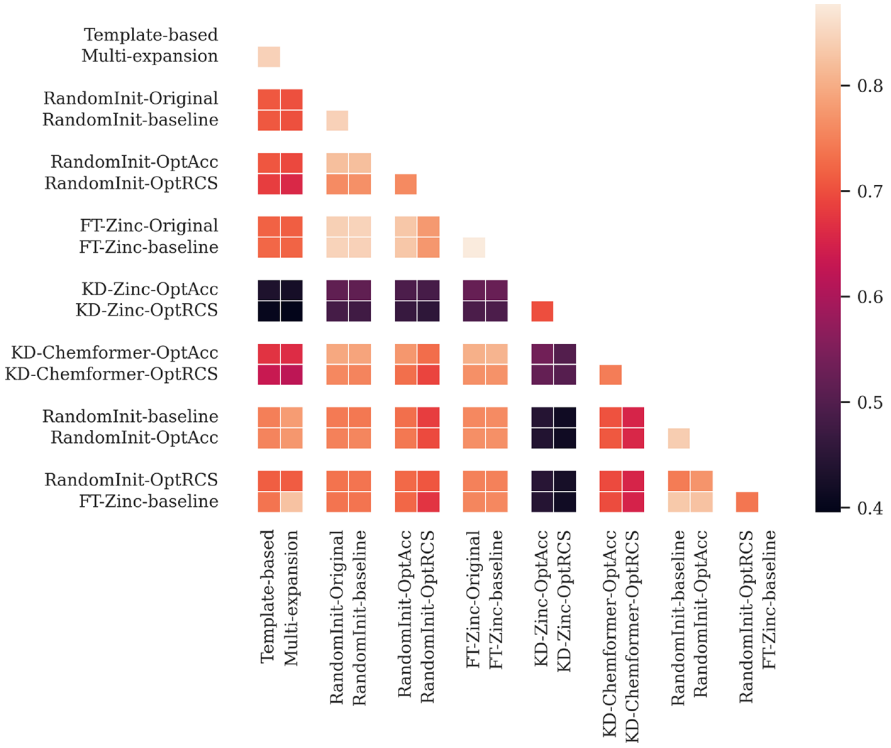


Figure 6. Comparison of route similarity of the highest ranked routes of each model. Route similarity indicates that most models have high route similarity and therefore high route quality, with the exception of KD-Zinc models, which predict significantly less reactants per reaction step.

KD-Chemformer-OptRCS and template-based models, respectively. Five illustrative examples are shown in Figure A2 to give intuition about the general differences between template-based and template-free methods. For some disconnections, the second reactant may be small such that it exists in the stock, e.g., a bromination agent or the reactant in the example above. However, it is likely that the second reactant in some cases requires further disconnections to reach the starting materials.

DISCUSSION

In this study, we adopted alternative transformer architectures and knowledge-distillation (KD) to boost the retrosynthesis prediction efficiency with Chemformer. Model training variations were compared to the original Chemformer, uncovering the complex interplay between the single-step model and multi-step search. Specifically, we focused on the speed-up in the single-step model variants and how this influenced multi-step retrosynthesis performance. This builds on earlier work with the AiZynthFinder tool[12,26] which has been used extensively in drug discovery projects.[4] The models developed herein can furthermore be used in other retrosynthesis frameworks such as Synthesus[14] or ASKCOS.[49]

Single-Step Retrosynthesis Prediction. *Alternative Model Architectures.* The initial findings indicated several important aspects. First, our investigation extended to alternative model architectures, including the Switch Transformer[20] and the Perceiver model.[23] The Switch Transformer showed similar accuracy as the baseline Transformer model but was significantly slower. This was possibly due to a suboptimal implementation and the usage of only a single GPU, making the sparsity effects of the original implementation less effective. The Perceiver model showed promise, having similar accuracy to the transformer trained from scratch while being faster. However, we found that these alternative architectures under-performed when coupled with KD. The Switch transformer and the Perceiver models were chosen as alternative architectures as these architectures were relatively closely related to, though projected to be faster than the original Transformer architecture. The fact that these model variants under-performed when trained with KD indicates that the benefits of KD may not be applicable to all model architectures and emphasizes the need for careful evaluation. One notable aspect here is that the Perceiver model, which uses different dimensions, was unable to use the feature-based KD, and thus solely relied on the response-based KD. Future work could focus on implementing and evaluating the relation-based KD,[16] which uses comparisons of in-between batch distances and angles to match between models, making them more impervious to dimensional changes.

Hyper-Parameter Optimization. During hyper-parameter optimization, we found that the Pareto front contained mostly smaller decoders and larger encoders. This is expected behavior as the inference calculations were based on autoregressive greedy search, which iteratively calls the decoder to make its predictions. Using Optuna, we trained 150 models, including three training settings from random initialization, KD using Chemformer, and KD using the pretrained Zinc model. However, the Optuna search allowed for duplicate parameter suggestions, resulting in on average seven to eight duplicate model settings. This is likely because some models had especially poor validation averages (around 0% accuracy margin). Additionally, no model reached a high accuracy, none higher than 46.7%. This is likely due to the theoretical maximum of the model parameters, where the original implementation parameters were set as the maximum.

https://doi.org/10.1021/acs.jcim.4c01821
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

Finally, the Pareto front mostly included pretrained KD models.

Fully Trained Models and KD. Translating the effects of hyper-parameter optimization to fully trained models, we observed mixed results. First, comparing optimized models with baseline implementations indicated that we were successful in creating increasingly faster models. Interestingly, KD models had around half of the baseline implementations but were similar or even faster than the fastest trained from scratch model with five times fewer model parameters. Another interesting finding is that speed did not necessarily translate one-to-one with decreases in carbon footprints. This is possibly due to the rather basic method of calculating carbon footprints through power usage over the inference time. However, we did observe a general trend where smaller, faster models translate to lower carbon emissions. Moreover, we also observed a substantial increase in the number of invalid smiles generated using smaller models, especially when using KD. This does indicate that a smaller decoder size comes with a cost, and KD might introduce competing objectives during training, meaning that the model might prioritize similarity to the original model over correct SMILES prediction. This could be the reason why the pretrained models dominated the Pareto front but had the worst performance during full-size training. Finally, we observed that using the last epoch for inference on USPTO-50k results in models more optimized on top-1 accuracy, where models using the optimal validation loss (around 100−200 epochs) result in models better optimized for top-10.

Overall in the fully trained models, KD generally leads to lower accuracy compared to standard training methods, which aligns with previous research,[15] possibly due to capacity issues.[50] Relation-based KD could play a bigger role possibly increasing accuracy in future research,[16] especially when considering alternative architecture. Additionally, using the top-10 accuracy in USPTO-50k might be a misleading benchmark, as we observed that the top-1 accuracy was inversely related to the top-10 accuracy in USPTO-50k but not in USPTO-full. This indicates that the long training on USPTO-50k does not translate well to beam search accuracy, as the model becomes overconfident in optimizing for low-diversity, high-accuracy responses over diversity.

Multi-step Retrosynthesis Is Less Dependent on Single-Step Expansion Models than Policy. *Impact of Single-Step Speed in Multi-step.* In this research we also investigated the relation between single-step speed and multi-step speed; however, this relation was less predictive than initially expected. We mostly find that multi-step search times are associated more with a higher number of model calls, which in turn is impacted by higher average policy probabilities. This average policy probability is a reflection of model confidence, thereby influencing the exploration/exploitation trade-off in the Monte Carlo search as it depends on the prediction probability. This is further supported by the smaller number of model calls made by the original models. Future research into increasing multi-step speed might prefer to look into optimizing the exploration/exploitation axis of the multi-step research.

Solvability and Exploration. Furthermore, there is also a relation between the percentage of solved targets (also referred to as solvability or success rate) and the number of calls, which can be interpreted as higher exploration, resulting in more solved routes. However, there also is a slight inverse relationship between the percentage solved and the single-step top-1 accuracy, meaning that the accuracy of the solved routes might not be as precise as routes from higher-accuracy single-step models. However, top-10 is positively related to the number of calls and solvability, especially when looking at using optimal validation loss. This indicates that future research interested in multi-step solvability should focus on top-10 accuracy, not top-1 accuracy in single-step models. Finally, we observe that the models trained on the larger USPTO-full model have an increased percentage solved, which is expected due to their higher chemical space coverage. However, this is also paired with a significantly decreased inference speed, even with identical model parameters. In previous research, the original Chemformer has been shown to under-perform on USPTO-50k compared to other models, but outperform on larger, more complex data sets.[12,13,51] This can likely be explained by the increase in the percentage of unique predictions from beam search, which might also translate to better solvability for more difficult targets. Although solvability does not correspond to route quality, it is a necessary condition to find solved routes,[14,46] and as we focus on comparison between models in this study we leave the assessment of route quality for future studies. We include several examples of retrosynthesis plans in the Support Information for the interested reader and publish all produced routes as a digital download.

Route Similarity. Because the template-based model is used extensively in drug discovery projects,[4] we are confident that this is a good baseline for comparing routes. Therefore, we used a route similarity metric[47] and found that most models produce highly similar routes compared with the baseline template-based model (around 0.7−0.8). In other words, the resulting routes are often very similar regardless of the solvability or speed. The exception to this was the routes generated with KD-Zinc models, which exhibit considerably lower similarity. The reason for this route disparity stems from the propensity of KD-Zinc models to predict only a single reactant for the reactions. Notably, these models were distilled from the pretrained model, which was trained to predict single molecules. These conclusions are supported by both route statistics and the illustrative example given in the route analysis case study.

In general, using single-step models in a multi-step setting, we need to further evaluate and determine the translation of effects of single-step characteristics on multi-step behavior. Specifically, in the field of retrosynthesis, multi-step behavior analysis is still an open question with end-users having various preferences. For example, a reduction in accuracy might be acceptable given the advantages in model size and efficiency gained. Specifically, as mentioned before, single-step retrosynthesis accuracy may not directly translate in a multi-step setting, where diversity might play a more important role[13] and speed might allow for higher exploration. In general, exploration versus time costs should be evaluated further, and exploration should be better calibrated based on single-step models, not necessarily viewed as independent.

CONCLUSIONS

In this research, we investigated the potential of hyper-parameter tuning, alternative model architectures, and knowledge distillation (KD) to accelerate both single-step and multi-step retrosynthesis prediction. The initial motivation was to reduce model size in a system that makes multiple model queries, hypothesizing that speed increases in a single-step

https://doi.org/10.1021/acs.jcim.4c01821
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

setting would have a compounding effect in a multi-step setting. While KD offered promising speed improvements, these gains were often accompanied by reduced accuracy and the effectiveness of KD varied significantly across different model architectures. This suggests that the benefits of KD are architecture-dependent and that careful tuning is required to avoid undesirable trade-offs. Further investigations into multispeed settings uncovered important aspects of single-step retrosynthesis models that impact multi-step, including top-10 accuracy influence on solvability and the influence of exploration on multi-step speed. Future research into retrosynthesis prediction should focus on optimizing the exploration/exploitation dynamics, particularly in multi-step scenarios, and further refine KD techniques, alternative architectures, and hyper-parameter optimizations to balance speed and accuracy. Investigating hybrid methods that combine the strengths of KD with advanced model architectures may also prove useful in improving both the efficiency and the robustness of retrosynthesis predictions. Although reducing model size and improving single-step speed are important, our results indicate that multi-step search efficiency is more significantly influenced by the diversity and confidence of single-step models. In Monte Carlo-based (MC) multi-step retrosynthesis, these factors play a crucial role in balancing exploration and exploitation during the search process, often outweighing the direct impact of single-step model speed.

■ ASSOCIATED CONTENT

Data Availability Statement

All raw data, generated data and trained models are available on FigShare: 10.6084/m9.figshare.27908229.v1. Project home page: https://github.com/PeterHartog/fast-retro. Operating system(s): Platform independent. Programming language: Python 3. Other requirements: several open source python packages. License: MIT.

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jcim.4c01821

Appendix with Tables and Figures (PDF)

■ AUTHOR INFORMATION

Corresponding Authors

Peter B.R. Hartog − *Molecular AI, Discovery Sciences, R&D, AstraZeneca, 431 83 Mölndal, Sweden; Institute of Structural Biology, Molecular Targets and Therapeutics Center, Helmholtz Munich - German Research Center for Environmental Health (GmbH), 85764 Neuherberg, Germany;* orcid.org/0000-0001-6406-6234; Email: peter.hartog@astrazeneca.com

Samuel Genheden − *Molecular AI, Discovery Sciences, R&D, AstraZeneca, 431 83 Mölndal, Sweden;* orcid.org/0000-0002-7624-7363; Email: samuel.genheden@astrazeneca.com

Authors

Annie M. Westerlund − *Molecular AI, Discovery Sciences, R&D, AstraZeneca, 431 83 Mölndal, Sweden;* orcid.org/0000-0003-2288-5711

Igor V. Tetko − *Institute of Structural Biology, Molecular Targets and Therapeutics Center, Helmholtz Munich - German Research Center for Environmental Health (GmbH), 85764 Neuherberg, Germany;* orcid.org/0000-0002-6855-0012

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jcim.4c01821

Author Contributions

Conceptualization, P.B.R.H., I.V.T., S.G.; Methodology, P.B.R.H., A.M.W., S.G.; Validation, P.B.R.H., A.M.W.; Writing original draft preparation, P.B.R.H.; Writing, review and editing, P.B.R.H., A.M.W., E.S., I.V.T.; Figures: P.B.R.H.; Visualization, P.B.R.H.; Supervision, S.G., I.V.T.; Project administration, S.G., I.V.T.; Funding acquisition, S.G., I.V.T.

Funding

This study has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie Actions Innovative Training Network European Industrial Doctorate grant agreement "Advanced machine learning for Innovative Drug Discovery (AIDD)" No. 956832.

Notes

The authors declare no competing financial interest.

All authors have read and agreed to the published version of the manuscript.

■ ACKNOWLEDGMENTS

The author thank the doctoral candidates and supervisors from the Marie Skłodowska-Curie Innovative AIDD consortium for their support.

■ REFERENCES

(1) Strubell, E.; Ganesh, A.; McCallum, A. Energy and policy considerations for deep learning in NLP. *arXiv preprint arXiv:1906.02243* **2019**.

(2) Genheden, S.; Thakkar, A.; Chadimová, V.; Reymond, J.-L.; Engkvist, O.; Bjerrum, E. AiZynthFinder: a fast, robust and flexible open-source software for retrosynthetic planning. *J. Cheminform.* **2020**, *12*, 70.

(3) Kreutter, D.; Reymond, J. Multistep retrosynthesis combining a disconnection aware triple transformer loop with a route penalty score guided tree search. *Chemical Science* **2023**, *14*, 9959−9969.

(4) Shields, J. D.; Howells, R.; Lamont, G.; Leilei, Y.; Madin, A.; Reimann, C. E.; Rezaei, H.; Reuillon, T.; Smith, B.; Thomson, C.; et al. AiZynth impact on medicinal chemistry practice at AstraZeneca. *RSC Medicinal Chemistry* **2024**, *15*, 1085−1095.

(5) Zhong, Z.; Song, J.; Feng, Z.; Liu, T.; Jia, L.; Yao, S.; Hou, T.; Song, M. Recent advances in deep learning for retrosynthesis. *WIREs: Comput. Mol. Sci.* **2024**, *14*, No. e1694.

(6) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 2017, *30*.

(7) Karpov, P.; Godin, G.; Tetko, I. V. A transformer model for retrosynthesis. *International Conference on Artificial Neural Networks* **2019**, *11731*, 817−830.

(8) Tetko, I. V.; Karpov, P.; Van Deursen, R.; Godin, G. State-of-the-art augmented NLP transformer models for direct and single-step retrosynthesis. *Nat. Commun.* **2020**, *11*, 5575.

(9) Schwaller, P.; Laino, T.; Gaudin, T.; Bolgar, P.; Hunter, C. A.; Bekas, C.; Lee, A. A. Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. *ACS central science* **2019**, *5*, 1572−1583.

(10) Irwin, R.; Dimitriadis, S.; He, J.; Bjerrum, E. J. Chemformer: a pre-trained transformer for computational chemistry. *Machine Learning: Science and Technology* **2022**, *3*, No. 015022.

(11) Hastedt, F.; Bailey, R. M.; Hellgardt, K.; Yaliraki, S. N.; del Rio Chanona, E. A.; Zhang, D. Investigating the Reliability and

Interpretability of Machine Learning Frameworks for Chemical Retrosynthesis. *Digit. Discov.* **2024**, *3*, 1194−1212.
(12) Westerlund, A. M.; Manohar Koki, S.; Kancharla, S.; Tibo, A.; Saigiridharan, L.; Kabeshov, M.; Mercado, R.; Genheden, S. Do Chemformers Dream of Organic Matter? Evaluating a Transformer Model for Multistep Retrosynthesis. *J. Chem. Inf. Model.* **2024**, *64*, 3021−3033.
(13) Torren-Peraire, P.; Hassen, A. K.; Genheden, S.; Verhoeven, J.; Clevert, D.-A.; Preuss, M.; Tetko, I. V. Models Matter: the impact of single-step retrosynthesis on synthesis planning. *Digital Discovery* **2024**, *3*, 558−572.
(14) Maziarz, K.; Tripp, A.; Liu, G.; Stanley, M.; Xie, S.; Gaiński, P.; Seidl, P.; Segler, M. H. S. Re-evaluating retrosynthesis algorithms with Syntheseus. *Faraday Discuss.* **2025**
(15) Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* **2015**.
(16) Park, W.; Kim, D.; Lu, Y.; Cho, M. Relational knowledge distillation. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019; 3967−3976.
(17) Tay, Y.; Dehghani, M.; Bahri, D.; Metzler, D. Efficient transformers: A survey. *ACM Computing Surveys* **2023**, *55*, 1−28.
(18) Eigen, D.; Ranzato, M.; Sutskever, I. Learning factored representations in a deep mixture of experts. *arXiv preprint arXiv:1312.4314* **2013**
(19) Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q.; Hinton, G.; Dean, J. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538* **2017**
(20) Fedus, W.; Zoph, B.; Shazeer, N. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *J. Mach. Learn. Res.* **2022**, *23*, 5232−5270.
(21) Lee, J.; Lee, Y.; Kim, J.; Kosiorek, A.; Choi, S.; Teh, Y. W.Set transformer: A framework for attention-based permutation-invariant neural networks. *International conference on machine learning* 2019, 3744−3753.
(22) Wang, S.; Li, B. Z.; Khabsa, M.; Fang, H.; Ma, H. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768* **2020**
(23) Jaegle, A.; Gimeno, F.; Brock, A.; Vinyals, O.; Zisserman, A.; Carreira, J. Perceiver: General perception with iterative attention. *International conference on machine learning*. 2021; 4651−4664.
(24) Jaegle, A.; Borgeaud, S.; Alayrac, J.-B.; Doersch, C.; Ionescu, C.; Ding, D.; Koppula, S.; Zoran, D.; Brock, A.; Shelhamer, E.et al.Perceiver io: A general architecture for structured inputs & outputs. *arXiv preprint arXiv:2107.14795* **2021**
(25) Andronov, M.; Andronova, N.; Wand, M.; Schmidhuber, J.; Clevert, D.-A. Accelerating the inference of string generation-based chemical reaction models for industrial applications. *arXiv preprint arXiv:2407.09685* **2024**
(26) Saigiridharan, L.; Hassen, A. K.; Lai, H.; Torren-Peraire, P.; Engkvist, O.; Genheden, S. AiZynthFinder 4.0: developments based on learnings from 3 years of industrial application. *J. Cheminform.* **2024**, *16*, 57.
(27) Sterling, T.; Irwin, J. J. ZINC 15−ligand discovery for everyone. *J. Chem. Inf. Model.* **2015**, *55*, 2324−2337.
(28) Genheden, S.; Norrby, P.-O.; Engkvist, O. AiZynthTrain: robust, reproducible, and extensible pipelines for training synthesis prediction models. *J. Chem. Inf. Model.* **2023**, *63*, 1841−1846.
(29) Lowe, D. *Chemical reactions from US patents (1976-Sep2016)*. 2017; https://figshare.com/articles/dataset/Chemical_reactions_from_US_patents_1976-Sep2016_/5104873.
(30) eMolecules, *Chemical Building Blocks: Chemical Scaffolds*. 2023; https://www.emolecules.com/products/building-blocks, [Accessed 01−01−2023].
(31) Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of chemical information and computer sciences* **1988**, *28*, 31−36.
(32) Mendez, D.; Gaulton, A.; Bento, A. P.; Chambers, J.; De Veij, M.; Félix, E.; Magariños, M. P.; Mosquera, J. F.; Mutowo, P.; Nowotka, M.; et al. ChEMBL: towards direct deposition of bioassay data. *Nucleic acids research* **2019**, *47*, D930−D940.
(33) Bjerrum, E.; Rastemo, T.; Irwin, R.; Kannas, C.; Genheden, S. PySMILESUtils−Enabling deep learning with the SMILES chemical language. *ChemRxiv*. **2021**
(34) Wiseman, S.; Rush, A. M. Sequence-to-sequence learning as beam-search optimization. *arXiv preprint arXiv:1606.02960* **2016**
(35) Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461* **2019**
(36) Varuna Jayasiri, N. W. *labml.ai Annotated Paper Implementations*. 2020; https://nn.labml.ai/.
(37) Paszke, A.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 8024−8035.
(38) Falcon, W.; *The PyTorch Lightning team, PyTorch Lightning*. 2019; https://github.com/Lightning-AI/lightning.
(39) Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A next-generation hyperparameter optimization framework. *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*; Association for Computing Machinery: New York, NY, United States, 2019; 2623−2631.
(40) Bergstra, J.; Bardenet, R.; Bengio, Y.; Kégl, B. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*; Curran Associates, Inc., 2011, *24*.
(41) Bergstra, J.; Yamins, D.; Cox, D. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. *Int. Conf. Mach. Learn.* **2013**, *28*, 115−123.
(42) Westerlund, A. M.; Barge, B.; Mervin, L.; Genheden, S. Data-driven approaches for identifying hyperparameters in multi-step retrosynthesis. *Mol. Inform.* **2023**, *42*, No. e202300128.
(43) Woltz, D. J.; Was, C. A. Availability of related long-term memory during and after attention focus in working memory. *Memory & Cognition* **2006**, *34*, 668−684.
(44) Yang, Z.; Cui, Y.; Chen, Z.; Che, W.; Liu, T.; Wang, S.; Hu, G. TextBrewer: An Open-Source Knowledge Distillation Toolkit for Natural Language Processing. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*; Association for Computational Linguistics, 2020; 9−16.
(45) Budennyy, S.; Lazarev, V.; Zakharenko, N.; Korovin, A.; Plosskaya, O.; Dimitrov, D.; Akhripkin, V.; Pavlov, I.; Oseledets, I.; Barsola, I.; et al. Eco2ai: carbon emissions tracking of machine learning models as the first step towards sustainable ai. *Dokl. Math.* **2022**, *106*, S118−S128.
(46) Genheden, S.; Bjerrum, E. PaRoutes: towards a framework for benchmarking retrosynthesis route predictions. *Digital Discovery* **2022**, *1*, 527−539.
(47) Genheden, S.; Shields, J. D. A Simple Similarity Metric for Comparing Synthetic Routes. *Digit. Discov.* **2025**
(48) Thakkar, A.; Kogej, T.; Reymond, J.-L.; Engkvist, O.; Bjerrum, E. J. Datasets and their influence on the development of computer assisted synthesis planning tools in the pharmaceutical domain. *Chemical science* **2020**, *11*, 154−168.
(49) Coley, C. W.; et al. A robotic platform for flow synthesis of organic compounds informed by AI planning. *Science* **2019**, *365*, No. eaax1566.
(50) Cho, J. H.; Hariharan, B. On the efficacy of knowledge distillation. *Proceedings of the IEEE/CVF international conference on computer vision*; IEEE: Seoul, Korea, 2019; 4794−4802.
(51) Tu, H.; Shorewala, S.; Ma, T.; Thost, V. Retrosynthesis prediction revisited. *NeurIPS 2022 AI for Science: Progress and Promises*. 2022.

https://doi.org/10.1021/acs.jcim.4c01821
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX