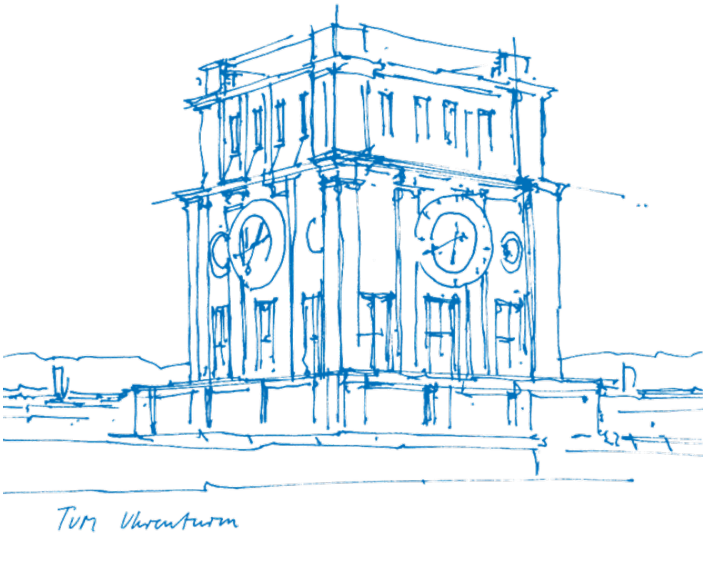
Technical University of Munich
TUM School of Natural Sciences

Prediction of yields of chemical reactions

Varvara Voinarovska



Technische Universität München
TUM School of Natural Sciences

Prediction of yields of chemical reactions

Varvara Voinarovska

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

Doktorin der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitz:
Prof. Dr. Bernd Reif

Prüfer der Dissertation:
1. Prof. Dr. Michael Sattler
2. Priv.-Doz. Dr. Igor Tetko

Die Dissertation wurde am 20.09.2024 bei der Technischen Universität München eingereicht und durch die TUM School of Natural Sciences am 14.10.2024 angenommen.

To all people searching for themselves

Abstract

This dissertation investigates the transformative potential of Machine Learning (ML) techniques in optimizing the economic efficiency of chemical reactions, with a particular focus on enhancing reaction yields. Yield, a fundamental chemistry metric, encapsulates a reaction's efficiency by measuring the output of the desired product relative to all resource inputs. Reactions that result in low yields could hurdle a bigger synthesis, drastically lowering the overall yield of the final product. Hence, improving yield prediction is crucial for advancing synthetic chemistry.

The primary objective of this research is to develop robust quantitative and qualitative models that can accurately predict yields for a diverse array of well-defined chemical reaction types. This study aims to bridge the gap between theoretical potential and practical application in synthetic chemistry by leveraging comprehensive yield data sourced from industry leaders AstraZeneca and Enamine. The research identifies prevalent reaction types with accessible yield information to ensure a broad and representative dataset.

A significant milestone in this project is the compilation of this diverse dataset, which forms the foundation for developing sophisticated classification and regression models. These models are designed to predict reaction yields within a precise range of 0% to 100%, thereby providing valuable insights into the efficiency of different synthetic pathways. This predictive capability aims to improve current practices and facilitate the discovery of new, more efficient synthetic routes.

This study explores various representations of chemical reactions to enhance model performance and accuracy. Among these, the Simplified Molecular Input Line Entry System (SMILES) notation and reaction fingerprints are employed to capture the intricate details of chemical transformations. These representations serve as the input features for the ML models, enabling a more nuanced understanding of the factors influencing reaction yields.

Beyond the development of predictive models, this research also addresses the broader context of yield prediction in chemistry. It examines the inherent challenges associated with current yield prediction methodologies, such as data quality, reaction conditions variability, and limitations of existing modeling techniques. By identifying these issues, the study provides solutions and sets the stage for future research in the field.

This dissertation aims to optimize chemical synthesis processes and facilitate more informed decision-making in synthetic chemistry by integrating advanced ML techniques with rich, curated datasets. By improving our ability to predict reaction yields, this research contributes to the broader goal of making chemical synthesis more efficient, cost-effective, and sustainable.

Zusammenfassung

Diese Dissertation befasst sich mit dem transformativen Potenzial von Techniken des maschinellen Lernens (ML) bei der Optimierung der wirtschaftlichen Effizienz chemischer Reaktionen, wobei der Schwerpunkt insbesondere auf der Verbesserung der Reaktionsausbeute liegt. Die Ausbeute, eine grundlegende Kennzahl der Chemie, fasst die Effizienz einer Reaktion zusammen, indem sie die erhaltene Menge an angestrebtem Produkt im Verhältnis zu allen Ressourceneinsatzen misst. Reaktionen, die zu geringen Ausbeuten führen, könnten eine größere Synthese behindern und die Gesamtausbeute des Endprodukts drastisch senken. Daher ist die Verbesserung der Ausbeutevorhersage für die Weiterentwicklung der synthetischen Chemie von entscheidender Bedeutung.

Das Hauptziel dieser Forschung ist die Entwicklung robuster quantitativer und qualitativer Modelle, die die Ausbeute für eine Vielzahl gut definierter chemischer Reaktionstypen genau vorhersagen können. Diese Studie zielt darauf ab, die Lücke zwischen theoretischem Potenzial und praktischer Anwendung in der synthetischen Chemie zu schließen, indem umfassende Ausbeutedaten der Branchenführer AstraZeneca und Enamine genutzt werden. Die Forschung identifiziert vorherrschende Reaktionstypen mit zugänglichen Ausbeuteinformationen, um einen breiten und repräsentativen Datensatz sicherzustellen.

Ein wichtiger Meilenstein in diesem Projekt ist die Zusammenstellung dieses vielfältigen Datensatzes, der die Grundlage für die Entwicklung ausgefeilter Klassifizierungs- und Regressionsmodelle bildet. Diese Modelle sind darauf ausgelegt, Reaktionsausbeuten in einem präzisen Bereich von 0% bis 100% vorherzusagen und so wertvolle Einblicke in die Effizienz verschiedener Synthesewege zu liefern. Diese Vorhersagefähigkeit zielt darauf ab, aktuelle Praktiken zu verbessern und die Entdeckung neuer, effizienterer Synthesewege zu erleichtern.

Diese Studie untersucht verschiedene Darstellungen chemischer Reaktionen, um die Leistung und Genauigkeit des Modells zu verbessern. Dazu werden die SMILES-Notation (Simplified Molecular Input Line Entry System) und Reaktionsfingerabdrücke verwendet, um die komplizierten Details chemischer Transformationen zu erfassen. Diese Darstellungen dienen als Startwerte für die ML-Modelle und ermöglichen ein differenzierteres Verständnis der Faktoren, die die Reaktionsausbeuten beeinflussen.

Über die Entwicklung von Vorhersagemodellen hinaus befasst sich diese Forschung auch mit dem breiteren Kontext der Ausbeutevorhersage in der Chemie. Sie untersucht die inhärenten Herausforderungen, die mit aktuellen Methoden zur Ausbeutevorhersage verbunden sind, wie Datenqualität, Variabilität der Reaktionsbedingungen und Einschränkungen bestehender Modellierungstechniken. Durch die Identifizierung dieser Probleme bietet die Studie Lösungen und bereitet den Boden für zukünftige Forschung auf diesem Gebiet.

Ziel dieser Dissertation ist es, chemische Syntheseprozesse zu optimieren und fundiertere Entscheidungen in der synthetischen Chemie zu ermöglichen, indem fortschrittliche ML-Techniken mit umfangreichen, kuratierten Datensätzen integriert werden. Indem diese Forschung unsere Fähigkeit verbessert, Reaktionsausbeuten vorherzusagen, trägt sie zum umfassenderen Ziel bei, die chemische Synthese effizienter, kostengünstiger und nachhaltiger zu gestalten.

Contents

Abbreviations

AI	Artificial Intelligence
BA	Balanced Accuracy
BEE	BERT-Enriched Embedding
BERT	Bidirectional Encoder Representations from Transformers
DL	Deep Learning
DoE	Design of Experiments
dppf	1,1’-Bis(diphenylphosphino)ferrocene
ECFP	Extended Connectivity Fingerprint
ELN	Electronic Lab Notebook
FPR	False Positive Rate
HTE	High-Throughput Experimentation
LSTM	Long Short-Term Memory
MAE	Mean Square Error
MCC	Matthews Correlation Coefficient
ML	Machine Learning
MPS	Multi-Product Synthesis
NanoSAR	Nano Structure-Activity Relationship (high-frequency synthetic process coupled with biophysical screening)
NLP	Natural Language Processing
PCA	Principal Component Analysis
QSAR	Quantitative Structure-Activity Relationship
RFC	Random Forest Classifier
RFR	Random Forest Regressor
RMSE	Root Mean Absolute Error
ROC AUC	Area Under Receiving Operating Characteristic Curve
RXNFP	[1] Chemical Reaction Fingerprints
Seq2Seq	Sequence-to-sequence
$\mathrm{S_nAr}$	Nucleophilic aromatic substitution

TPR	True Positive Rate
t-SNE	t-Distributed Stochastic Neighbor Embedding
USPTO	United States Patent and Trademark Office Dataset, extracted by Lowe[2]

1 Introduction

Autobots, transform and roll out!

Optimus Prime, *Transformers*

We live in an incredibly complex world where every process is interconnected and influences others. Ecosystems function through vast networks of animals, plants, and bacteria that provide food and energy to each other. Humans are also part of this intricate web, carrying within them the complexity of internal processes that ensure the coordinated functioning of their bodies. This orchestration is managed by the biomachinery within each cell, which signals what the body should do and how best to survive under various circumstances.

Due to the complexity and multi-component nature of this biomachinery, there are many vulnerable points where the system can fail and require intervention. In most cases, our bodies can handle these disruptions through effective self-healing mechanisms developed by evolution. However, there are instances where the failure is so severe that the body cannot cope alone, and medicine and pharmaceuticals become essential.

Humanity has honed the art of healing for thousands of years, evolving from ancient herbal remedies to modern synthetic drugs that target specific proteins. Numerous medicines exist for thousands of diseases; as of 2019, the FDA approved 19,000 small-molecule drugs[3]. Each developed and approved medication has a rich backstory involving hundreds, if not thousands of people, 10-12 years of development, and an excess of $1 billion of dollars spent on it. The process begins with identifying the cause of a disease, pinpointing the malfunctioning protein or disrupted metabolic pathway, and searching for molecules that can address the problem. This involves, for instance, preliminary assays to identify targets and hits, molecular design, *in silico* docking, synthesis of promising candidates, further bioactivity testing, optimization, and, ultimately, trials in animals and humans. These subprocesses belong to the Design-Make-Test-Analyze (DMTA) cycle, which is the main concept in drug discovery, encompassing constructing and testing hypotheses in drug discovery.

Each stage of the DMTA cycle is expensive. To make drugs more accessible, efforts are focused on reducing costs and increasing environmental sustainability at each stage: minimizing animal testing, reducing synthesis and testing requirements, enhancing modeling based on existing data, automating synthesis with minimal waste, and replacing costly assays with *in silico* docking and in general utilize artificial intelligence to reduce wet-lab experiments[4].

Synthesis involves solving numerous challenges related to predicting reaction properties. Key questions include: What is the reaction mechanism? What will the product be, and with what selectivity? Which conditions are best? Which synthetic route should be chosen? What will the reaction yield be, and is it worth the resources? These questions demand effective solutions.

To achieve cost and sustainability goals, the synthesis methods used in the pharmaceutical industry must be as efficient and reliable as possible, leveraging the most promising reactions from fundamental academic research. The viability of a reaction is assessed based on parameters like selectivity and yields for a wide range of substrates. However, some substrates react unpredictably or yield low outputs even with "well-behaved" popular[5] reactions with good selectivity and yield, such as Suzuki-Miyaura coupling, amide coupling, and $\mathrm{S_nAr}$. Such substrates should be avoided in multi-step syntheses to prevent low overall yields. In high-throughput synthesis, avoiding problematic substrates minimizes waste and improves environmental sustainability.

While traditional approaches have relied heavily on chemists' expertise, intuition, and accumulated knowledge, there is an increasing reliance on advanced modeling techniques to predict reaction properties. Like the human body, a chemical reaction is a complex system, albeit with fewer degrees of freedom. Nevertheless, the ability to model its behavior and outcomes, particularly yields, is a significant scientific and computational challenge.

Ab initio modeling of chemical reactions can provide deep insights into reaction mechanisms and outcomes[6]. However, this approach often requires substantial computational power and time, making it impractical for routine use. Consequently, there is a growing trend towards leveraging accumulated data to develop predictive machine learning (ML) models[7]. These models can provide valuable predictions more quickly and efficiently, making them highly beneficial for practical applications in chemical synthesis.

ML in chemical synthesis has advanced in many areas[8], such as retrosynthesis, direct synthesis, and design of experiments, and has now successfully become part of organic chemists' synthesis routines. However, predicting reaction yields remains particularly challenging. The yield of a reaction, which reflects the efficiency with which reactants are converted into the desired product, usually in terms of a percentage of the theoretical chemical conversion, is a critical metric. High yields are essential for minimizing waste, reducing costs, and ensuring the overall efficiency of synthetic processes. Accurate yield predictions can significantly enhance the design of synthetic libraries, reduce the production of undesirable byproducts, improve environmental sustainability, and lower the costs associated with synthesis. Yet, the yield is a number resulting from multidimensional interactions of reactants, reaction conditions, purification methods, and more. This complexity is both the challenge and the beauty of predicting this crucial metric.

1.1 Introduction to the Design-Make-Test-Analyze cycle

The Design-Make-Test-Analyze (DMTA) cycle is a systematic and iterative framework essential for drug discovery and development in the pharmaceutical industry. This iterative methodology ensures candidate compounds' efficient and robust progression from initial concept to clinical evaluation. For an average project, hundreds of DMTA cycles are necessary to improve the potency of weakly active compounds discovered during High-Throughput Screening. The DMTA cycle includes four critical stages—design, synthesis (make), biological evaluation (test), and data analysis—that collectively enable the optimization and validation of potential therapeutic agents. I illustrate it in Figure 1.1.

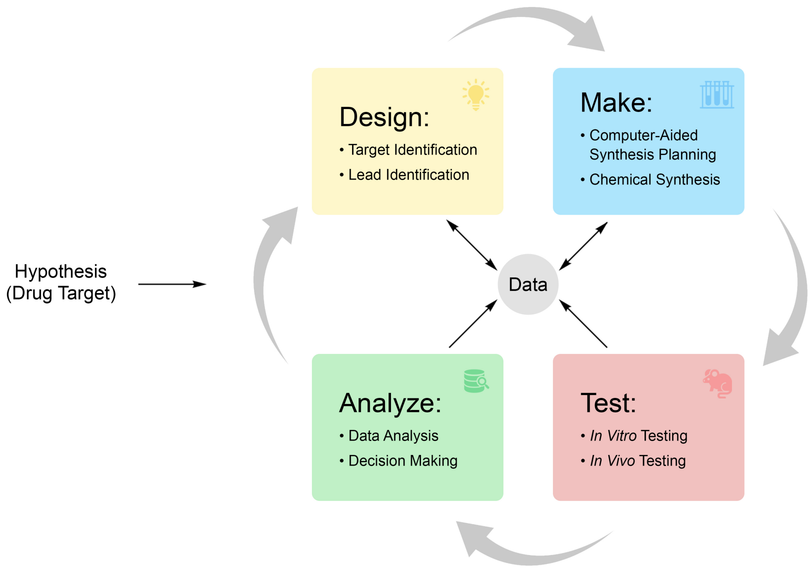


Figure 1.1 Design-Make-Test-Analyze cycle with key steps in each of the stages.

The cycle starts with the target identification, a specific biological target implicated in a disease process, such as a protein, gene, or pathway. This possible druggable receptor or enzyme, a gene, and this target is the subject of the hypothesis of the DMTA cycle. This target could be involved in various diseases, from cancer to metabolic disorders.

Design

The Design stage involves tailoring compound structures to achieve desired target activity and property constraints. This involves computational modeling, structure-based drug design, and/or high-throughput screening (HTS) of chemical libraries to identify potential lead compounds. Historically, this process relied heavily on human intuition, tacit knowledge, and limited historical data, making it one of the most challenging aspects of drug discovery[9]. However, the advent of computational tools and increased data availability have enabled more rational drug design. Today, generative AI tools[10–12] promise to capture complex multi-objective constraints using reinforcement and supervised learning.

Make The Make stage involves the synthesis and purification of compounds. Traditionally, this step relied on the chemists' experience, literature, and information retrieval systems, often leading to a bias toward synthesizing easier-to-obtain compounds rather than the most promising ones[13]. It has several challenges; for example, synthesizing compounds can be complex and resource-intensive. Synthesis planning is also a challenge; rule-based algorithms supported early synthesis planning, but these had limited scope[14]. The emergence of data-driven AI/ML tools for reaction and retrosynthesis prediction has significantly advanced this area[15–17].

Test

The Test stage involves assaying compounds to gather data on their effectiveness and safety. Traditionally, this process was labor-intensive and manual, demanding substantial human effort in sample preparation and assay HTS. Synthesized compounds are tested in laboratory settings using cell cultures or isolated biological components to evaluate their initial efficacy and toxicity, which helps narrow down the most promising candidates. These promising candidates are then evaluated in animal models to assess their safety, efficacy, and pharmacokinetic properties, providing crucial insights into how the compounds behave in living systems. Automation and robotics have alleviated some challenges, enhancing efficiency and accuracy. Additionally, active learning and confidence estimation can prioritize compounds for testing, further improving model accuracy in specific chemical spaces[18].

Analyze

The Analyze stage focuses on studying the data generated from previous stages to prioritize compounds for subsequent DMTA cycles. Early analysis was limited to rudimentary retrieval systems and localized Structure-Activity Relationship (SAR) analysis. However, advanced tools enable comprehensive data analysis in federated environments[19, 20].

In this stage, the results from the testing phase are thoroughly examined to understand the compound's behavior at molecular and biological levels. This includes evaluating its interactions with the intended target, pharmacokinetic profile, and potential off-target effects. Based on this analysis, decisions are made regarding the compound's suitability for further development. This might involve selecting the most promising lead compound for advancement to preclinical and clinical studies or deciding to modify or discontinue development based on the findings.

Throughout the DMTA cycle, the data and insights gained from each stage inform subsequent iterations, allowing for refinement and optimization of the drug development process.

1.2 Detour to Machine and Deep Learning concepts introduction

In this section, I will describe recent computer science breakthroughs employed in the current mainstream of chemoinformatics and help to solve problems of the DMTA cycle.

1.2.1 Transformers architecture

The Transformer architecture, introduced by Vaswani et al.[21] in 2017, is a deep learning model originally used in natural language processing tasks, such as translation and prediction of the sequences. Unlike predecessors' recurrent or convolutional architectures, Transformers process entire sequences of words simultaneously, enabling parallelization and capturing complex dependencies over long distances. This self-attention mechanism allows each word to attend to all other words in the sequence, determining its relevance in the context. By stacking multiple self-attention layers and feed-forward neural networks, Transformers can effectively model intricate linguistic patterns, making them highly effective in various tasks such as machine translation[22], text summarization[23, 24], and sentiment analysis[25, 26]. Newer variations like decoder-only architectures have become foundational in contemporary Large Language Models like ChatGPT and GPT-4[27]. Beyond language processing, analogies between tokens in human language and entities in other domains, such as amino acids in proteins, like in AlphaFold[28], or atoms in chemical reactions, have broadened the Transformer's impact, influencing various scientific disciplines.

The first employment in chemistry in this architecture was in 2018 when Schwaller et al.[29] introduced a Molecular Transformer based on embeddings of a textual representation of molecules called SMILES strings, which were used as sequences and used to predict the products of the reactions.

1.2.2 Graph Neural Network architecture

Graph Neural Networks (GNNs) represent a significant advancement in deep learning, tailored to effectively process and analyze graph-structured data. First proposed in their modern form by Kipf and Welling in 2016[30], GNNs have revolutionized how we approach tasks involving non-Euclidean data structures, such as social networks, molecules, and transportation systems. Unlike traditional neural networks, which struggle with data that lack a regular grid-like structure, GNNs excel by directly operating on graphs, capturing the relational information and dependencies between nodes.

One of the most influential variants of GNNs is the Message Passing Neural Network (MPNN), introduced by Gilmer et al. in 2017[31]. MPNNs work by iteratively passing and aggregating messages between nodes, allowing each node to update its state based on its neighbors' features. This message-passing mechanism enables the network to learn complex patterns and representations of the graph's structure, making it particularly effective for tasks like node classification, graph classification, and link prediction.

The versatility and effectiveness of GNNs have led to the development of various specialized architectures and techniques. For example, Graph Convolutional Networks (GCNs)[30] and Graph Attention Networks (GATs)[32] extend the basic GNN framework by incorporating convolutional operations and attention mechanisms, respectively. These innovations further enhance the ability of GNNs to model complex dependencies and improve their performance on a wide range of tasks.

The pioneering work by Duvenaud et al.[33] demonstrated the potential of graph-based neural networks in predicting molecular properties, setting the stage for numerous subsequent advancements. GNNs can leverage their inherent ability to capture the intricate relationships within a molecule by representing molecules as graphs, where atoms are nodes and bonds are edges.

1.2.3 Reinforcement Learning and Bayesian Optimization

Reinforcement learning[34] (RL) deals with planning and sequential decision-making problems. In RL, an agent interacts with an environment to gather data, learn about the environment, and perform actions to maximize a long-term objective. Each time the agent takes an action, the environment transitions to a new state, which the agent can observe either fully or partially. The agent also receives a short-term reward based on the action taken. Reinforcement learning is used for molecular design, with the most notable work on that being the Olivecrona et al.[35] using the REINFORCE[36] algorithm and subsequent improvement of the work resulting in REINVENT[10].

Bayesian optimization (BO) is a sequential model-based optimization technique that efficiently explores and exploits the search space to find the optimal solution. For example, if a chemist is performing experiments to optimize the temperature and pressure conditions for a chemical reaction to achieve the highest yield. BO balances exploration (trying out uncertain or less-known conditions, such as a new temperature range) with exploitation (focusing on conditions that have shown promising results, like a specific pressure that previously led to high yields). It does this by iteratively updating a probabilistic surrogate model (a mathematical approximation of how the reaction yield changes with temperature and pressure) and using this model to guide the search toward the most promising conditions. Because of this approach, BO often outperforms expert practitioners who might rely on trial-and-error or intuition, as well as other advanced global optimization algorithms [37, 38].

BO is applied to the design of experiments (DoE). DoE aims to sample conditions that help model reaction parameters and understand interactions. Combined with a response surface model, DoE uses knowledge from previous experiments to guide the selection of future ones. It can be applied to diverse search spaces, including parameterized reaction domains, and allows for the selection of multiple parallel experiments, making it ideal for optimizing chemical processes. Bayesian optimization can use the information on existing experiments to propose a new set of conditions to evaluate, reaching the most optimal set of conditions with the highest yield in fewer iterations[39].

1.3 Intro to the Computer-Aided Synthesis Planning

Yield prediction is only a small part of the possible ML application within synthesis predictions, and to be more within the context, we will need to step back and point out the exact location of this problem within the whole field of applying ML for synthesis prediction.

1.3.1 Retrosynthetic prediction

If we were to look at the general process of development of a drug in the industry, we would see that inside of the DMTA cycle, the "Make" part of the synthesis is reliant on the concept Computer-Aid Synthetic Planning, a term coined by Corey in 1985[40], which describes algorithmic procedures on the retrosynthesis process, formalized by him earlier in 1967[41], which approaches the problem of synthesis in reverse: starting with the end compound and proceeding to the accessible starting materials via feasible bond disconnections in the end compound. The first approach to utilize computer algorithms in retrosynthesis problems resulted in developing the LHASA program[42] that was designed to automate the retrosynthetic analysis process, which involves working backward from a target molecule to identify feasible pathways for its synthesis, as illustrated in Figure 1.2. The program employs a combination of logical rules and heuristic strategies to propose retrosynthetic disconnections, effectively breaking down the target molecule into simpler precursor compounds. AI-powered synthesis planning tools aid chemists in enhancing their synthetic chemistry expertise by suggesting feasible synthetic pathways. Additionally, these tools enable chemists to make informed decisions, thereby enhancing efficiency and productivity through minimizing synthesis failures[43, 44].

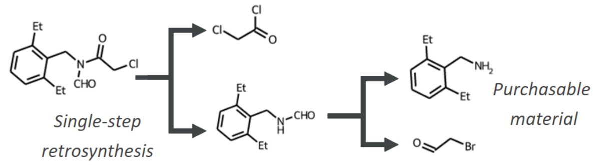


Figure 1.2 The scheme illustrates the retrosynthetic path of the molecule towards the purchasable materials.

Retrosynthesis as a problem has two distinct categories: single-step disconnection prediction and multi-step disconnection prediction, which produces a tree of disconnections. Single-step route-planning strategies are generally categorized into two main types: rule or template-based methods and template-free methods. Rule-based methods use manually coded rules and heuristics derived from reaction databases and literature to propose synthetic routes. In this approach, reaction rules are extracted and encoded manually. For instance, Synthia/Chematica[45–47] is a retrosynthetic software utilizing a library of expert-encoded rules for chemical synthesis planning. However, a drawback of rule-based methods is their limited scalability with the ever-growing chemical literature, leading to incomplete coverage in their knowledge base.

Automated rule-based methods have emerged to overcome these limitations, leveraging computational techniques to extract reaction rules from datasets. These automated methods utilize template extraction algorithms, relying on atom-mapped reaction examples represented as SMIRKS[48] patterns to extract transformations from reaction datasets. Despite their effectiveness, such approaches face challenges, including high computational costs associated with subgraph isomorphism calculations and a lack of chemical intelligence[45, 49].

An alternative rule-based approach employs data-driven DL techniques. For instance, Segler et al.[50] pioneered a neural-symbolic approach to autonomously extract retrosynthetic rules from the Reaxys database without expert intervention. These rules were subsequently used for reaction prediction in conjunction with a modern Monte-Carlo tree search algorithm to identify promising retrosynthetic steps. This data-driven approach promises to overcome the scalability and knowledge limitations of traditional rule-based methods.

A distinct approach from traditional rule-based methods involves employing template-free methods for reaction prediction and retrosynthetic transformations. These methods draw parallels from NLP and frame forward or retrosynthetic prediction as a neural machine translation task[51]. Molecules, represented as SMILES[52] strings, are likened to sentences, treating chemical reactions as linguistic translation challenges.

Liu et al.[53] pioneered the template-free model for retrosynthetic analysis, introducing a Seq2Seq model leveraging encoder–decoder-based model. This model maps SMILES representations of reactants to corresponding product representations bidirectionally, employing bidirectional LSTM cells with additive attention mechanisms for token-wise alignment. Their findings revealed that this model performs comparably to rule-based expert systems in retrosynthetic reaction prediction tasks. Other template-free methodologies, such as graph-based, chemical reaction networks, and similarity-based approaches, have also shown promising outcomes. With the development of Transformers, this model became the most popular model in retrosynthesis with the pioneering work of Karpov *et al*[54].

Several notable retrosynthetic planning tools have implemented these techniques, including AiZynthFinder[17] and the Chemistry42TM[55] platform. ASKSOS[56] and IBM RXN for Chemistry[57] are open web services for retrosynthesis prediction. Notably, in 2018, Klucznik et al. [46] disclosed the first successful execution of a multistep synthesis route proposed by Synthia synthesis planning software, where they designed synthetic pathways for eight diverse and challenging target molecules using the Synthia software.

Compared to single-step retrosynthesis, multi-step retrosynthesis focuses on developing novel route search algorithms, often utilizing a fixed single-step model to identify retrosynthetic disconnections. Pioneering efforts in this field employ neural-guided Monte-Carlo Tree Search (MCTS)[58] and template-based approaches to map synthesis routes. More recent strategies adopt a template-free model by merging neural-guided MCTS with reaction feasibility heuristics[59] or directly applying synthesizability heuristics in conjunction with a forward synthesis model[60].

Multi-step retrosynthesis repeatedly utilizes the chemical information embedded in single-step retrosynthesis models. However, current research tends to treat single-step and multi-step tasks as separate entities, even though multi-step algorithms rely on single-step models, which are generally fixed. Likewise, single-step models are often developed without considering their application in multi-step processes. This gap between single-step and multi-step retrosynthesis was investigated in the work of Hassen and Torren-Perraire[61].

This field is rapidly developing with multiple perspectives already available[7, 8, 44].

1.3.2 Forward prediction

Predicting reaction outcome, a counterpart to retrosynthesis is an equally important task since these are two sides of the same coin.

Historically, starting from the 1980s, this task was performed with methodologies such as physical simulation of transition states, rule-based expert systems, and inductive learning methods. Expert systems, like CAMEO[62, 63], use empirical rules derived from literature to predict reaction outcomes by analyzing mechanistic reasoning and reaction intermediates.

Ugi et al.[64] developed formal techniques, which incorporated in the Interactive Generation of Organic Reactions (IGOR) tool, and they were notable for their ability to predict new reaction mechanisms, verified experimentally by Herges and colleagues[65, 66]. Gasteiger's team[67, 68] developed a system for predicting reaction courses, from isomer generation to absolute rate constants. Zefirov et al. developed the SYMBEQ[69] program, which employs a formal-logical approach, further contributing to the predictive capabilities in organic chemistry. Chen and Baldi's [70] developed an expert system that enhanced reaction predictions with over 1500 manually composed reaction transformation rules, enabling detailed mechanistic predictions, retrosynthetic analysis, and combinatorial library design. Recent Deep Learning applications could be broadly categorized into template-based, graph-edit-based, and sequence-based approaches. Kayala et al.[71, 72] constructed an in-house dataset of elementary reactions and developed a neural network model to predict reactions by identifying electron sources and sinks using a dataset of 11,000 elementary reactions. Wei et al.[73] treated reaction prediction as a classification problem, training molecular fingerprints to predict reaction templates. Segler et al.[74] scaled this approach, ranking reaction rules from a large database and using summed reactant fingerprints. Coley et al.[15] addressed the issue of multiple product matches from templates by ranking these products. The specificity of templates versus the number of templates extracted is a key trade-off in template-based methods.

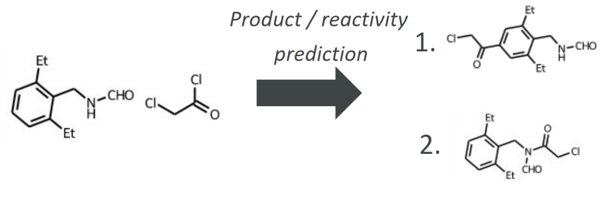


Figure 1.3 The scheme illustrates the prediction of the products of a reaction of aminoacetaldehyde with choloroacetyl chloride. It shows two possible outcomes based on the chloroacetyl chloride functional nature: Friedel-Crafts reaction in 1 and amide formation in 2.

Graph-based approaches introduced by Jin et al.[75] and others[76, 77] use graph convolutional neural networks to predict bond changes without specifying the size around the reaction center. Subsequent advancements included gated graph neural networks[78] and graph transformation[79] policy networks.

Sequence-based approaches, which represent reactions as text using SMILES notation, have shown promise[80], with models like the Molecular Transformer[29] achieving high performance. This model eliminates the need to distinguish between reactants and reagents, allowing training on any reaction dataset. Recent studies have integrated symbolic rules with GNNs[81] and developed graph sequence decoders[82], though the Molecular Transformer remains the best-performing approach.

Transfer learning approaches have been explored to extend predictive performance to more challenging reactions and smaller datasets. Pesciullesi et al.[83], and others demonstrated improvements in specific reaction types by training models on related tasks. These efforts include studying regio- and stereoselective reactions[84, 85] and applying advanced models like the Hopfield network[86] for zero and few-shot learning scenarios.

1.3.3 Conditions and reagents prediction

As we see, retrosynthesis provides valuable insights into the feasibility of constructing a molecule from available building blocks. However, while the retrosynthetic design offers a strategy for the synthesis, it lacks the details necessary for practical execution in wet lab experiments. Factors such as reaction conditions (e.g., temperature and duration), specific operational steps tailored to compound classes or reactions, and the selection of solvents and reagents fall within the expertise of domain specialists. Traditionally, chemists rely on their experience or consult literature and reaction databases to determine appropriate procedures. First, Computer-Aided approaches included ab initio calculations, which were employed to predict suitable solvents for reaction[87] and expert systems[88].

As the field moves towards automated synthesis and seeks to enhance synthesis throughput, various research groups have recently developed ML models to automatically suggest experimental conditions. In 2018, Gao et al.[89] developed a neural network trained on millions of organic reactions sourced from Reaxys[90]. Their model could predict the reaction catalyst, up to two solvents, two reagents, and the temperature without depending on specific reaction classes. Walker et al.[91] focused on solvent prediction for five chosen reaction classes, experimenting with three models tailored to this limited chemical space. In Maser et al.[92] work, they approached predicting up to eight labels for four reaction classes, covering factors such as compound identity (metal, ligand, base, additive, and solvent), temperature, and pressure ranges. On the other hand, Vaucher et al.[93] framed their task as predicting the experimental steps a chemist would take in the lab. In addition to forecasting reaction parameters like temperature and duration, their model could anticipate specific operations such as filtration, phase separation, extraction, or the gradual addition of compounds based on precursor and target molecule features. Also, the reagent prediction task could be used to de-noise data and improve predictions on the direct reaction prediction task, as shown in the work of Andronov et al.[94]

Although this aspect of CASP has seen less work than retrosynthesis and product prediction, recent works show growing interest in this topic.

1.3.4 Reaction optimization using ML

Each reaction out of the box, especially when encountering new substrates, is often non-optimized regarding time, temperature, and reagents. A common way to optimize reaction is the grid search approach, which changes one experimental condition at a time while fixing others[95]. The search space for such a problem is usually so vast that optimal conditions are rarely found using traditional grid search. Alternatively, exhaustive exploration of all combinations of reaction conditions through batch chemistry gives a higher likelihood of identifying the globally optimal condition but is laborious and costly. A streamlined and effective framework for optimizing chemical reactions is paramount for academic research and industrial production.

Numerous efforts have been made to utilize automated algorithms to optimize chemical reactions[96]. Jensen at al.[95, 97] applied the simplex method to optimize reactions in microreactors, while Poliakoff et al.[98] developed the stable noisy optimization by branch and fit (SNOBFIT) algorithm for optimizing reactions in supercritical carbon dioxide. Jensen's group[99] also optimized the Suzuki–Miyaura reaction through automated feedback, focusing on discrete variables. Additionally, several studies have focused on optimizing chemical reactions in flow reactors[100]. For instance, Lapkin et al.[101] introduced a model-based design of experiments and a self-optimization approach in flow, while Ley et al.[102] established a Web-based reaction monitoring and optimization system. Furthermore, Bourne et al.[103] developed automated continuous reactors utilizing high-performance liquid chromatography and online mass spectrometry for reaction monitoring and optimization. deMello and colleagues[104] designed a microfluidic reactor for the controlled synthesis of fluorescent nanoparticles, and Cronin's group[105] provided a flow-NMR platform for monitoring and optimizing chemical reactions. Also, the part of reagent prediction includes a wide variety of works that use Bayesian Optimization[106] to improve and make more efficient current reaction frameworks[107–109]. Reinforcement Learning was used to find optimal conditions for four microdroplet reactions in under 30 min[110].

1.3.5 Molecular design

If we step back from the "Make" part of the DMTA cycle, just before the "Design" stage turns into "Make," we would end up in the molecular design, which is not part of CASP, but it is an important area that should be addressed since to build a retrosynthetic path, a target molecule must first be identified. This molecule should be designed with input from an experienced medicinal chemist who can provide an informed design. Expanding the capabilities of medicinal chemists to generate more molecules with desired properties using reinforcement learning is currently an active area of research.

In this section, I will focus on the most novel methods, which include generative models. As this section provides only a broad overview of the state-of-the-art in this field, I will only summarize it.

The use of Reinforcement Learning (RL) in drug discovery began with Guimaraes et al.[111] and Sánchez-Lengeling et al.[112]'s work. This approach uses SMILES, a widely-used string representation of molecules. In this setup, the RL agent's states are partially completed SMILES strings, and the action space consists of selecting the next character to add to the string.

Sánchez-Lengeling et al.[112] introduced the Objective-Reinforced Generative Adversarial Networks for Inverse-design Chemistry (ORGANIC) based on the SeqGAN approach of Yu et al.[113]

In 2017, Olivecrona et al.[35] proposed a different RL method for drug discovery using the same state and action spaces. They used RL to enhance the RNN's likelihood of constructing molecules with desirable properties. They argued that a policy-based approach is more suitable than a value-based approach and used the REINFORCE algorithm for optimal policy learning. The reward functions were based solely on the desirability of the sequences created. Later, the same group developed REINVENT, which incorporates a memory unit in the scoring function to propose a more diverse range of molecules, as detailed in a paper by Blaschke et al. [10]

An alternative to string-based molecular representations is two-dimensional graphs, which offer increased robustness and interpretability of partially constructed graphs as molecular substructures.

You et al.[114] pioneered RL work for graphical molecular construction. They defined the state space of the RL agent as the set of graphs constructible from scaffold subgraphs and the action space as the possible extensions to the existing graph, either by connecting existing nodes or adding additional scaffold subgraphs. This method showed significant improvements in molecular property optimization and targeting compared to earlier approaches.

1.3.6 State of the Art in Modeling Yield

Adapted with permission from "When Yield Prediction Does Not Yield Prediction: An Overview of the Current Challenges"[115]. Copyright 2023 American Chemical Society.

Historically, predicting reaction yields has posed significant challenges. The emergence of the Brønsted[116] and Hammett[117] equations in the 1920s and 1940s marked a milestone in physical organic chemistry, linking reactivity to chemical structures. In the 1980s, chemists began employing basic methods to predict the properties of small organic molecules, with the first application of Neural Networks for Structure-Activity Relationships introduced in 1992[118]. Successes in Quantitative Structure-Activity Relationship (QSAR) techniques using Random Forest and Support Vector Machines characterized the 2000s[119–121].

Classical ML models from the late 1980s to the early 2010s initially imitated chemists' rules for predicting physical properties and reaction outcomes[122]. However, limited computational capabilities hindered their advancement. Yet, by the mid-2010s, advancements in microelectronics spurred the development of sophisticated ML techniques. Notable progress was made by Emami et al.[123] in 2015, utilizing thermodynamics calculations to achieve significant correlations on a small set of compounds. Subsequently, Raccuglia et al.[124] employed a support vector machine-based decision tree to predict reaction success. The public release of over a million reactions extracted from patents in 2016 by Lowe[2] propelled further advancements, culminating in the development of intricate models rooted in cutting-edge Deep Learning methods[8, 125, 126].

The chemical reaction yield prediction can be divided into two categories, closely tied to the scale of data used for modeling. I illustrate these categories in a scheme 1.4.

The first category comprises traditional fingerprint-based methods reminiscent of those used in QSAR modeling for smaller chemical systems. These methods focus on smaller reaction spaces, tailoring models to optimize specific experiments and aiming for precision within a particular context. Benchmark datasets typically used here are High-Throughput Experiments (HTEs). Feature analysis is integral to this approach, as scientists aim to enhance model accuracy and interpretability by identifying crucial features.

In contrast, the second category involves more recent Deep Learning techniques that leverage language models and graph encodings, which are suitable for handling large datasets. These techniques navigate larger datasets and deploy more complex models capable of handling vast volumes of data. The primary objective is to develop a comprehensive general reactivity model to predict yields across various reaction types.

While this classification provides a useful framework for understanding current research trends, it's important to note that there are exceptions. For instance, there are examples of using fingerprint-based approaches on larger datasets and employing DL on HTE data.

Low-data ML & Active Learning

The optimization of chemical reactions via High-Throughput Experimentation often demands significant resources. This has led researchers to investigate alternative strategies, especially active learning, to navigate situations with limited data. These strategies aim to glean maximum insights from such narrow datasets by pinpointing and harnessing the most important and informative features. The datasets derived from a single experimental setup, usually HTE, are referred to by us as "low-data" experiments. Usually, the experiment settings are as such: the number of data points derived from a single experiment does not exceed ten thousand single reactions.

In a pioneering attempt at yield prediction using machine learning, Ahneman et al.[127] tackled the problem on the Buchwald-Hartwig HTE dataset by leveraging multiple density functional theory (DFT) calculated descriptors and a range of ML techniques, including Random Forest and simple Neural Networks, reaching Root Mean Squared Error (RMSE) 7.8% and R^2 value of 0.92 for the best Random Forest Model (RF) for 70/30 train/test random split set. For leave-one-additive-out the average RMSE was 11.3% and R^2 0.83. However, their methodology was later scrutinized by Chuang and Keiser[128], who pointed out potential redundancy and the minimal informational value of the DFT features, especially considering their computational cost since they reached RMSE of 7.9% and R^2 of 0.91 with random features for the same splitting. Despite this criticism, subsequent research by Żurański et al.[129] indicated that DFT features could indeed offer valuable insights into reaction mechanisms and exhibit enhanced generalization across diverse reaction spaces, demonstrating RMSE between 5-25% for leave-one-additive-out approach with RF. Building on this, Sandfort et al.[130] found that a combination of features often outperforms simplistic one-hot encodings, reaching R^2 score of 0.93, while one-hot showed R^2 of 0.89 on 70/30 random split of BH HTE dataset. In another work, Dong et al.[131] studied the importance of specific features in yield prediction using the SHAP (Shapley Additive exPlanations) library in tandem with XGBoost models, and SHAP usage gives an insight into the most important features, such as electronic descriptors of aryls and ligands. Also, the XGBoost model showed a good performance on the BH HTE dataset with a 90/10 random split of RMSE 5.01% and R^2 of 0.97, on the leave-one-additive-out the XGBoost model outperformed RF. Johansson et al.[132] demonstrated that learning just a fraction of the HTE dataset can be enough to achieve high prediction accuracy. They employed various models, including simple neural networks, complex neural networks, random forests, and Bayesian matrix factorization models. The study utilized an uncertainty-based active learning strategy known as Margin and reached an AUROC of 0.9 using only selected 10% of the BH HTE dataset. Prior work on active learning for predicting outcomes of Suzuki coupling was conducted by Eyke et al.[133], although Active Learning was not outperforming random

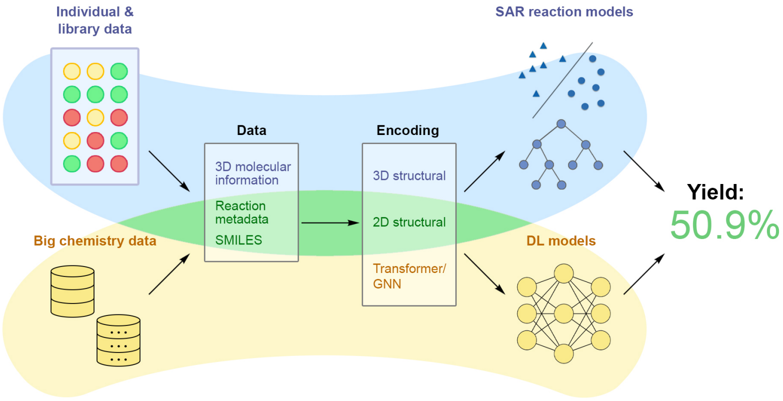


Figure 1.4 Two current State-of-the-Art approaches in yield prediction. The top row illustrates a more classical approach, while the bottom row illustrates the modern approach.

learning until the Active Learning approach had less than 17% of the Suzuki dataset. The authors employed this approach to optimize the number of experiments required to learn the essential features of reactions.

Kexin et al.[134] propose MetaRF, an attention-based random forest model optimized by a meta-learning framework for few-shot yield prediction, and introduce a dimensionality reduction-based sampling method to improve few-shot learning performance. The methodology shows the performance of R^2 of 0.7738 for leave-one-ligand-out and shows R^2 of 0.648 using only selected 2.5% of the BH HTE dataset.

Haywood et al.[135] compared different SVR kernels with different descriptors, including DFT calculated and structural for the BH HTE dataset, and found that structural fingerprints perform slightly better than the DFT ones, with RMSE of 17.4% and R^2 of 0.51 for the structural and RMSE of 23.1% and R^2 of 0.24 for DFT in leave-one-additive-out setting. The authors also attempted to assess the model applicability domain, investigating leave-one-aryl halide-out, leave-one-base-out, and others. They claim that the HTE data needs to be more diverse to allow building a better generalizable model. Using different fingerprints, Bayesian modeling, and the BH HTE dataset as a benchmark, Ranković et al.[136] optimized the selection of additives that lead to higher-yielding reactions. The authors highlighted that employing Bayesian optimization modeling should facilitate the reaction optimization process using HTE. The development of a chemoinformatics workflow for achieving high yields in Buchwald-Hartwig couplings was explored in a study by Fitzner et al.[137]. The investigation focused on developing a new descriptor to reduce the number of experiments necessary for capturing critical information using an active learning approach. To assess the success of the descriptor, they used the Spearman coefficient ρ that takes values between -1 and 1, and their custom XGBoost model reached a value of 0.5. This research also studied the obstacles preventing the achievement of good results in modeling Buchwald-Hartwig C-N coupling reactions.

Reker et al.[138] developed LabMate.ML is a computational framework for leveraging random, unbiased experiments to navigate the selected reactivity space employing adaptive machine learning.

The studies above highlight the active learning strategies employed in yield prediction, the importance of feature selection and engineering, and the efforts to optimize experimental workflows and effectively capture information from limited data for various chemical reactions.

Big-data Deep Learning models

In DL, reactions are typically featurized using either SMILES representation as strings of tokens or molecular graph representation with nodes and edges. "Big data" refers to datasets derived from many experiments of the same reaction type and more general datasets that combine multiple reaction types from diverse sources. These datasets typically contain tens of thousands of data points or more.

Although Transformers[21] using molecules SMILES were successfully employed for molecular property prediction[139, 140], Yield-BERT, developed by Schwaller et al.[125], was a groundbreaking model that successfully implemented the Transformer architecture for yield prediction, reaching R^2 of 0.951 for random 70/30 BH HTE, and RMSE of 12.07% and R^2 of 0.81 for Suzuki dataset on 70/30 random split. Data augmentation played a pivotal role in enhancing the capabilities of Yield-BERT, especially in situations with sparse datasets. This enhancement increased the model's robustness and endowed it with the capacity to assess the uncertainty inherent in yield predictions. In a related study, Baraka et al.[141] employed a Multimodal Transformer-based Model for predicting yields in Buchwald-Hartwig and Suzuki-Miyaura reactions, reaching R^2 of 0.959 for BH HTE on 70/30 random split and RMSE of 5.5 and R^2 of 0.833 for Suzuki and RMSE of 11.5 on 70/30 random split. Their findings emphasized that amalgamating diverse modalities into the prediction process can significantly improve results for these specific chemical reactions. Kojima and Sagava[142] employed the ReactionT5 Transformer model and reached the current state-of-the-art Buchwald-Hartwig HTE dataset with a 70/30 random split with R^2 of 0.927 and RMSE 7.330.

The most widely used frameworks for Deep Learning models that view reactions as graph entities are GNN[143] and MPNN[31]. As an example of this, Sato et al.[144] merged MPNN with self-attention mechanisms for yield predictions, the model resulted in R^2 of 0.972 when using Mol2Vec[145] atom embedding for BH HTE dataset in 70/30 random split. Their work highlighted the importance of particular atoms within the model's calculations. However, their method encountered challenges in predicting outcomes for certain chemotypes within the benchmark datasets. In another study, Youngchun et al.[146] employed MPNN to enable uncertainty-aware learning of reaction yields using the benchmark datasets, introducing the parameter λ, which is responsible for the relative strength of two objectives (minimize the conventional mean squared error and maximization of the log-likelihood over the training dataset). With λ=0.1, the model reached R^2 score of 0.974 for a 70/30 random split for the BH HTE dataset. They have shown that higher predicted variances are often concomitant with higher prediction errors, which provides a criterion to selectively dismiss certain predictions. In another work, Saebi et al.[147] tested various techniques and reported the YieldGNN. This model performed well on BH HTE data, R^2 of 0.957 for YieldGNN with no chemical features. Nonetheless, its efficacy deteriorated when tested on a chemically diverse dataset from AstraZeneca's Electronic Lab Notebooks (AZ ELN), R^2 of 0.049.

In the context of yield prediction, the Transformer architecture has demonstrated a potential benefit over GNN models. This success opens avenues to explore the interpretability of these networks, in particular, to understand their internal mechanisms of "interpreting" reactions. This was exemplified by the creators of Yield-BERT, where they compared the model's learned attention patterns with reaction mapping[1].

Neves et al.[148] introduced a novel technique that augmented the Transformer model standard SMILES encoding with reaction equivalents. Their investigation demonstrated the potential advantages of using this approach to improve industrial synthesis operations. Their methodology employed a binary classification, where reactions yielding 5% or less were labeled as unsuccessful. Uncertainty estimates were analyzed for both the successful and unsuccessful classes. When validating the model on the internal ReactLake reaction database using a temporal split, it was shown that 52.8% of negative reactions can be correctly flagged and thus experimentally avoided. The overall model's performance was satisfactory, with a recorded ROC AUC value of 0.76 in experimental validation.

Yarish et al. [149] developed the directed message-passing neural network (RD-MPNN) yield prediction models, which they tested on Enamine's proprietary reaction data. Their binary classification model showed a commendable ROC AUC of 0.78. When extended to a ternary classification setting, the model displayed an accuracy of 0.51 across multiple reaction classes. Interestingly, the RD-MPNN's performance was on par with the leading results obtained on the BH HTE benchmark dataset and surpassed other models when tested on the Suzuki dataset, with a coefficient of determination (R^2 0.93 for BH HTE, RMSE 10.35%, R^2 0.86 for the latter). Also, the authors performed the analysis of erroneous predictions. They identified key challenges, including issues associated with product isolation by chromatography and reduced yields due to steric hindrance and competing side reactions.

Jian et al. [126] developed a unique SMILES-based model for yield prediction. Based on a special tokenization procedure, an LSTM-based architecture, and data from USPTO and proprietary sources, they could obtain an RSME of around 20%.

1.3.7 Outlook

All of these fields aim to create a singular, all-inclusive system where a chemist can input the properties of the target molecule, and the machine will suggest molecules corresponding to the desired properties, along with a comprehensive retrosynthetic pathway. This system will outline the necessary reagents and conditions for each synthesis step and predict the possible yield of each direct reaction.

These synthetic paths will be ranked according to the chemist's current priorities. For example, the system might prioritize pathways with the least carbon footprint, emphasizing sustainability and green chemistry principles. Alternatively, it could focus on achieving the highest possible yield to maximize efficiency and reduce waste. Cost considerations can also be integrated, with the system recommending pathways that use the cheapest starting materials available, thereby optimizing the economic feasibility of the synthesis. Additionally, the system can minimize the number of steps in the synthesis to streamline the process and reduce potential points of failure.

This holistic approach speeds up drug discovery and development while helping create more effective and sustainable pharmaceuticals. By automating routine tasks and offering smart insights, chemists can concentrate on innovation and critical decisions, ultimately advancing medicine and healthcare.

1.4 Data

Adapted with permission from "When Yield Prediction Does Not Yield Prediction: An Overview of the Current Challenges"[115]. Copyright 2023 American Chemical Society.

To understand the importance of data in yield prediction, it's essential to examine the data at every stage, from the initial experiment to the final dataset used by a chemoinformatician. Regardless of whether a model is simple or complex, it won't be effective if applied to noisy, poorly prepared data.

In this section, I will discuss the currently available data on this topic, both public and proprietary, methods for generating computer-readable data, sources of noise, and practices for addressing this noise.

Yield prediction requires high-quality, consistent datasets. The best-suited datasets for yield prediction are derived from HTE, ensuring that all experiments are run under consistent conditions. If HTE data is unavailable, a thorough data cleaning procedure is necessary to ensure the data's high quality.

1.4.1 Experimental methods to generate reaction data

While large quantities of reaction data are already available, it's important to highlight some promising experimental methods that help generate high-quality data in today's AI-driven world.

One of the key concepts developed in recent years is the automation of organic synthesis[150] and drug discovery in general[151]. This includes advances in automatic solid and liquid handling, precise dispensing, automatic compound purification using catch-and-release techniques, and autonomous control of reaction parameters such as temperature, pressure, homogeneity, and color. Implementing reaction automation has increased the throughput of compound synthesis and reaction reproducibility by eliminating errors and mishandling from human interaction. HTE is the most important method of generating large amounts of data for benchmarking. Benchmark datasets frequently employed in yield prediction include the Buchwald-Hartwig coupling HTE (Buchwald-Hartwig HTE or BH HTE) dataset[127], the Suzuki coupling HTE dataset[152].

By combining automated synthesis and purification, researchers could generate 14 classes of organic compounds using the Suzuki-Miyaura cross-coupling reaction while recording high-quality reaction data[153].

Further, increasing reaction data generation throughput can also be achieved by lowering the scale of individual experiments. This was exemplified in a study where more than 1500 Buchwald-Hartwig experiments were performed in less than a day using as little as 0.2 mg of starting material per reaction[154]. However, it is crucial to note that the reaction data generated by this method can only be used for predicting reaction feasibility and rough yield estimation, as no isolated yield information can be obtained.

Continuous flow chemistry methods are gaining popularity in the synthesis community. They permit a wider range of reaction types to be performed, such as photo- and electrochemistry, and the use of more reactive intermediates due to the possibilities of *in situ* generation and capture. One method used to quickly generate a diverse range of reactions is segmented flow, where segments of pure solvent separate individual reaction samples in a single flow reactor[152]. This technique allowed more than 5700 Suzuki-Miyaura reactions to be performed and automatically purified over an uninterrupted 4-day process.

The subsequent work demonstrated that a similar approach could be applied to diazonium cross-coupling chemistry and parallelized across 16 reaction channels[155], thus increasing the output of reaction data.

Both batch and continuous flow chemistry methods can be directly coupled with a computer control system to form a closed-loop, autonomous synthesis unit[156]. It was shown that the computer control could directly utilize the generated Suzuki-Miyaura reaction data. As a result of the active learning Design of Experiment (DoE) approach, all the products of interest were obtained in high yield without any human intervention.

ELNs (Electronic Lab Notebooks) are also important when considering a data choice for the research. They provide a structured and secure way to record data, which could be later employed in ML, although very few are available in the public domain. A few datasets are derived from this source, with AstraZeneca 750 ELN on Buchwald-Hartwig reaction[147] as one of them.

1.4.2 Extraction of data from textbooks, patents, articles; and available data

However, with data generation comes the side of accumulating and extracting the already available corpus of data from articles, patents, and other sources.

The biggest reaction data vendors employed daily by chemists are Elsvier with the associated Reaxys database[90] and CAS with associated SciFinder tool[157]. The advantage of these proprietary data vendors is the scale and the fact that the publisher annotates the data. This data is not freely available for ML research and requires a huge subscription fee for usage. Other currently available reaction databases include other commercial products like Pistachio[158], which contains a vast amount of patent data.

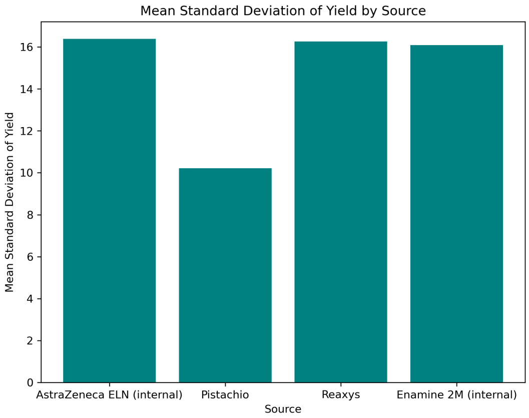


Figure 1.5 The mean yield deviation between the inner data and Reaxys datasets is consistent, but the Pistachio dataset exhibits a lower standard deviation (std) in comparison.

Open Reaction Database (ORD), an open-access initiative [159], was introduced recently, aiming to curate and host reaction data in a format tailored for training machine learning models. A significant feature of this initiative lies in its potential as a hub for sharing industry-specific datasets, which might otherwise stay confined and not be accessible to the broader scientific community. The most presented dataset in ORD is the US Patent Office (USPTO) extracted dataset[2]. USPTO dataset is gathered by text-mining patents from the United States, covering publications from 1976 to September 2016, and therefore encapsulates sparse and diverse chemical reaction data. Also, this initiative database contains other important community datasets such as Suzuki HTE[152] and Buchwald-Hartwig HTE[127]. Recently, the data from the Pfizer HTE dataset [160], which contains 40K data points on different reactions, including hydrogenations, Buchwald-Hartwig, and Ulmann reactions, was added to ORD. HTE datasets originate from high-throughput screenings that aim at finding the best reaction conditions and represent a comprehensive exploration of many combinations of reaction variables.

The HTE and patent datasets display distinct differences in content and quality. While HTE datasets primarily focus on a specific segment of the chemical reaction space, they provide detailed information on certain reaction templates tested with various selected precursors, such as reactants, solvents, bases, catalysts, etc. On the other hand, reactions found in patents encompass a much wider scope in the chemical landscape, the extent and nuances of which will be further discussed in section 4.

Regarding data quality, HTE datasets can represent reactions and yield measurements carried out using the same analytical equipment, ensuring consistent and high-quality data collection[161]. On the other hand, yields documented in patents and journal papers are measured using various equipment used by different institutions. Moreover, the original patent documentation frequently omits essential details, like certain reagents or specific reaction conditions. The inherent challenges of text mining only add to these issues, often leading to noisy and incomplete datasets. Still, it must be acknowledged that chemists working on individual experiments most likely take more care in the purification and analysis of reactions than the massive work-up required for HTE.

1.4.3 Complexity of chemical reactions as a physical object

Predicting reaction yield is challenging due to the complex interaction of many factors. Organic reactions, in particular, can take different paths under various conditions, leading to a range of products and yields. The most significant factors affecting experimental and recorded yield are listed in Table 1.2.

Determining and reporting reaction yields can vary due to terms like crude yield, isolated yield, conversion yield, and selectivity, each highlighting different aspects of the yield. Isolated yield often appears lower than crude yield because of losses during purification. Conversion yield measures the proportion of reactants turned into desired products, while selectivity indicates how exclusively the desired product is formed. Crude yield, although providing a better estimate of chemical reactivity, can be less accurate due to contaminants and side products. Therefore, choosing the most relevant yield term is crucial for accurately evaluating a chemical reaction.

The research carried out by Murray et al.[175] illuminated the numerous factors that significantly impact the results of chemical reactions. Their results indicated that understanding all the variables influencing a Suzuki reaction for a single pair of reactants would require an astonishing six billion experiments. These findings highlight the deep complexity and challenges scientists face when unraveling the intricate details of chemical reactivity.

To illustrate the inherent noise in using yield as a numeric metric for chemical reactions, I analyzed data from various sources where reactions were performed multiple times, and each experiment was recorded. By examining the mean and standard deviation of yields in these datasets, I aimed to evaluate the feasibility and accuracy expectations of regressive yield modeling. I excluded those with a yield of 0 to focus on successful reactions. Additionally, I removed yield pairs of the form [0.0,*value*], assuming that a zero yield likely indicates small-scale test reactions without product isolation. Values differing by ±1% were also filtered out to account for potential rounding errors. As shown in Fig.1.5, the analysis revealed a standard deviation of approximately 16% across general datasets containing various reaction types. This suggests that general reactivity models face significant data-related challenges, and their root mean square error cannot be expected to be lower than 16% in such cases.

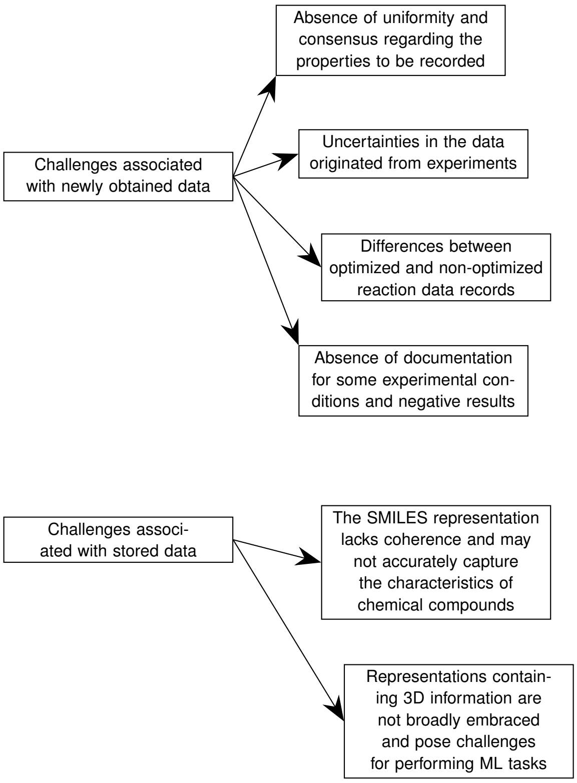

Challenges associated with newly obtained data
Absence of uniformity and consensus regarding the properties to be recorded
Uncertainties in the data originated from experiments
Differences between optimized and non-optimized reaction data records
Absence of documentation for some experimental conditions and negative results
Challenges associated with stored data
The SMILES representation lacks coherence and may not accurately capture the characteristics of chemical compounds
Representations containing 3D information are not broadly embraced and pose challenges for performing ML tasks

Figure 1.6 Main problems that chemoinformaticians are facing when working with chemical datasets.

Overcoming these challenges requires a strong partnership between synthetic chemists and chemoinformaticians. Combining essential knowledge about molecular reactivity, properties of all components,

Dataset	Number of reactions
Synthesis of islatravir by biocatalytic cascade[162]	3
Copper-Catalyzed Enantioselective Hydroamination of Alkenes[163]	3
Development of an automated kinetic profiling system with online HPLC for reaction optimization[164]	7
Coupling of a-carboxyl sp3-carbons with aryl halides[165]	24
Building a Sulfonamide Library by Eco-Friendly Flow Synthesis[166]	39
Microwave-assisted Biginelli Condensation Dataset[167]	48
Deoxyfluorination screen[168]	80
Chemistry informer libraries: a chemoinformatics enabled approach to evaluate and advance synthetic methods[169]	90
Imidazopyridines dataset[170]	384
Linking Mechanistic Analysis of Catalytic Reactivity Cliffs to Ligand Classification[171]	450
AstraZeneca Electronic Lab Notebook (AZ ELN 750)[147]	750
Photodehalogenation HTE[172]	1152
HTE Pd-catalyzed cross-coupling screen[154]	1536
Nano CN PhotoChemistry Informers Library[173]	1728
NiCOlit[174]	1752
Predicting reaction performance in C-N cross-coupling using machine learning (Buchwald-Hartwig HTE)[127]	4312
A platform for automated nanomole-scale reaction screening and micromole-scale synthesis in flow (Suzuki HTE)[152]	5760
HiTEA, Pfizer HTE dataset combined of Ulmann, Buchwald-Hartwig and hydrogenation reactions[160]	39K
Reaxys (non-patents)[90]	~1,3M
USPTO curated from ORD[147]	~1,7M
Pistachio[158]	6,9M

Table 1.1 The table displays datasets with available yield information available for download from ORD[159]. Proprietary datasets not included in ORD are highlighted in bold.

Table 1.2 Factors Influencing Recorded Yield of a Chemical Reaction

Factors Influencing Yield	Explanation
Low Reactivity	Reactants may not fully react, resulting in a low yield of the desired product.
Side Reactions	Other thermodynamically possible reaction paths may be followed, leading to side products and lower yield.
Reactant/Reagent/ Catalyst Deactivation	Deactivation of reactants, reagents, or catalysts caused by other reaction system components.
Thermodynamic and Kinetic Factors	Reaction conditions (temperature, pressure, concentration, etc.) can affect the reaction rate and yield.
Contaminants	Impurities in reactants or reagents can interfere with the reaction and reduce the yield.
Sensitivity to Environment	Reactions may be sensitive to environmental factors like air, moisture, or light.
Product Degradation/ Reactivity	The desired product may be too reactive or unstable, leading to further reactions or degradation.
Product Isolation	Difficulties isolating or purifying the product can result in a lower yield.
Recording Errors	Errors in the process of recording a reaction to ELN or in the steps of incorporation of ELN records to a database.

and their interactions is essential for accurate predictions. The presence of dependable, high-quality data is a fundamental element driving progress in predicting yields for chemical reactions.

1.4.4 Complexity of recording chemical reactions

There is no fixed agreement on the structure of how the experimental or extracted data is recorded and stored. This discrepancy could lead to incorrect recording of reagents, irrelevant yield records, missing conditions of reactions, etc. This section will discuss general incongruities in recording yield and chemical data, which I summarize in Figure 1.6.

I must address that yield data is often incomplete for reported reactions. Only the major product is typically recorded, with side products frequently omitted. Even when side products are included, the distribution may not be normalized to 1. Consequently, much of the reaction data is unsuitable for yield models without extensive preprocessing.

Schwaller et al.[125] noted that the USPTO includes data from sub-gram and gram reaction scales. Lower reaction scales typically indicate "test reactions," preliminary experiments assessing feasibility. In contrast, higher-scale reactions, often called "optimized" reactions, usually involve thoroughly exploring reaction conditions to identify those that yield the maximum product.

Fitzner et al.[176] examined biases and diversity within chemical literature, highlighting the shortcomings in current reaction data. They provided data-driven guides by analyzing over 62,000 Buchwald-Hartwig couplings from multiple databases. These guides recommend reaction conditions and help identify less common ligands that perform optimally when matched with specific substrate properties users choose.

Schleinitz et al.[174] conducted a curated extraction of Ni-catalyzed reactions, emphasizing the importance of thorough data extraction from scholarly articles and optimization tables that support reaction optimization experiments. They also benchmarked various advanced machine learning methods, revealing a clear selection bias in published works and highlighting the significant lack of reported negative data.

In their recent study, Strieth-Kalthoff et al.[177] also examined biases in reported reaction data. They focused on three main sources of bias: experimental errors, experimental selection bias, and result reporting bias. By modeling these biases, they concluded that the interplay between data sparsity and the absence of negative data is the primary constraint on deriving predictive models for chemical reactions.

As emphasized in the editorial by Maloney et al.[178], there is a notable lack of reported negative reaction data. They note that many HTEs conducted in academia often remain inaccessible in machine-readable formats. Additionally, researchers presenting novel reactions in publications frequently neglect to mention unsuccessful trials that contributed to discovering the conditions for successful ones.

Maloney and co-authors propose a more granular differentiation of unsuccessful experiments, dividing them into three specific categories:

- Experiments with neither remaining starting material nor detectable product;
- Experiments where the majority, if not all, of the starting material remains unreacted;
- Experiments not conducted as initially planned.

Having access to such detailed negative reaction data would not only allow for a clearer distinction between unreactive combinations and those that are overly reactive, leading to intricate mixtures but also aid in identifying reactions that deviate from best practices. This would enable a more accurate association between the failed experiments and the systems' inherent reactivity.

The significance of negative reaction data and other experimental details often omitted or inconsistently recorded in conventional publication templates was emphasized in a recent review [179]. Among various considerations, the authors argue that organic synthesis lacks a community-accepted standard for reporting reaction information compared to other domains, such as crystallographic or NMR data. In an initial attempt to address this issue, [180], the authors proposed the XDL markup language format, designed to capture comprehensive experimental details, including the timing of additions, temperature, and standard types of chemical equipment and glassware. Consequently, reaction data reported in this format would be machine-readable and writable, allowing for the post-processing of historical reaction data and generating new data through fully automated synthesis. To facilitate data extraction from the literature and convert it into machine-readable format, Qian et al.[181] and Wilary and Cole[182] introduced tools for automated extraction of reactions and reaction conditions from diagrams and schemes. This tool holds promise in addressing the data extraction challenges previously mentioned. The data gathered over the generation and storage process should be written out in machine-readable format, leading us to discuss how to store and preprocess the data, making it suitable for usage.

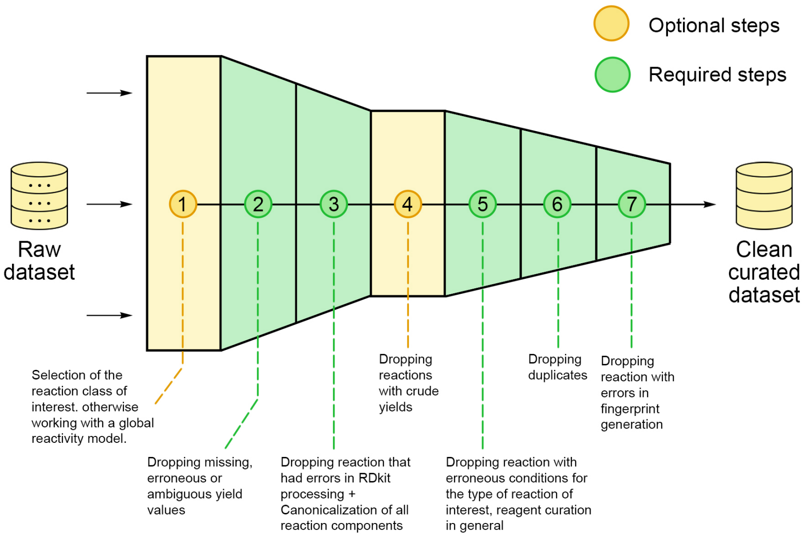

Optional steps
Required steps
Raw dataset
1
2
3
4
5
6
7
Clean curated dataset
Selection of the reaction class of interest. otherwise working with a global reactivity model.
Dropping missing, erroneous or ambiguous yield values
Dropping reaction that had errors in RDkit processing + Canonicalization of all reaction components
Dropping reactions with crude yields
Dropping reaction with erroneous conditions for the type of reaction of interest, reagent curation in general
Dropping duplicates
Dropping reaction with errors in fingerprint generation

Figure 1.7 Steps required in data preprocessing for reactions.

1.5 Data preparation and cleaning

There is no general consensus on reaction data preparation and cleaning; only a few papers discuss it. There is a standard agreement on the preprocessing of molecules[183, 184], but the reactions are more complicated since they have more components and are more complex objects *per se*.

Gimadiev et al.[185] introduced a 4-step protocol for cleaning molecular structures using data from Reaxys, USPTO, and Pistachio. They proposed a general logic for chemical structure curation, transformation curation, and reaction conditions. The curation protocol involves functional group standardization, valence checking, and curation of reaction transformations through reaction balancing and atom mapping methods. However, they did not extend their work to applications like predictive modeling. Their comprehensive overview of the challenges in reaction data standardization highlighted issues such as inaccurate data recording and parsing. Despite the thoroughness of their data curation pipeline, it may be overly broad for specific tasks like predicting reagents or stereochemistry due to its procedures for removing ions, stereochemistry, and radicals.

Genheden et al.[186] released a rxnutils package which is designed to specifically preprocess reactions from end-to-end for retrosynthesis purposes and a possible constructor for your own pipelines for data preprocessing with easy integration of your own classes.

Andronov et al.[94] developed a cleaning pipeline that includes atom mapping, removal of isotope information, and SMILES canonicalization, which was then used to train a transformer model for single-step retrosynthesis.

Coley et al.[187] proposed a holistic strategic approach for designing experimental datasets with modeling applications in mind and including recommendations on maximizing the information gained per experiment. Their work proposes considering the quality of the data and model reactivity qualities with respect to the quality.

The authors of the ORDerly[188] framework proposed a summary and merging of the past approaches, which includes additional frequency filters, removing reactions with too many components, and removing reactions with incorrect yield. Yield prediction requires a more specific procedure involving reaction-specific cleavage, a broad scheme I illustrate in Fig.1.7

1.6 Data representation, data encoding

In this section, I utilize the classification of different molecular and reaction representations described in work by Wigh and colleagues[189].

1.6.1 String/linear representations

There are several ways a molecule could be represented with a string, like molecular formula, generic name, IUPAC name, InChI (International Chemical Identifier), CAS RN, and others. But for the sake of the relevance of my research, I will look in more detail at chemoinformatics-relevant linear representations of molecules and reactions.

SMILES, SMARTS and SMIRKS

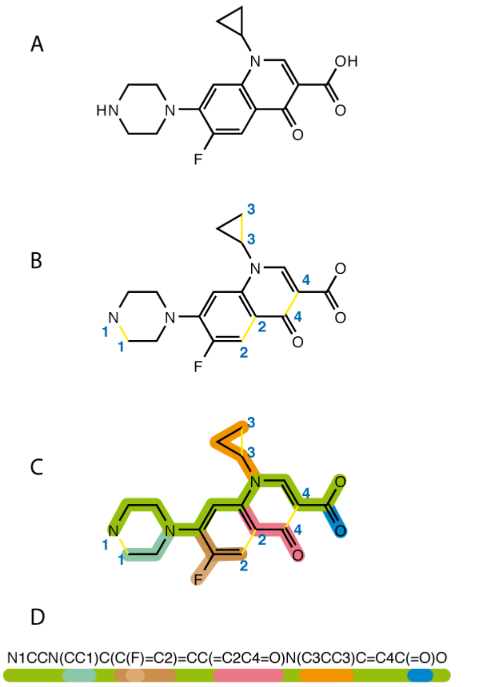


Figure 1.8 Illustration of SMILES of ciprofloxacin, a fluoroquinolone antibiotic. Figure A shows the structure of the antibiotic. Figure B shows the bonds of the cycles to be broken to form a linear string. Figure C highlights substrings that correspond to the same colored substructures in SMILES in Figure D. SMILES require breaking cycling structures to be able to be recorded in a linear way. Image credit: Wikipedia.

SMILES[52] (Simplified Molecular Input Line Entry System) is a text-based notation for recording the structures of chemical compounds and reactions. It encapsulates the same data as an extended connection table but is more flexible due to its linguistic nature. With a simple set of atom and bond symbols and minimal grammar rules, SMILES is a language for storing chemical details and fostering insights. I illustrate the process of the transformation of a chemical structure into the string in Figure 1.8.

SMARTS[190], based on SMILES, specifies structural patterns within chemical compounds. While SMILES represent entire molecular structures, SMARTS define specific features or arrangements, facilitating advanced search and analysis.

SMIRKS[48] or reaction SMILES is a representation of a reaction where "." separate reaction components and reactants are separated from products with "»" with a possibility to place reagents or solvent smiles between the arrows.

The drawbacks of all the line notations described are that they are not unique, and there are ways to describe the same molecule with different strings. Generating agreed unique SMILES is derived from an established set of rules called "canonicalization." Also, SMILES could be syntactically incorrect, which was tried to be tackled with the introduction of SELFIES[191, 192] and DeepSMILES[193].

However, its use comes with inherent challenges, such as non-standardized representations, difficulties in depicting complex metalorganic compounds, and the possibility of generating chemically inconsistent yet technically valid strings. Sodium hydroxide, for instance, can be denoted as $[Na+].[OH-]$. Yet, it could also be represented as $[Na]O$, $NaOH$, $O.[NaH]$ among other possible variants, some of which could be treated as invalid entries in most chemoinformatics packages, such as RDKit[194], for example. These discrepancies can introduce ambiguity and make data preprocessing more complicated.

The limitations of SMILES representation become more apparent in the context of complex entities, for example, transitional metalorganic compounds[195], such as palladium catalysts often employed in Buchwald-Hartwig coupling reactions. Molecules like $Pd(Ph_3P)_2^{2+}$ and $Pd(Ph_3P)_4$ might be erroneously represented in a similar fashion using SMILES, introducing potential discrepancies into the data. In addition, palladium complexes can be denoted in neutral and ionic forms, raising the likelihood of generating incorrect SMILES notations, which can adversely impact the molecular encoding. Moreover, during data storage, SMILES representations of diverse palladium catalyst ligands could mistakenly be classified as duplicates, potentially resulting in unintended exclusions from the final dataset. I visually illustrate their problems in Fig.1.9. Also, for some approaches, reaction SMILES should have the correct mapping of reactant atoms to the product atoms, which is a challenge in the field[59].

1.6.2 Table representations

Among the array of formats available for molecular data storage, 3D formats such as MOL, SDF, and MDL RXN stand out for their level of detail and clarity in representing molecular structures. Yet, despite their detailed nature, they do not enjoy the same widespread acceptance as string-based molecular representations. The need for nontrivial preprocessing further reduces their use in chemistry reaction-related ML tasks.

Table and coordinate formats are widely employed in QM/MM simulations and ML to learn the QM properties of small molecules, such as scalar, vector, or tensor fields[196, 197]. The most suitable models for such natural graph instances are MPNNs; the most prominent example of them is SchNet[198].

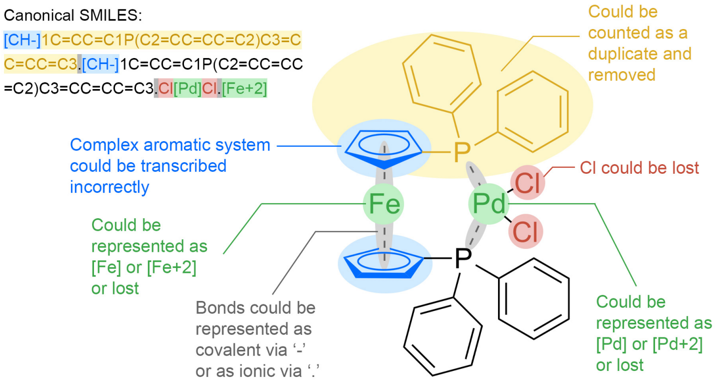

Canonical SMILES:
[CH-]1C=CC=C1P(C2=CC=CC=C2)C3=C
C=CC=C3.[CH-]1C=CC=C1P(C2=CC=CC
=C2)C3=CC=CC=C3.Cl[Pd]Cl.[Fe+2]
Could be counted as a duplicate and removed
Complex aromatic system could be transcribed incorrectly
Cl could be lost
P
Cl
Fe
Pd
Cl
P
Could be represented as [Fe] or [Fe+2] or lost
Bonds could be represented as covalent via '-' or as ionic via '.'
Could be represented as [Pd] or [Pd+2] or lost

Figure 1.9 Illustration of potential inaccuracies in the depiction of molecules using PdCl2(dppf) as an exemplar. This Pd-containing catalyst finds extensive application in diverse couplings, encompassing Suzuki coupling and Buchwald-Hartwig reactions.

1.6.3 Data encoding

The history of fingerprint encoding can be traced back to the 1960s with the creation of the first substructure-based fingerprints, notably the Morgan fingerprints[199] (structurally equivalent to ECFP fingerprints[200]). Over the decades, these substructure-centric fingerprints have retained their prominence, capturing the critical chemical attributes of a compound. More recently developed fingerprints harness the capabilities of Deep Learning models, including GNNs and Large Language Models. This section will discuss fingerprints applied in reaction prediction or reaction yield.

Structural and Molecular Fingerprints

Extended-Connectivity Fingerprints[200](ECFPs) are circular topological fingerprints crafted for molecular characterization, similarity searching, and structure-activity modeling. ECFP fingerprints are generated through a process that circularly captures the molecular structure's connectivity patterns. These fingerprints encode structural features and their connectivity, enabling efficient similarity searching and structure-activity modeling by comparing the presence or absence of specific molecule substructures. RDKit[194] provides a free implementation of these fingerprints under the original name of Morgan fingerprints.

DRFP[201] utilizes circular substructures from molecules and hashes their SMILES representations, drawing inspiration from chemical fingerprints like ECFP and MHFP[202]. DRFP does not incorporate atom-mapping-based weights to distinguish between reactants and reagents, nor does it mandate the calculation of molecular properties for the reagents. DRFP does not involve arithmetic operations on individual molecular fingerprints, such as the atom pair fingerprint.

Kallisto[203] is a framework for efficient and robust generation of atomic features based on geometric molecule input. It offers several important QM-level features such as polarizability, van-der-Waals radii, proximity shells, and others. This framework serves as a bridge between more costly calculations for the whole reaction object (like transitional states) and more simple calculations of substructures. I used it in a framework for generating 3D features for reactants; I describe the methodology in more detail in Section 2.

Section 2 discusses these fingerprints in more detail.

QM-derived fingerprints include multiple QM-accessible properties, a set of which is designed specifically for each problem. These properties include molecular, atomic, and vibrational, including HOMO/ LUMO, polarizability, lengths of important for modeling bonds, etc.

Computer-Learned Fingerprints

CGR, or Condensed Graph of Reaction, is a representation that combines reactants and products into a single 2D graph, encompassing both conventional and changing bonds. Developed by Varnek and colleagues[204], the CGR approach encodes molecular structures using fragment occurrence in a matrix. It offers a superposition of reactant and product molecules, describing alterations in atoms and bonds, reminiscent of transition state concept[205]. This approach has seen increasing adoption in recent cheminformatics research, leading to the creation of an open-source toolkit by Varnek and colleagues to facilitate wider CGR utilization[206]. However, it's worth noting that this approach relies on correct reaction atom mapping, a current challenge in the field.

Chemprop[207] is based on CGR as a transformation from linear to graph representation and then passed to a directed message-passing neural network (D-MPNN) block. The main weakness is that SMILES are not accepted in stereochemistry, and some mapping of reactions is rejected. Section 2 discusses this model in more detail.

CDDD[208] (Continuous and Data-Driven Descriptors) are molecular descriptors derived by unsupervised training on a large dataset of biologically relevant molecules (extracted from ZINC and PubChem) using translation from InChi to SMILES task, using RNN and CNN architecture. The main weakness of these fingerprints is that they can produce fingerprints only for the molecules within a specific application domain. Also, it doesn't accept SMILES with stereocenters.

RXNFP[1] is a molecular fingerprinting method developed by Schwaller that employs a transformer neural network derivative to BERT[23] to represent chemical reactions as numerical vectors. The task of learning was to restore masked SMILES tokens. By training on a Pistachio[158] dataset, RXNFP learns to encode reactions into continuous vector representations, enabling applications such as reaction outcome prediction, reaction classification, and similarity search. Section 2 discusses this fingerprint in more detail.

1.7 Aims

So far, I have discussed the intricacies of yield prediction and the current state-of-the-art models, data, and their place in CASP. To further advance the field, I will use current data to explore and address some critical questions related to yield prediction:

- Is developing a predictive model for specific reaction classes using real-world data feasible?
  This question explores the potential of leveraging extensive datasets from experimental reactions to build robust predictive models. Analyzing patterns and correlations within the data aims to create models that can accurately predict the outcomes of various reaction classes. This involves understanding the underlying chemistry and employing advanced machine-learning techniques to handle the complexity and variability inherent in real-world data.

- How do the intricacies and quality of data influence predictive accuracy?
  The accuracy of predictive models heavily depends on the quality and granularity of the input data. This question addresses how factors such as data completeness, consistency, and the presence of noise impact model performance. It also examines the role of detailed reaction conditions, reagent properties, and intermediate states in refining predictions. Understanding these intricacies can lead to better data curation and preprocessing methods, enhancing the reliability of yield predictions.

- What improvements can be made from both data and modeling perspectives?
  This question aims to identify potential enhancements in data collection, management, and model development. On the data side, improvements might include increasing the volume and diversity of data, standardizing data formats, and integrating data from various sources to enrich the training datasets. From the modeling perspective, advancements could involve developing more sophisticated algorithms that can handle the complexities of chemical reactions, incorporating domain-specific knowledge into models, and improving computational efficiency. Additionally, enhancing the interpretability and explainability of models to provide actionable insights for chemists is a key focus area.

In chapter 4, I discuss how the general reactivity models cannot capture the yield information from real-world data. In chapter 5, I use the BEE model to predict the yield of the Enamine dataset and discuss the current problems with Transformers-like architectures usage in yield prediction. In chapter 6, I will discuss how elaborate data preprocessing and simplifying yield prediction as a classification problem still pose great challenges in data transferability.

2 Methods

It's more fun to compute.

Kraftwerk, *It's more fun to compute*

2.1 Basic Tools

2.1.1 Python Programming language

Python is a high-level, interpreted programming language known for its simplicity, readability, and versatility. It is the most used in research for data analysis, machine learning, and scientific computing.

2.1.2 Pandas package

Pandas is a Python library for data manipulation and analysis. The main objects it operates with are DataFrames (Python object derived from tabular data) and Series (Python object derived from column data). It provides efficient tools for cleaning, exploring, transforming, and analyzing structured data, making it indispensable for data preprocessing, statistical analysis, and visualization in data science projects.

2.1.3 Numpy package

NumPy is a Python library for numerical computing, providing powerful array objects and essential mathematical functions. It enables efficient operations on large multidimensional arrays and matrices, making it essential for numerical simulations, data analysis, and machine learning tasks.

2.2 Machine Learning tools

2.2.1 Scikit-learn Package

Scikit-learn[209], often referred to as sklearn, is a widely-used machine learning library in Python, providing a comprehensive suite of tools for various tasks such as classification, regression, clustering, and dimensionality reduction. It offers simple and efficient tools for data preprocessing, model selection, evaluation, and deployment.

2.2.2 Imblearn

Imbalanced-learn or Imblearn[210] is an open-source package based on Scikit-learn, which provides tools for classification with imbalanced classes. It is useful to use different kinds of over- and undersampling metrics to navigate the model's performance when dealing with imbalanced learning.

2.2.3 Rxnutils package

Rxnutils[186] is a Python package developed by Kannas *et al.* aimed at standardizing chemical reactions. This package comprises various pipeline routines for data preprocessing, such as dropping invalid entries, reaction mapping, etc. I used the 1.7.0 version of this package. For Chapter 6, the following pipeline was used to preprocess the reaction data from AstraZeneca's internal database:

```
query_dataframe:
  query: ~rsmi.isna() and rsmi != ''
remove_unsanitizable:
  in_column: rsmi
  out_column: rsmi_processed
desalt_molecules:
  in_column: rsmi_processed
  out_column: rsmi_processed
detect_reactive_functions:
  in_column: rsmi_processed
  smarts_lib: ${REACTIVITY_SMARTS_LIB}
  rsmi_column: rsmi_processed
  max_reactants: 5
count_components:
  in_column: rsmi_processed
atombalance:
  in_column: rsmi_processed
  out_column: nheavy_atoms_diff
correct_reagents:
  smiles_in_column: reagent_smiles
  ids_in_column: reagent_ids
  roles_in_column: reagent_roles
  corrections_path: ${REAGENTS_CORRECTIONS}
  only_roles: [reagent, catalyst]
sample_reagent_smiles:
  smiles_column: CorrectedReagentsSmiles
  roles_column: CorrectedReagentsRoles
  only_roles: [reagent, catalyst]
  out_column: ReagentCatalystSmiles
```

2.2.4 Random Forest

Random forest[211, 212] is an ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes for classification or the mean prediction for regression. Each tree in the forest is built on a random subset of the training data and uses a random subset of features at each split, which helps to reduce overfitting and improve generalization[213]. By averaging the predictions from many trees, random forest enhances model accuracy and robustness, leveraging the diversity among the individual trees to create a strong overall model. Random Forest is known for its robustness to overfitting, handling of high-dimensional datasets, and capability to capture complex relationships in the data.

2.2.5 Gradient Boosting

Gradient boosting[214] is an ensemble ML technique that builds models sequentially, where each new model attempts to correct the errors made by the previous models. It starts with an initial weak model and iteratively adds additional weak models, typically decision trees, to form a strong predictive model. Each added model is trained to predict the residual errors of the combined ensemble of all previous models, effectively minimizing the loss function through gradient descent in function space. This method is powerful for both regression and classification tasks, often yielding high accuracy and robust performance by combining the strengths of multiple models.

2.2.6 Support Vector Machines

Support Vector Machines[215] (SVM) is a powerful supervised learning algorithm used for classification and regression tasks. SVM works by finding the hyperplane that best separates the data into different classes in a high-dimensional space. The optimal hyperplane is the one that maximizes the margin between the closest points of the classes, known as support vectors. For non-linearly separable data, SVM employs kernel functions to transform the data into a higher-dimensional space where a linear separator can be found. This approach allows SVM to effectively handle complex patterns and achieve high classification accuracy.

2.2.7 Lazy Predict

Lazy Predict is a Python library that automates the preliminary model selection and evaluation by quickly building and comparing multiple machine learning models with default settings. It provides a convenient way to get an overview of how different algorithms perform on a given dataset without requiring extensive manual configuration. Lazy Predict helps users identify promising models for further fine-tuning and optimization, making it a valuable tool for rapid prototyping and initial exploration in machine learning projects.

2.2.8 Hyperparameter optimization

Hyperparameter optimization refers to selecting the best set of hyperparameters for a machine learning model. Unlike model parameters, which are learned during training, hyperparameters are predefined and control aspects of the learning process, such as learning rate, regularization strength, and the number of layers in a neural network. The choice of hyperparameters can significantly impact model performance, making their optimization crucial for achieving the best possible results. This process typically involves exploring the hyperparameter space using techniques like grid search, random search, or more advanced methods like Bayesian optimization to identify the combination that yields the highest validation accuracy or minimizes the loss function.

2.2.9 Optuna

Optuna[216] is an automatic open-source hyperparameter optimization software framework designed to find the best hyperparameters for machine learning models efficiently. It uses a sophisticated search algorithm that combines techniques such as Bayesian optimization, which models the objective function to be optimized and balances exploration and exploitation with other strategies like pruning and parallelization. Optuna's key feature is its flexibility and ease of use, allowing users to easily define a search space and objective function. By dynamically adjusting the search based on past evaluations, Optuna efficiently converges to optimal hyperparameter configurations, enhancing model performance and reducing computational cost. I used it to search for optimal parameters for Random Forest, Here, I present a simplified code snippet illustrating the parameters Optuna optimized while searching for the best Random Forest model.

In the snippet, the following parameters are used: *thresholds* are the thresholds by which the float yield data will be divided into classes.

n_estimators_rfc is the number of trees in Random Forest.

max_depth_rfc is the depth of a single tree in a Random Forest.

max_features_rfc is the number of features to consider when looking for the best split in Random Forest.

```
#Proposing thresholds for yield division
thresholds = [
    trial.suggest_int(f"threshold_{i}", 1, 99) for i in range(num_classes - 1)
]
#Parameters of Random Forest model
n_estimators_rfc = trial.suggest_categorical(
    "n_estimators_rfc", [300, 500, 800]
)
max_depth_rfc = trial.suggest_categorical("max_depth_rfc", [None, 3, 5])
max_features_rfc = trial.suggest_categorical("max_features_rfc", ["sqrt", 200])
```

2.3 Visualization tools

2.3.1 Matplotlib package

Matplotlib[217] is a comprehensive library for creating a variety of visualizations in Python. It offers various plotting functions and customization options, making it suitable for various data visualization tasks.

2.3.2 Seaborn package

Seaborn[218] is a statistical data visualization library built on top of Matplotlib in Python. It provides an easy-to-use interface for creating attractive and informative statistical graphics. Seaborn simplifies generating complex visualizations such as heatmaps, violin, and pair plots.

2.3.3 t-SNE

t-Distributed Stochastic Neighbor Embedding[219] (t-SNE) is a dimensionality reduction technique widely used for visualizing high-dimensional data in lower-dimensional spaces. Originally proposed by Laurens van der Maaten and Geoffrey Hinton in 2008, t-SNE aims to preserve the local structure of the data while revealing its global patterns.

The method works by modeling the high-dimensional data as a probability distribution in both the original high and lower-dimensional space (often two or three dimensions for visualization purposes). It then minimizes the difference between these distributions, typically using the Kullback-Leibler divergence, which is a type of statistical distance: a measure of how one reference probability distribution P is different from a second probability distribution Q. This optimization process effectively maps the data points from the high-dimensional space to the lower-dimensional space, where similar data points in the original space are projected to nearby points in the visualization.

One of the key advantages of t-SNE is its ability to reveal clusters and patterns in the data that might be difficult to discern in the original high-dimensional space. This makes it a powerful tool for exploratory data analysis and visualization, allowing researchers to gain insights into the underlying structure of their data.

2.4 Model evaluation metrics

2.4.1 Confusion Matrix and Binary Classification Metrics

The confusion matrix is a table that summarizes the performance of a classification model by comparing the model's predicted labels with the actual labels. It is commonly used for binary classification but can also be adapted for multi-class classification. The matrix provides counts for:

- **True Positives (TP)** – instances correctly classified as positive.
- **False Positives (FP)** – instances incorrectly classified as positive.
- **True Negatives (TN)** – instances correctly classified as negative.
- **False Negatives (FN)** – instances incorrectly classified as negative.

This layout is the foundation for calculating performance metrics such as accuracy, precision, recall, and F1-score.

	Predicted Positive	**Predicted Negative**
Actual Positive	**True Positive (TP)**	**False Negative (FN)**
Actual Negative	**False Positive (FP)**	**True Negative (TN)**

2.4.2 Multi-Class Classification Metrics

For multi-class classification, the confusion matrix is extended to multiple classes. I use the One vs Rest (OvR) approach, where each class is treated as the "positive" class while all other classes are treated as "negative." Metrics for each class k can then be defined as:

- **True Positives (TP$_k$)**: Instances correctly classified as class k.
- **True Negatives (TN$_k$)**: Instances correctly classified as not belonging to class k (the sum of all other correctly classified instances).
- **False Positives (FP$_k$)**: Instances incorrectly classified as class k.
- **False Negatives (FN$_k$)**: Instances of class k incorrectly classified as another class.

Using the OvR approach allows the calculation of accuracy, precision, recall, and F1-score for each class individually and enables an overall assessment of the model's performance across multiple classes.

2.4.3 Precision

Precision is a metric used in classification tasks to measure the accuracy of positive predictions. It quantifies the proportion of correctly predicted positive cases out of all cases predicted as positive. In multi-class classification, precision can be computed using macro and micro averaging. Macro averaging calculates precision independently for each class and then takes the average, while micro averaging aggregates the contributions of all classes before computing precision, thus giving equal weight to each class.

$$\text{Precision}_{\text{macro}} = \frac{1}{C}\sum_{i=1}^{C}\frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i}$$

$$\text{Precision}_{\text{micro}} = \frac{\sum_{i=1}^{C}\text{TP}_i}{\sum_{i=1}^{C}(\text{TP}_i + \text{FP}_i)}$$

Where C represents a number of classes.

2.4.4 Balanced accuracy

Balanced accuracy is a performance metric commonly used in classification tasks, especially when dealing with imbalanced datasets. It calculates the arithmetic mean of sensitivity (true positive rate) and specificity (true negative rate), providing a balanced assessment of classifier performance across all classes. This metric is particularly useful in situations where class distributions are uneven, ensuring that the evaluation is not skewed by the dominance of one class over the others.

$$\text{Balanced Accuracy} = \frac{1}{C}\sum_{c=1}^{C}\frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c}$$

where:

- C is the total number of classes.
- TP_c is the number of true positives for class c.
- FN_c is the number of false negatives for class c.

2.4.5 ROC-AUC

The Receiver Operating Characteristic Area Under the Curve (ROC-AUC) is a performance metric commonly used to evaluate the discriminatory power of a binary classification model across different thresholds. It quantifies the model's ability to distinguish between positive and negative classes by plotting the true positive rate against the false positive rate, with a higher AUC indicating better classification performance.

$$\text{ROC-AUC} = \int_0^1 \text{TPR}(FPR^{-1}(t))dt$$

2.4.6 F1 Score

The F1 score is a widely used performance metric in binary classification tasks, particularly when dealing with imbalanced datasets. It represents the harmonic mean of precision and recall, providing a single measure that balances the correctness and completeness of the model's predictions. The F1 score ranges from 0 to 1, with a higher value indicating better model performance. It is particularly useful when false positives and false negatives carry different costs, and the goal is to minimize both types of errors.

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

where

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

Here, TP represents true positives, FP false positives, and FN false negatives.

2.4.7 Matthews Correlation Coefficient

Matthews Correlation Coefficient (MCC) is a statistical metric commonly used to assess the quality of classification models, particularly in scenarios with imbalanced class distributions. It considers true positives, true negatives, false positives, and false negatives, providing a balanced measure ranging from -1 to 1, where 1 indicates perfect prediction, 0 indicates random prediction, and -1 indicates total disagreement between prediction and observation. It's particularly robust when classes are of different sizes or when the cost of false positives and false negatives is high and uneven. The MCC for multi-class classification is typically calculated using a confusion matrix.

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

2.4.8 Geometric Mean

The geometric mean metric is a statistical measure commonly used to assess the performance of classifiers, particularly in scenarios with unbalanced class distributions. It calculates the square root of the product of class-wise sensitivity (true positive rate) and specificity (true negative rate), effectively balancing the influence of both classes. This metric is especially valuable when one class dominates the dataset, ensuring that the performance evaluation reflects the classifier's ability to correctly predict instances from all classes, regardless of their prevalence.

$$\text{Geometric Mean} = \sqrt{\text{Sensitivity} \times \text{Specificity}}$$

Where

$$\text{Sensitivity (TPR)} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{Specificity (TNR)} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

2.4.9 MAE and RMSE

Mean Absolute Error (MAE) is a statistical metric commonly employed to evaluate the accuracy of a predictive model. It calculates the average of the absolute differences between predicted and observed values, providing a measure of the model's average prediction error magnitude without considering the direction of errors.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Root Mean Square Error (RMSE) is a statistical measure used to assess the accuracy of a predictive model by quantifying the differences between predicted and observed values. It calculates the square root of the average of squared differences between predicted and actual values, providing a single value indicative of the model's overall performance.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

n represents the number of data points, $y_i$is the true value, $\hat{y}_i$ is the predicted value.

2.4.10 R2 coefficient

The R^2 coefficient, also known as the coefficient of determination, is a statistical measure used to evaluate the goodness of fit of a regression model. It quantifies the proportion of the variance in the dependent variable that is predictable from the independent variables, with values ranging from 0 to 1. A higher r^2 value indicates that the independent variables explain a larger proportion of the variability in the dependent variable.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

Where n is the number of data points, y_i is the true value, $\hat{y}_i$ is the predicted values, and $\bar{y}$ is the mean of the true values.

2.4.11 Hold-out test set evaluation

Hold-out test set evaluation is a basic validation method in which the dataset is split into two non-overlapping subsets: a training set and a test set. The training set is used to train the model, while the test set is reserved for evaluating the model's performance on unseen data. This approach is simpler and faster than cross-validation but may provide a less robust estimate of the model's performance due to the reliance on a single train-test split. Despite this, the hold-out method remains valuable as part of cross-validation, where each fold essentially represents a different hold-out split. Both methods aim to estimate the model's generalization ability, though cross-validation reduces the variance of performance estimates by averaging results across multiple folds.

2.4.12 Cross-validation

Cross-validation is a robust technique used to assess the performance of machine learning models by partitioning the dataset into multiple subsets or folds. In each iteration, the model is trained on a combination of several folds (the training set) and validated on the remaining fold (the validation set). This process is repeated until each fold has been used as a validation set. Cross-validation is essentially an extension of the hold-out method, where the dataset is split multiple times into different train-test combinations, thus mitigating the randomness that can occur in a single hold-out split. This approach helps evaluate the model's generalization ability and reduces overfitting, particularly when the available data is limited. Cross-validation provides a more reliable model performance estimate than a simple hold-out validation.

2.4.13 Train-Validation-Test Splitting

In machine learning and deep learning, datasets are often divided into three distinct sets: a training set, a validation set, and a test set. The training set is used to fit the model, while the validation set is used to monitor the model's performance during training. Rather than the model "seeing" the validation set during training, the validation set provides a means to evaluate the model's performance after each epoch (i.e., each complete pass through the training data). This evaluation helps to tune hyperparameters, select the optimal model configuration, and detect overfitting.

The test set is kept separate throughout the training process and is used for the final model evaluation. The model is evaluated on this test set after training is completed, typically at a selected "checkpoint." A checkpoint is a saved state of the model at a specific point during training, often corresponding to the point at which the model achieves the best validation score. However, it is important to note that the "best" validation score is context-dependent and may refer to the lowest error (e.g., Root Mean Squared Error) or the highest accuracy, depending on the evaluation metric used. This method ensures that the test set is used solely for the final, unbiased performance assessment without influencing the model during training.

2.4.14 Imbalances in classes

If the class distribution is imbalanced, some methods aim to tackle this problem and avoid models to classify all instances as the majorly represented class. For undersampling and oversampling, I used the Imblearn package.

- Undersampling
  Undersampling involves reducing the number of samples from the majority classes to match the minority class. This can help balance the class distribution, but it comes with the risk of losing valuable information from the majority classes, which may affect the model's performance. I used random oversampling, which removes samples from the majority classes until the desired balance is achieved.

- Oversampling
  Oversampling increases the number of samples in the minority classes to balance the dataset without losing information. I used SMOTE (Synthetic Minority Over-sampling Technique)[220],

which generates synthetic samples by interpolating them between existing minority class samples. SMOTE helps to create more diverse and informative synthetic samples, reducing the risk of overfitting compared to simple duplication.

- Class weighting
  Class weighting assigns different weights to classes during model training, giving more importance to the minority classes. This approach adjusts the learning process to be more sensitive to underrepresented classes, often leading to improved performance in these classes. I used the default Scikit-learn implementation class_weight='balanced' provided for the RF classifier. The weight assigned to each class c is calculated as:

  $$w_c = \frac{n_{\text{samples}}}{n_{\text{classes}} \times n_c}$$

  Where:

  - w_c is the weight for class c.
  - n_{samples} is the total number of samples in the dataset.
  - n_{classes} is the total number of unique classes.
  - n_c is the number of samples in class c.

  This penalizes misclassifications of minority classes more heavily, encouraging the model to focus on correctly predicting these classes.

2.4.15 Data scaling

In data preprocessing, it is crucial to scale the feature vectors to ensure that each feature contributes equally to the model and to eliminate biases introduced by the differing scales of the data. This scaling can help improve the performance of various machine learning algorithms, particularly those that rely on distance metrics such as support vector machines.

For this purpose, I used the StandardScaler from the sklearn package. The StandardScaler standardizes features by removing the mean and scaling to unit variance. This process transforms the data such that each feature has a mean of zero and a standard deviation of one, which is often beneficial for many machine learning algorithms.

The formula used by the StandardScaler to transform a feature is as follows:

$$x' = \frac{x_i - \mu}{\sigma}$$

where x' is the scaled feature value, x_i is the original feature value, μ is the mean of the feature values and σ is the standard deviation of the feature values.

Applying this transformation makes the features become dimensionless and on a common scale. I used the Standard Scaler in Chapter 6 for ECFP and Kallisto proximity shells features.

2.5 Machine learning in yield prediction

2.5.1 Yield-BERT

Yield-BERT[125] is a deep learning model designed specifically for predicting chemical reaction yields. It leverages the BERT (Bidirectional Encoder Representations from Transformers) architecture to learn representations of reaction conditions and reactant molecules. I discussed it as one of the breakthrough methods in Chapter 1, and here I want to address more specifics of this method for my research. This method operates on tokenizing reaction strings, and it is important to maintain the same ordering of reactants and reagents in the string to avoid introducing additional uncertainty in the model and causing

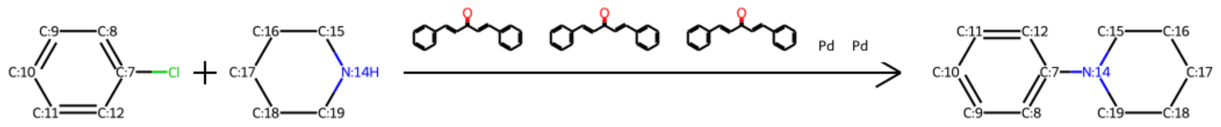


Figure 2.1 Mapping of an exemplary Buchwald-Hartwig reaction derived from a mapped reaction SMILES string [C:7]1(Cl)[CH:12]=[CH:11][CH:10]=[CH:9][CH:8]=1.[NH:14]1[CH2:19][CH2:18][CH2:17][CH2:16][CH2:15]1>C1C=CC(/C=C/C(/C=C/C2C=CC=CC=2)=O)=CC=1.C1C=CC(/C=C/C(/C=C/C2C=CC=CC=2)=O)=CC=1.C1C=CC(/C=C/C(/C=C/C2C=CC=CC=2)=O)=CC=1.[Pd].[Pd]>[C:7]1([N:14]2[CH2:19][CH2:18][CH2:17][CH2:16][CH2:15]2)[CH:12]=[CH:11][CH:10]=[CH:9][CH:8]=1

a lack of reproducibility. I used the following order of reaction component strings for the chapter 4. The order of reagents was arbitrary, meaning that it was not sorted by the role of the reagent, and as originally provided in the reaction data:

$$reagents.reactants >> products$$

And the following order of reaction component strings for the chapter 6:

$$essentialreagent.reactants >> products$$

The concept of essential reagent is explained in more depth in Chapter 6. The Yield-BERT model has two essential parameters to tune: dropout and learning rates. Dropout is a regularization technique used in neural networks to prevent overfitting. It involves randomly "dropping out" (i.e., setting to zero) a fraction of neurons during each training iteration. This helps the network to not rely too heavily on any individual neuron and thus forces the network to learn more robust and generalizable features. The learning rate is a hyperparameter that controls the step size at each iteration while moving toward a minimum of the loss function. It determines how quickly or slowly a neural network model updates its weights during training. I used the original learning rate and lowered the dropout rate to 0.1.

2.5.2 Chemprop

Chemprop[207] is D-MPNN, and it operates on a Condensed Graph of Reaction for reaction-related modeling. This means that a reaction object needs to be constructed and passed to a regular D-MPNN block for training. The reaction object is constructed from the mapped reaction for which I use RXNMapper[1].I illustrate the concept of mapping in Figure 2.1. Reaction mapping aims to map the atoms of reactants onto product atoms. The concept of mapping is complex, and currently, some tools allow reaction mapping, such as RXNMapper[1], Indigo[221], and others. The problem of reaction mapping belongs to the mathematical problem of subgraph isomorphism, which is believed to be intractable without additional domain knowledge that could reduce the complexity of the problem(NP-hard problem)[222]. Chemprop provides an option to adjust the MPNN layer to take a reaction and a molecule as input and encode them with separate MPNNs. I included an essential reagent for this option. Chemprop concatenates information to each atomic and bond feature vector in reaction mode. I used flag –reaction-mode reac_prod, for which each atomic feature vector holds information on the state of the atom in the reactant and concatenates information on the state of the atom in the product. To add the essential reagent, I used additional flag –smiles-columns. I used Chemprop 2.0.4 and the default training parameters.

2.6 Featurization

2.6.1 Kallisto proximity shells

The Kallisto proximity shells fingerprint[203] was developed as a bridge between purely substructural motifs fingerprints such as ECFP, single numerical molecule description, such as combined molecular volume, and QM descriptors such as Gibbs free energy of reacting species. This fingerprint aims to construct a 3D-information-containing vector that describes the charges and steric factors around the reactive atoms in the reactants.

This is achieved by the pipeline illustrated in 2.2. This pipeline first detects the reactive functions of our reaction, like whether the reacting species are amine and acid, for example, or halides. Then, a 3D conformer of the reactants is generated from which charges and proximity shells can be calculated. Lastly, the fingerprint is assembled from the calculated 3D properties into a circular fingerprint centered on the reactive atoms.

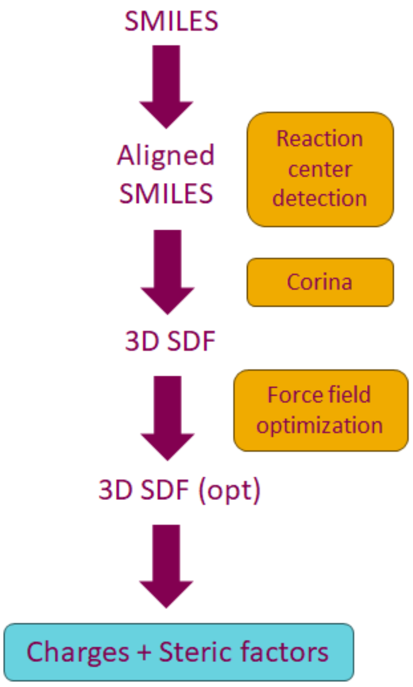


Figure 2.2 Scheme for generation of Kallisto proximity shells and charges fingerprint.

In more detail, to identify the reaction center and reacting atoms, the reaction SMILES components are compared against a set of SMARTS to identify potential reactive functions. This comparison examines the occurrence of functions in both reactant and product parts, assuming that a function no longer present or occurring less frequently in the product part is involved in the chemical transformation. Then, those reactive functions are aligned to have a reaction center in a correct enumeration, and the identified functions in both reactants are used as anchor points. Each SMARTS is designed to match a specific atom, with consistent charges and steric values recorded per neighboring layer. Then, we use Corina to prepare a 3D conformer of each molecule. This conformer is optimized with Merck Molecular Force Field (MMFF) with RDKit[194], and from this 3D structure, it can create a fingerprint based on 3D properties. The fingerprint is similar to a circular fingerprint, meaning it also has a radius from the reaction center to calculate proximity shells and charge properties. It calculates steric hindrance and partial charges for a radius of up to eight. This results in a Kallisto Proximity fingerprint comprising 198 float values for a reaction involving two reactants and yielding one product. Reactant alignment is crucial since it produces a fingerprint with meaningful sorting of fingerprint values, and order is important. Each of the 99+99 values could be decoded into a simple formula of calculation:

- 3 sterimol numbers of the molecule that start the feature vector
- (6 calculated atomic charges + 6 calculated proximity) times 8, and each repeat is responsible for a bigger atomic radius from the detected reaction center, meaning that the first 6+6 are for atomic radius=0, the second is for atomic radius=1, and so on. One can see how the substructures match illustrated in Figure 2.3.

Kallisto fingerprint is a fingerprint of an additive length, meaning that if we have a single molecule, the length of the fingerprint will be 99; for two molecules, it will be 198, and so on. In my work, I consider only bimolecular reactions and reactions with a number of components different than two that are filtered out. Kallisto fingerprint does not encode reagents.

2.6.2 ECFP

Extended Connectivity Fingerprint (ECFP)[200] is a circular fingerprint method widely used in chemoinformatics to represent molecular structures in machine learning tasks. ECFP captures the molecule's presence and arrangement of substructures by iteratively expanding circular neighborhoods around each atom. Starting from each atom, ECFP generates initial identifiers based on the atomic properties and bonds. These identifiers are then iteratively updated by hashing the concatenation of the atom's identifier with those of its neighboring atoms up to a specified radius. This iterative process ensures that ECFP encodes complex topological and chemical information, capturing various substructures like rings, chains, and functional groups. The resulting hashed identifiers set bits in a fixed-length binary vector, creating a compact and informative molecule representation. I used the ECFP fingerprint of reaction, implemented in RDKit by the name Morgan fingerprints, and used a radius of two for chapter 4, corresponding to the ECFP4 fingerprint and radius of three for chapter 6, corresponding to the ECFP6 fingerprint. This fingerprint has a length of 2048, and it is a binary fingerprint.

For chapter 4, I used the order of reaction components as described for the Yield-BERT model, and the reaction was encoded as a concatenation of reactants and reagents and product Morgan fingerprints. This implementation used the default RDKit function *rdkit.Chem.rdChemReactions.CreateStructuralFingerprintForReaction* with default parameters.

For the chapter 6, the fingerprint construction was the following: the fingerprint was calculated separately for each reaction molecule, two reactants, and one product. Then, the fingerprints of reactants were subtracted from the product's fingerprint. I didn't encode reagents while generating the ECFP fingerprint for reaction. This calculation used implementation from rxnutils package[186].

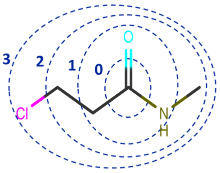


Figure 2.3 Scheme of the concept of molecular radius. One could see substructures matching molecular radius up to 3.

2.6.3 RXNFP

RXNFP[1] is a fingerprint method designed to represent chemical reactions for machine learning applications in cheminformatics. Unlike molecular fingerprints that capture individual molecules, RXNFP focuses on encoding the transformation of reactants into products. It employs a neural network to generate fixed-length vector representations by processing reaction SMILES strings, including reactants and products. This approach allows RXNFP to capture intricate details of the reaction mechanism, such as bond changes and atomic rearrangements, providing a comprehensive depiction of the reaction. RXNFP fingerprint reflects the encoding of the Transformer BERT model used as the RXNMapper tool. This fingerprint has a length of 256 and consists of floats. I used the same reactants reagents order described for the Yield-BERT for Chapter 4.

2.6.4 DRFP

DRFP[201] is a molecular fingerprinting method used to represent chemical reactions for machine learning applications. Unlike other methods, DRFP does not distinguish between reactants and reagents, including them in the SMILES notation. The algorithm extracts all circular substructures with radii of 0, 1, 2, and 3 and all rings from both the reactants and products, storing these as two sets of SMILES-encoded molecular n-grams. It then computes the symmetric difference between these sets to form the final set of molecular n-grams. This final set is hashed into a vector of 32-bit integers and folded into a fixed-length binary vector using a modulo operation, resulting in a 2048-bit binary fingerprint. I used the same reactants reagents order described for the Yield-BERT for Chapter 4.

2.7 NextMove reaction tagging

NameRxn[158] is a software tool developed by NextMove Software that automates the extraction and tagging of reaction types from the chemical literature. Using a large rule-base of known reaction mechanisms and transformations, NameRxn categorizes reactions—such as those extracted from pharmaceutical ELNs or literature—into a NameRxn code, such as "3.1.1 Bromo Suzuki coupling," and assigns an RXNO ontology identifier, such as "RXNO:0000140." The RXNO ontology, maintained by the Royal Society of Chemistry, provides a standardized framework for identifying named reactions.

3 Data

Eins, zwei, drei, vier
Fünf, sechs, sieben, acht
One, two

Kraftwerk, *Nummern*

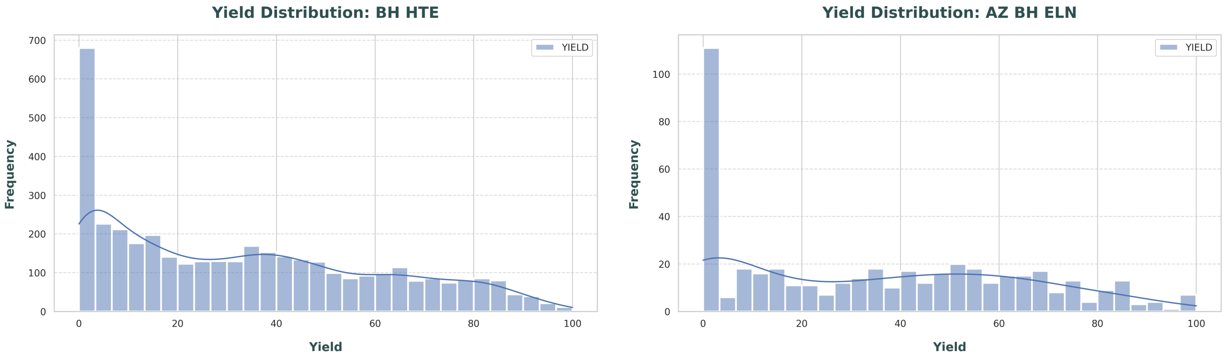
In this chapter, I provide a detailed description of the datasets used in this research. The datasets were obtained in tabular formats, with columns representing reacting species, catalysts, reagents, reaction conditions, and reaction yields. These datasets serve as the foundation for developing and testing the machine learning models applied throughout the research.

3.1 Buchwald-Hartwig HTE Dataset

The Buchwald-Hartwig HTE (High-Throughput Experimentation) dataset, published by Ahneman *et al.* [127], is a key dataset used for studying the Buchwald-Hartwig amination reaction, a vital carbon-nitrogen bond-forming reaction in organic chemistry. This dataset was generated through systematic experimentation and varying reaction parameters such as catalysts, ligands, bases, solvents, and temperatures. The dataset includes thousands of reactions, providing yield data for each, making it ideal for reaction yield prediction tasks. A more detailed discussion of the dataset's cleaning and preprocessing steps is provided in Chapter 4. Figure 3.1a illustrates the yield distribution of the reactions in this dataset, highlighting its variability and the range of conditions tested.

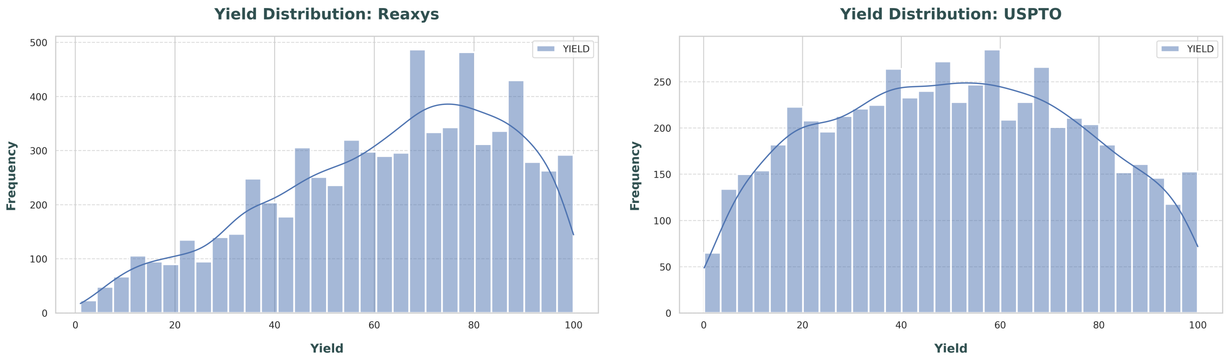
3.2 AstraZeneca Buchwald-Hartwig ELN Public Dataset

This dataset was first introduced by Saebi *et al.* [147] and consists of 1000 instances of the Buchwald-Hartwig reaction collected from AstraZeneca's internal electronic lab notebook (ELN) entries. It has been filtered to include only publicly available data, ensuring the confidentiality of proprietary compounds. The reactions are accompanied by detailed information on reactants, catalysts, and yields. Figure 3.1b presents the yield distribution for this dataset.



(a) Ahnemann HTE Buchwald-Hartwig reactions yield distribution.

(b) AstraZeneca public dataset Buchwald-Hartwig reactions yield distribution.



(a) Reaxys dataset Buchwald-Hartwig reactions yield distribution.

(b) USPTO dataset Buchwald-Hartwig reactions yield distribution.

3.3 Reaxys Dataset

The Reaxys dataset [90] was accessed through AstraZeneca's internal scrape of the Reaxys database, which compiles chemical reactions from the scientific literature using text mining algorithms. Text recognition is not perfect, so certain chemical structures or yield inaccuracies are present. Nevertheless, this dataset covers various reactions, providing valuable insights into yield outcomes under diverse conditions. The specific reactions used in my research are discussed in Chapters 6 and A, where the yield distributions are visualized. Figure 3.2a presents the yield distribution of the reactions after applying a customized data cleaning procedure, further described in Chapter 4.

3.4 USPTO Dataset

The USPTO (United States Patent and Trademark Office) dataset [2] is a well-established resource in computational chemistry, containing millions of reactions extracted from patent documents. It includes information on reactants, products, catalysts, and reaction conditions, allowing researchers to use this dataset as a benchmark for yield prediction and retrosynthesis. Some parsing errors are inevitable as the dataset was generated via text parsing from patent filings. In this work, I utilized a subset of the USPTO data and its yield distribution is shown in Chapter 6. Additional yield distributions for data used in Chapter 4 can be found in Figure 3.2b.

3.5 AstraZeneca ELN Internal Dataset

This dataset was derived from reactions conducted within AstraZeneca from 2008 to 2024. These reactions were logged in AstraZeneca's ELN system and later transformed into a structured, tabular format accessible through internal servers. It contains many reactions, including detailed information about reactants, catalysts, solvents, and yield outcomes. The yield distributions for the subset of reactions used in this research are presented in Chapters 6 and A.

3.6 Enamine Dataset

Enamine Ltd provided the Enamine dataset and consists of two temporal subsets: Enamine 2M and Enamine 280K. The Enamine 2M dataset contains approximately 2 million reactions from 2015-2019, while the Enamine 280K dataset includes 280,000 reactions from 2019-2021. General yield distributions for these datasets are illustrated in Chapter 5, while more detailed yield analyses for specific reaction types are presented in Chapter 6.

4 Data source and diversity matters for yield prediction

I know the pieces fit.

Tool, *Schism*

Adapted with permission from "When Yield Prediction Does Not Yield Prediction: An Overview of the Current Challenges"[115]. Copyright 2023 American Chemical Society.

4.1 Objectives

In this chapter, I investigate the data problems that influence medium- and large-scale data modeling scenarios. As an example of a challenging reaction, I use the Buchwald-Hartwig reaction and its complex impact on the modeling process and the feasibility of modeling in general.

This section is structured into two cases: HTE and real-world data modes, corresponding to modeling using the HTE Buchwald-Hartwig dataset and modeling with USPTO and Reaxys Buchwald-Hartwig reaction selections, respectively. The Buchwald-Hartwig reaction is important in the pharmaceutical industry and has received attention in the modeling community. Despite the attention, there are still few successful cases of it, as it is an example of a reaction with inherent complexity determined by the choice of ligands and conditions.

4.2 HTE data mode: HTE Buchwald-Hartwig amination yield prediction

Ahneman et al. contributed significantly to the yield prediction with their groundbreaking work on the Buchwald-Hartwig reaction, Figure 4.1, within a high-throughput experimentation framework[127]. The reaction dataset in this work was generated using high-throughput experimentation in three 1536-well plates, enabling exhaustive variation of reaction components. The initial dataset retained 3955 reaction data points after eliminating essential control experiments and reactions involving the additive 7. This work used 15 aryl halides, 23 additives, four palladium catalysts, and three bases.

Ahneman et al. used a range of molecular properties derived from DFT-level theory simulations of the reaction components as descriptors. These descriptors included the HOMO and LUMO energies, NMR shifts, dipole moments, electronegativities, and others. The authors evaluated several machine learning models, ranging from linear models, k-nearest Neighbours, Random Forest Regression, Support Vector

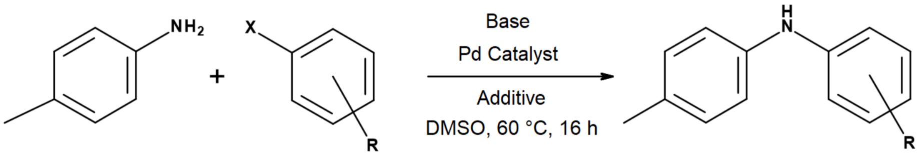


Figure 4.1 Buchwald-Hartwig Amination reaction[127] scheme

Regression, and Bayes generalized linear models to a shallow Artificial Neural Network. Their findings pointed towards the Random Forest model as the top performer.

Their research, however, did not go without contention. Chuang and Keiser critiqued their methodology, presenting evidence that substituting the DFT descriptors with random values or adopting simple one-hot encoding yielded comparable model performances[128]. They posited that the significance Ahneman et al. attributed to the DFT features might have been overstated. Instead of dismissing these claims, Ahneman and his team acknowledged this critique. They concurred on the importance of integrating random controls in subsequent research, emphasizing its critical role in enhancing the robustness and validity of future work[223].

This dataset possesses several unique characteristics worth noting in the context of yield prediction. Firstly, it contains vast, dense reaction data encompassing diverse combinations of reactants, ligands, and reagents, all annotated with the respective yield. This enables the visual representation of the data, as shown in Figure 4.2, clustered into different regions colored by yield. It is possible to identify areas with low and high yields from that.

Furthermore, the high data density and the subsequent cluster analysis offered valuable insights into the scenarios where specific ligands in the HTE setup resulted in suboptimal yields. A more comprehensive examination of this phenomenon was undertaken in the study by Fitzner et al.[176].

The consistent experimental setup maintained throughout the entire HTE campaign ensured the dataset was conducive to accurate predictions of numerical yield values. In such a low-noise environment, the model is more capable of discerning patterns from the relevant reactions, capturing critical information from adjacent data points, and making accurate extrapolations, resulting in highly precise predictions.

Nevertheless, the constraints of the HTE datasets must be recognized. The data is bound by the specific experimental design employed, implying that the model's predictive capability is limited to the scope of this design. Predicting the reaction outcomes for ligands or conditions absent from the dataset could be unreliable or even unfeasible, given the absence of respective training data. This underlines the importance of assessing the applicability of the model domain before its deployment.

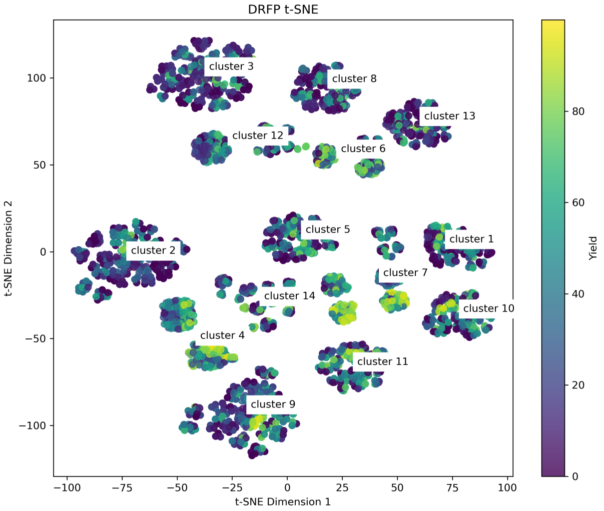


Figure 4.2 t-SNE (t-distributed stochastic neighbor embedding) plot for the Buchwald-Hartwig High-Throughput Experimentation dataset, based on DRFP features. Clusterized with K-Means, number of clusters=14.

I undertook experiments to replicate existing results and evaluate the model's generalization capabilities to obtain a more comprehensive understanding of the state-of-the-art approaches applied to this dataset.

I decided to employ two modeling approaches that reflect current trends in reaction yield modeling:

- A classical tree- and kernel-based ML models utilizing reaction fingerprints.
- The Yield-BERT model, utilizing SMILES encoding, as reported in[125]

Reaction fingerprints (ECFP4,6[200], RXNFP[1], DRFP[201]), described previously in more details in Section 2 were used for SVR[215], RFR[212], and Gradient Boosting Regression[214] (GBR) models. For the modeling process, I used Scikit-Learn[209] Python library.

The selected model types also exemplify various Machine Learning approaches. Random Forest Regression and Gradient Boost Regression are ensemble methods; the former ensembles decision

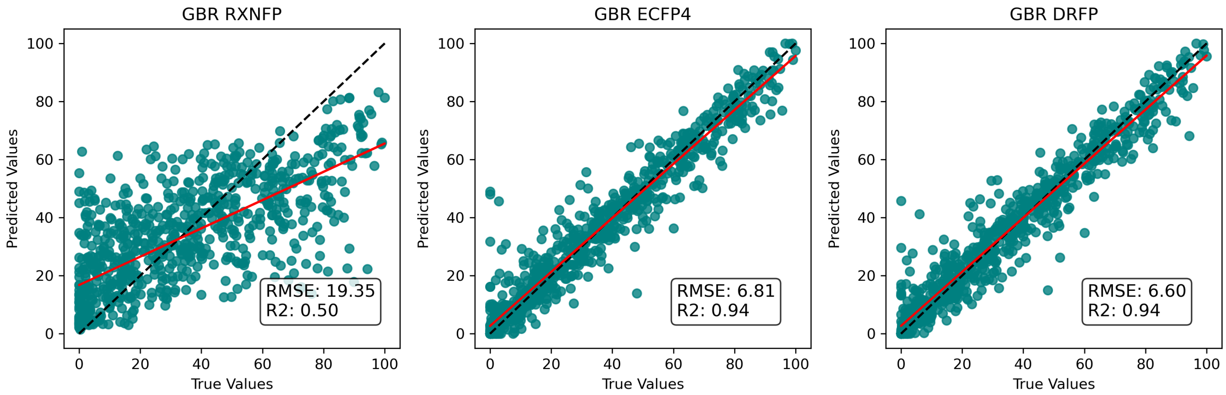


Figure 4.3 Comparison of the GBR model's performance using different encodings and fingerprints, trained with a random 80:20 ratio and 5-fold cross-validation. RMSE - Root Mean Square Error, R^2 - determination coefficient. The red line represents numpy linear fit. RFR and SVR models were excluded from the main figure for clarity, and their detailed results can be found in the Appendix Figure A.1.

Cluster №	1	2	3	4	5	6	7	8	9	10	11	12	13	14
RMSE	7.71	8.50	12.97	13.54	4.77	23.66	13.33	7.66	9.15	5.59	4.46	17.90	9.56	7.78
R^2	0.90	0.86	0.66	0.73	0.96	0.36	0.76	0.88	0.87	0.96	0.98	0.40	0.84	0.92
Mean yield	28.10	25.19	23.33	53.01	30.31	45.94	58.16	23.04	31.28	38.45	40.38	31.75	21.82	35.77

Table 4.1 Leave-one-out cluster performance of Gradient Boosting Regression model based on DRFP features. For the visual representation of the model's performance, see Figure A.4.

trees, while the latter ensembles weak models. On the other hand, Support Vector Regression utilizes support vector machines to learn the best-fit hyperplane to categorize the data.

I chose these different fingerprint methods to compare various approaches for encoding reactions as objects. RXNFP represents a pure data-driven encoding approach, while ECFP and DRFP represent structural approaches. This comparison allows us to gain insights into the strengths and limitations of each method in the context of yield prediction.

For embedding purposes and to avoid any possible bias connected to how different methods align the reaction components, I use the following order to build the reaction object:

$$reagents.reactants >> products$$

Initially, the models performed modestly on a random split, as shown in Figure 4.3. The results reveal that, among the simple models, the DRFP[201] encoding exhibits the best performance, slightly outperforming ECFP4 fingerprints.

I conduct further evaluations on the different parts of the chemical space occupied by the dataset. In Figure 4.2, the t-SNE dimensionality reduction performed on DRFP features and the fact that the dataset nicely separates into different clusters. I employed a leave-one-cluster-out validation setup with clusters defined based on the DRFP features. As summarized in Table 4.1, the results indicate generally satisfactory performance, albeit with some variability in clusters that may be considered combinations of smaller subclusters.

Upon analysis of the results, it became evident that the model's efficacy tends to diminish less when the mean of a given cluster is closer to the mean of the overall distribution. Conversely, there is a marked decline in performance when the yield of a cluster deviates substantially from the overall mean. This indicates that the model probably struggles when predicting yields at more extreme values.

Furthermore, I investigated the model's ability to extrapolate across reactants by executing a leave-one-reactant-out validation, explicitly focusing on aryl halides in Table 4.2 one could see the results of the model trained on leave-one-reactant-out. The visual results are depicted in Figure A.5. The first column row corresponds to chlorine-associated aryl halides, the middle column to bromine-associated

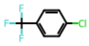
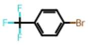
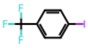
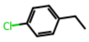
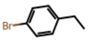
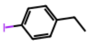
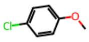
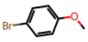
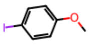
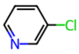
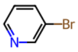
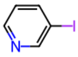
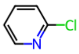
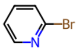
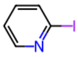
Left-out aryl halide			RMSE			R^2			Mean cluster yield		
Chlorine(Cl)	Bromine(Br)	Iodine(I)	Cl	Br	I	Cl	Br	I	Cl	Br	I
			12.29	10.69	15.41	-0.49	0.38	-0.31	12.51	26.90	33.71
			14.66	15.35	12.01	-13.84	0.61	0.75	3.87	43.51	52.58
			14.68	10.86	12.90	-3.36	0.64	0.48	1.94	25.89	32.54
			18.91	12.78	12.82	-0.52	0.8	0.8	13.85	43.0	51.26
			10.51	11.09	14.37	0.85	0.82	0.71	43.48	52.45	58.61

Table 4.2 The performance of Gradient Boost Regression model on DRFP features with leave-one-aryl halide out. For the graphical representation of the performance, see FigureA.5.

aryl halides, and the last column to iodide-associated aryl halides. The model performs moderately well when the left-out species is a chemically reactive aryl halide. Still, the performance deteriorates when the left-out species is less reactive, for example, chlorine-containing aryl halides. This observation highlights the model's susceptibility to variations in the chemical properties of the reactants and its potential limitation to generalize across the chemical space, even for a well-defined single chemical reaction type.

I also accessed Yield-BERT properties related to the BH HTE dataset, and they showed the same results, as reported in [125], although, on leave-one-reactant out, it showed better performance than simple models. See Appendix Figure A.3 for more information.

4.3 Real-world data mode: Diverse datasets Buchwald-Hartwig amination yield prediction

In this section, I investigate the case that illustrates the challenges of yield prediction and emphasizes the importance of advancing our knowledge in conditions encoding and enhancing the prediction methods overall. I showcase various aspects of yield prediction, underscoring the complexity involved.

To obtain the reaction data, I used the web interface of Reaxys[90](7K entries), access provided by the Technical University of Munich, and other available open-source datasets, such as AZ ELN 750[147](500 entries), Doyle's HTE Buchwald-Hartwig[127](4K entries), and data extracted from USPTO[2](6K entries). The reactions were cleaned from duplicates and invalid entries (non-parsed via RDKit), then mapped with RXNmapper[1], and were classified with

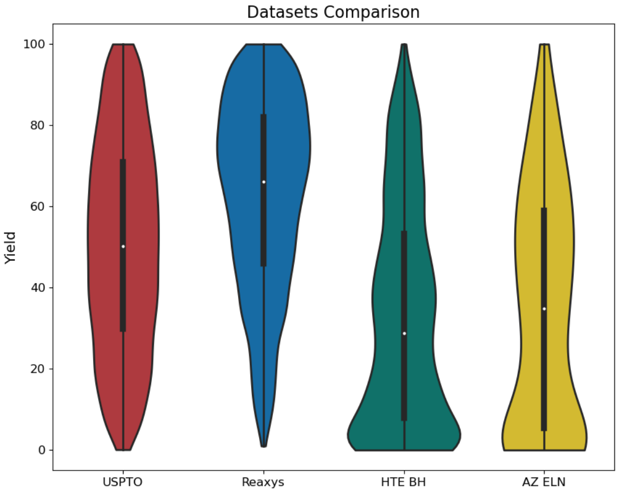
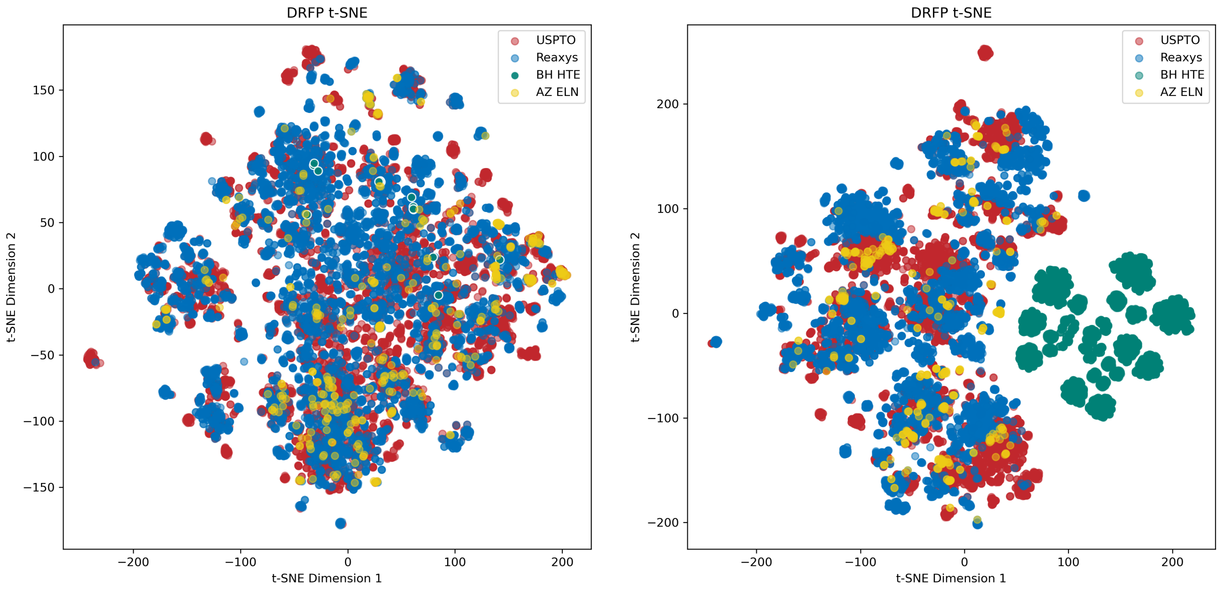


Figure 4.4 Violin plot for yield distribution for the datasets derived from public data and Reaxys.



(a) Conditions excluded

(b) Conditions included

Figure 4.5 The t-SNE plot depicts the distribution of reaction encodings based on DRFP representations. In A.11a, where all conditions are excluded, the encodings show an even distribution in hyperspace, but the amount of BH HTE reactions is reduced to only 15 since it is the amount of the unique aryl halides on which the reaction conditions were tested. In A.11b, when conditions are included, a notable separation occurs between the BH HTE dataset and others. This indicates that condition representations introduce diversity, adding a new layer of complexity to the encodings. I investigate the data recordings more in detail in Figure 4.7.

NameRXN[158]. Reaction data labeled with the Next Move classes 1.3.1, 1.3.2, 1.3.3, 1.3.4 (Chloro-, Bromo-, Iodo-, Trifluoxy-Buchwald-Hartwig Amination, correspondingly) was selected.

As shown in Figure 4.4, the datasets obtained from academic experiments and industrial patents are characterized by higher reported yields, whereas datasets derived from Electronic Laboratory Notebook records and High-Throughput Experimentation tend to often contain lower-yielding reaction data points. It is worth noting that while the US Patent and Trademark Office (USPTO) dataset demonstrates a similar, relatively uniform, yield distribution for this specific reaction, it is widely acknowledged that the general distribution of the USPTO data is significantly skewed towards high-yielding reactions [125].

Furthermore, I analyzed the distribution of reaction embeddings using t-SNE. This will serve as a qualitative analysis of the applicability domain of the models. Notably, when reagents were included, the High-Throughput Experimentation dataset exhibited distinct separation in the DRFP embeddings, as illustrated in Figure 4.5. Conversely, Reaxys, USPTO, and AZ ELN datasets occupied dissimilar regions within the chemical space. This discrepancy could be attributed to variations in the fundamental recording of reaction components, particularly in the context of Palladium catalysts; I investigate this more in Figure 4.7. This observation leads me to propose the hypothesis that Buchwald-Hartwig reaction experiments documented in patents and articles may demonstrate a higher degree of reagent diversity than HTE experiments and that we lack a general procedure for standardized recording of the reagents. I touched on this topic in the Introduction, and the modeling supports the necessity of better agreement on the standardization of recording reagents and catalysts of reactions.

Using the extracted data, I modeled using the same procedure detailed in the previous section. The analysis of the model performances, as reflected in the RMSE and R2 in Figure 4.6, reveals that the results achieved are unsatisfactory. When tested on Buchwald-Hartwig reaction data extracted from various sources, simple models perform the same as the more complex Yield-BERT model, see Appendix Figure A.6. The result gives us moderate performance with R2 0.23 on the inner USPTO

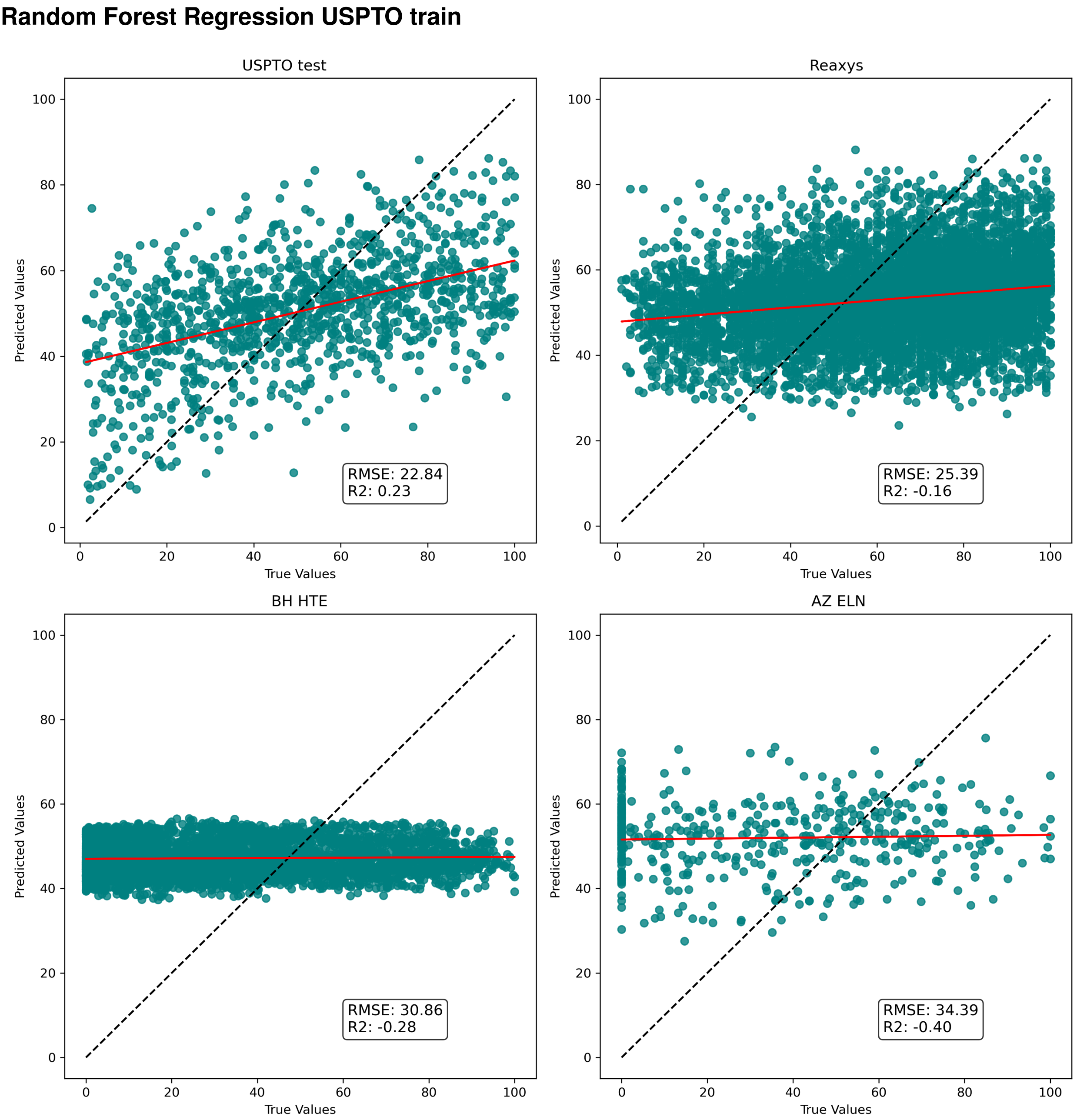
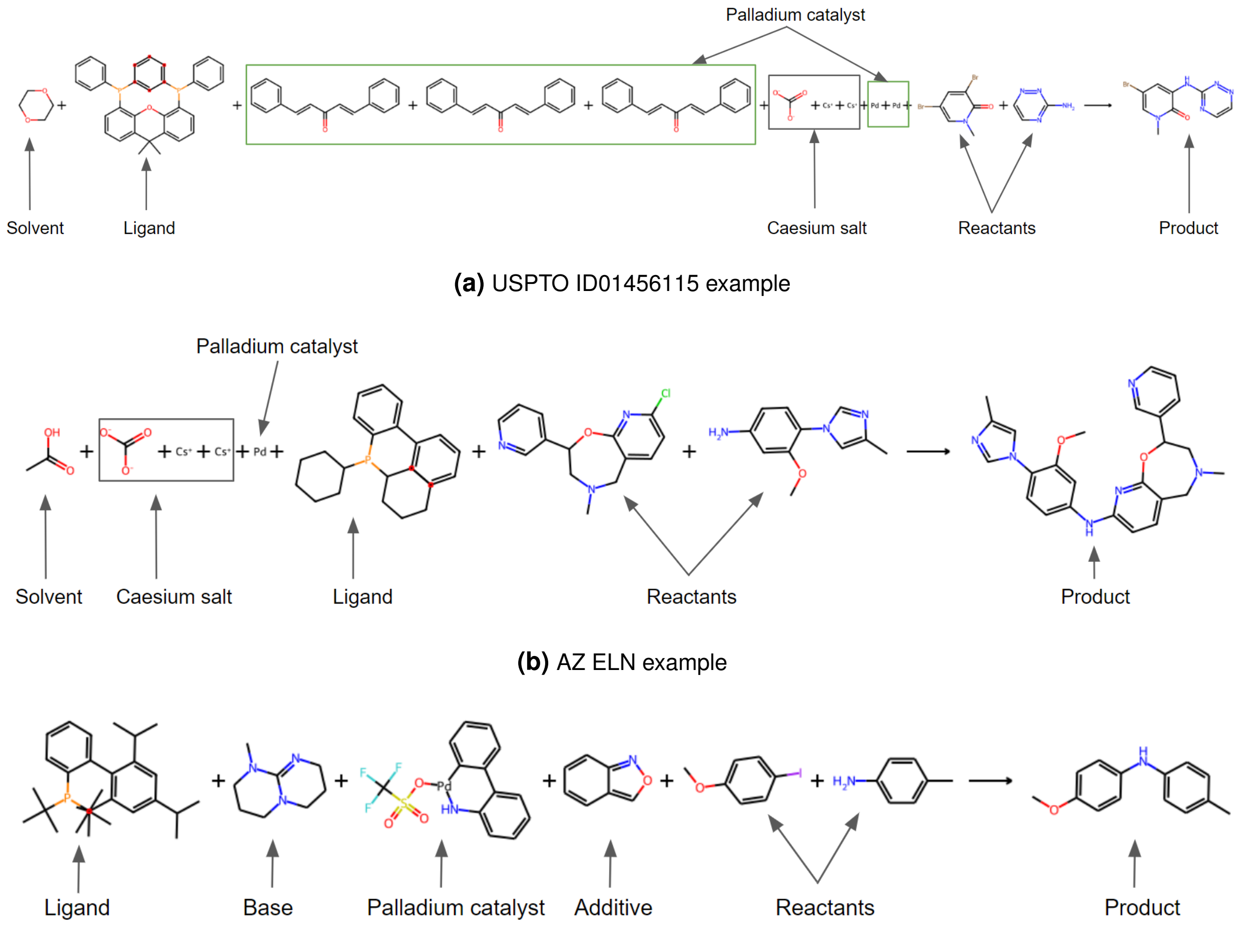


Figure 4.6 RFR model trained on USPTO Buchwald-Hartwig selection and tested on other datasets. For clarity, I only show DRFP fingerprint performance in these plots. Other fingerprints' performance can be found in the Appendix.



(a) USPTO ID01456115 example

(b) AZ ELN example

(c) BH HTE example

Figure 4.7 While essential components like aryl halides, amines, palladium catalysts, ligands, and bases are commonly used, variations in experimental conditions and the presence of additional additives or reaction components highlight the complexity of standardizing data for this reaction. This issue becomes particularly evident when comparing palladium catalyst representations across different data sources. For example, the catalyst appears "disassembled" in the USPTO entry, is represented as bare palladium in the AZ ELN, and takes the form of a complex pre-catalyst in the BH HTE dataset. These divergent representations of the catalyst get encoded differently and likely contribute to its separate clustering in t-SNE analysis.

test set and negative R2 for test sets. This lack of performance and generalization ability could stem from various factors, including noise within the data. However, as indicated by the t-SNE plots in Figure 4.5, there is considerable overlap between the USPTO and Reaxys dataset, indicating that the Reaxys reactions are within the applicability domain of the USPTO-derived model. The same can be said for at least the AZ ELN data but less for the HTE dataset. This observation implies that current featurization methods might struggle to capture the intricate nuances inherent to specific reactions.

Consequently, the challenges in capturing the intricate chemistry inherent in this specific reaction are not unexpected, and the results of these experiments corroborate the challenges posed by the vast and diverse chemical space. I will investigate this more in the following chapters, highlighting the challenges connected with data standardization and the lack of quality of the data derived from reliable sources.

4.4 Conclusion

This study shows the weaknesses of the current encoding methods and the hardships in tackling metalorganic compounds widely used in popular chemical reactions. One could also see that on the diverse data, current SotA models struggle to generalize, but all the models perform similarly well on the confined reaction space, showing that there are complexities with yield as a metric when used on the wide dataset and even on selected subspaces of a confined dataset. This leads to a desire for better data homogenization via more deliberate data preparation and filtering with the hypothesis that it may help refine the data selection and make it more learnable. I will further investigate the influences of the elaborate data preparation in Chapter 6.

5 Conditions-enriched Enamine yield modeling

> gonna make it right
> gonna get it done
> when it’s comin’ over me
> what i can’t deny
>
> Mariusz Duda, *How To Overcome Crisis*

5.1 Objectives

This study was performed during a one-month-long secondment in Janssen in February 2023. The objective of this study was to finetune the model called BEE or BERT-Equivalents-Enriched Embedding, developed by Janssen scientists on proprietary Enamine data from Enamine chemists’ experiments over 2015-2019 as the training set (Enamine2M in the text) and 2019-2021 as the test set (Enamine280K in the text). This experiment was intended to evaluate the model trained on Enamine data on Janssen internal data and vice versa. This would have shown the possibility of the transferability between differently derived data. The objective of this study was also to understand whether it is possible to develop a general reactivity model that would generalize well over a broad chemical space.

5.2 Paper summary

Neves et al.[148] article describes the model used for this project, emphasizing the importance of integrating reaction condition information into the model. Their motivation stemmed from the observation that reactions with identical reactants can yield different results under varying conditions, a factor often overlooked by current models. They employed the concept of general reactivity modeling and training across all reaction classes. Previous regression models on general reactivity have shown poor performance, partly due to the dataset’s bias toward high-yielding reactions. To address this, they opted for a binary classification model with a yield threshold of 5%.

Key components of their methodology include:

- The use of Yield-BERT as a base model.
- An additional layer with IDs (classes) corresponding to the concentrations of reactants and products.
- Pre-training on the USPTO dataset, where all IDs were set to 0 due to the absence of concentration information in USPTO.

The paper introduces a key innovation by embedding molecular ratios into the model. The model captures fundamental chemical differences influencing reactivity by incorporating the molar ratio between reagents and the limiting reactant. This approach enables more precise predictions of reaction outcomes based on the reactants’ stoichiometry.

One limitation of the SMILES representation is the dual role of the molecule separator "." which also denotes ionic bonds. A new embedding layer is introduced, assigning different representations to distinguish these roles, as represented in Figure 5.1. This ensures a continuous and accurate depiction

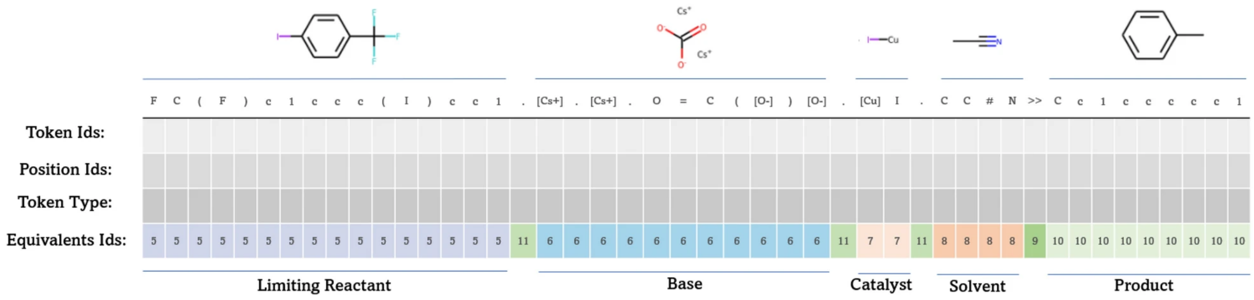


Figure 5.1 Principle of encoding additional conditions into the SMILES string on the example of a random reaction. The figure is taken from the original paper.[148]

of molecular structures, such as salts. This differentiation leverages existing molecular entity separation in the Janssen ELN, eliminating the need for additional algorithmic solutions.

The methodology draws an analogy to natural language processing, where "segment id" helps distinguish between questions and answers. Similarly, the BEE model treats reactants as the "question" and products as the "answer." This distinction is embedded into the model, enhancing its ability to differentiate between the roles of molecules within a reaction.

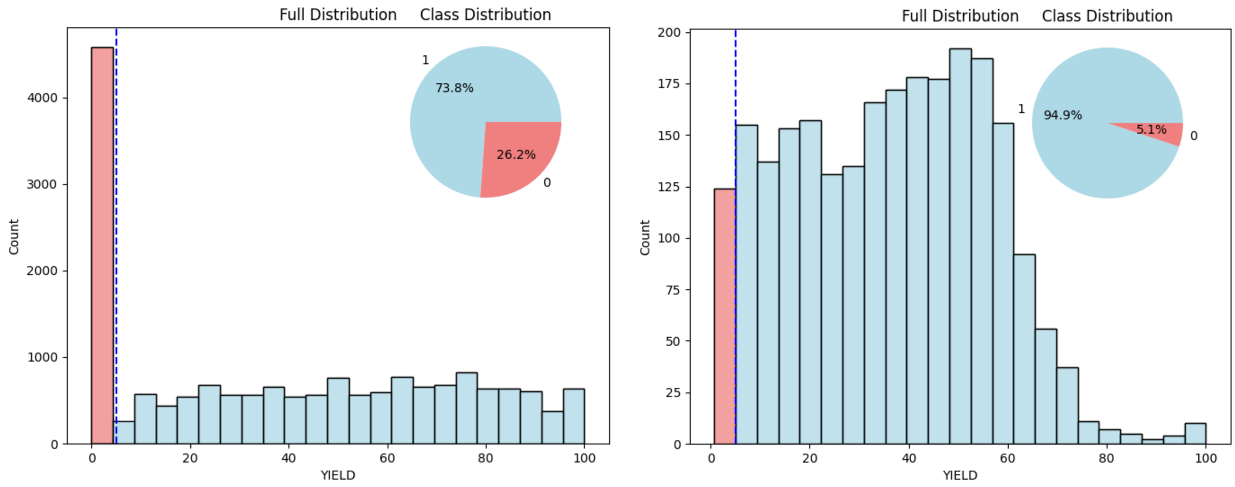
In cases where specific information is absent, particularly for solvents, the equivalent class in the embedding can infer the molecule's role. This feature is crucial for understanding the impact of various molecules on reaction outcomes. For instance, the model can identify that certain solvents or catalysts do not contribute atoms to the product but still influence reactivity.

The practical implementation involves converting molecules to embedding IDs, with each token, such as the separator ".", assigned a unique ID based on its context. This consistent representation helps the model learn the roles and impacts of different molecules. Numerical values like molar ratios, moles, and concentrations are converted into categorical IDs based on predefined thresholds derived from chemical knowledge and data distributions.

The new embedding layer is initialized during pre-training but remains inactive, with placeholders for the additional information. This setup allows the embedding layer to be enriched during fine-tuning by integrating vectors for each "Equivalent ID" with the standard embedding. This enrichment incorporates various aspects such as molecule roles, stoichiometric data, and reaction conditions, thereby enhancing the model's predictive capabilities.

5.3 Methodology

To apply this model to Enamine historical data, the data required preprocessing. The data did not have equivalents as described in the original work. Still, it had the time of reaction and temperature of the reaction, which were provided as a dictionary of the reaction protocol and the corresponding standardized time and temperature of this protocol. Each unique combination of time and temperature was assigned to the reaction as "segment id" for all reaction tokens. Reactions went through preprocessing as described in the paper and as described in[185], which involved aromaticity fixing, functional groups transformation into single representation form, and atom-to-atom mapping. For atom-to-atom mapping, RXNMapper[1] was utilized, and molecules were processed with CGRTools[206]. I used a pre-trained model on USPTO for 20 epochs, provided in the supporting paper repository, and I fine-tuned this model using Enamine2M data and Enamine280K as a hold-out test set to evaluate the training results. The data had class imbalance, as shown in Figure 5.2. I trained the model in a binary classification mode, where class 0 were Enamine reactions that yielded less than 5% and class 1 more than 5%. I also trained the model in a regression mode with an actual reaction yield. I also trained the model in the regression mode, which was gated to classification on the evaluation stage using the same threshold used for training classification models. Two losses were used: cross-entropy for classification and RMSE for regression.



(a) Enamine2M

(b) Enamine280K

Figure 5.2 I illustrate yield numeric value and class distributions after applying threshold at 5%. These plots show a significant imbalance in classes of Enamine data.

5.4 Results and Discussion

I trained a number of models, and the general summary of the most important experiments performed during the internship is summarized in Table 5.1. I started from a vanilla model trained in classification mode for 25 epochs and tried to improve its performance. During training, the model exhibited instability in classification mode, frequently getting stuck in a local minimum where it classified all data points as class 1. I implemented class-weight adjustments to address this and counteract the class distribution imbalance, but it didn't increase performance. I also addressed this by training the models in regression mode and applying a classifier gate on the prediction stage to evaluate the model. Additionally, I experimented with various dropout and learning rates, though these adjustments had minimal impact on performance.

The model's performance was suboptimal in regression mode, with an RMSE exceeding 25% and R^2 less than 0.1. The inclusion of enrichment did not statistically improve performance. One can see the performance of the best model on the Enamine280K test set in Figure 5.3.

Despite time constraints during the internship, significant insights were gained, although full exploration of all potential solutions was impossible.

The persistently low performance and lack of substantial improvement with enrichment are likely due to the low informational contribution of the dataset. The approach of assigning general classes to entire reactions rather than to individual reacting species may be misaligned with the dataset's structure. Additionally, the yield data from Enamine, recorded as crude or using various purification methods, might not accurately reflect true reactivity, and this critical information was missing from the dataset. Also, the hypothesis on the general reactivity model did not hold true, as it was shown that the model's performance was not good enough even within the same dataset.

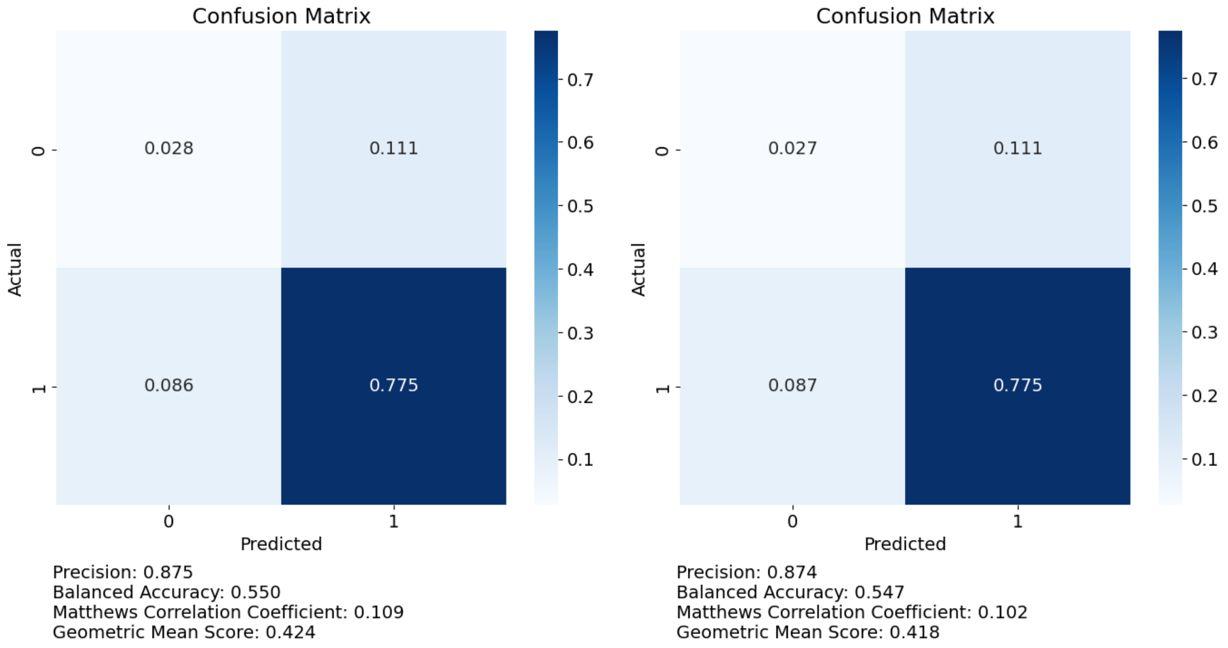
Moreover, the approach of incorporating equivalents is not broadly applicable, as this type of data exists in only a very limited number of datasets, making it challenging to apply to other sources, such as USPTO or Reaxys.

Unfortunately, I could not work directly with Janssen data or transfer the model to their servers, which would have been ideal for testing the hypothesis as intended.

Looking forward, continued work on this project could involve direct communication with Enamine representatives to obtain equivalents for the reactions included in the dataset. If this is not feasible,

Additional information	Test set	Epochs number	Eqvs incl	Class weight	Less dropout	F1 Score	ROC AUC	MCC	Precision
no	Inner hold-out	25	yes	no	no	0.928	0.778	0.676	0.873
no	Inner hold-out	25	no	no	no	0.926	0.773	0.669	0.870
no	Enamine 280K	25	yes	no	no	0.887	0.550	0.109	0.875
no	Enamine 280K	25	no	no	no	0.886	0.547	0.102	0.874
no	Inner hold-out	30	yes	yes	yes	0.867	0.666	0.393	0.817
no	Inner hold-out	30	no	yes	yes	0.864	0.666	0.386	0.819
no	Enamine 280K	30	yes	yes	yes	0.865	0.539	0.074	0.873
no	Enamine 280K	30	no	yes	yes	0.859	0.538	0.071	0.873
Regression as classification	Inner hold-out	40	yes	no	yes	0.852	0.5	0.0	0.742
Regression as classification	Inner hold-out	40	no	no	yes	0.883	0.621	0.436	0.791

Table 5.1 Results of final training epoch for the models trained as classifiers. All the models described there have the base model as 20-epochs pre-trained USPTO. I tried to lower the dropout rate of the model since it was high, and the high dropout rate negatively impacted the training, making it more unstable. Also, I tried to balance the class's contribution to the training, assigning more weight to underrepresented class. One could see that the best performance was achieved using the vanilla model with no adjustments and "equivalents" included.



(a) "equivalents" included

(b) "equivalents" excluded

Figure 5.3 Confusion matrices comparing the model's performance, fine-tuned on Enamine2M dataset on a test set of Enamine280K, with and without enrichment. The model: 25 epochs, no class weight, original dropout rate.

Janssen scientists could be asked to prepare a dataset with the same label assignment as the Enamine data, where each reaction is assigned a label based on a unique combination of temperature and time.

By aligning the datasets in this manner, it would be possible to rigorously test the models' transferability between these two datasets, thereby advancing the understanding of their applicability across differently derived data.

6 Multi-class yield prediction

What if you could look right
Through the cracks?
Would you find yourself
Find yourself afraid to see?

Nine Inch Nails, *Right Where It Belongs*

6.1 Objectives

The general objective of this project was to develop yield models that could be variable for routine usage to predict whether the new reactions for the synthesis of several classes have poor, moderate, or good yields. In more detail, I wanted to test whether the transferability problem discussed in previous chapters would be a problem if the training of the models is done on diverse datasets extracted from the public domain (USPTO) and more high-quality data from Reaxys and internal data of AstraZeneca and Enamine. The research also focused on developing a robust pipeline for data preprocessing and cleaning using the textual description of reaction procedures. The reaction yield as a regression metric is hard to predict since, as discussed in Section 1, the natural mean standard deviation of the datasets available is around 16%, which leads to the decision to try to mitigate this challenge by binning the data into several classes. In this chapter, I will focus on the amide coupling reaction findings, and other reaction information will be described in the Appendix section.

I also involved chemists from the automated synthesis lab to help provide their library experiments for some real-life applicable validation of the models, which I discuss in more detail in the results section of this chapter.

6.2 Data

I focused on performing modeling on some selected reaction classes since the global reactivity modeling is not feasible, as mentioned by Schwaller *et al.*[125]. I have started with the hypothesis that the models should learn in-domain knowledge of specific reaction specifics to render them useful for everyday use. Of course, in the future, an ensemble of such trained on specific reaction class models could play as a "global reactivity model." My second hypothesis is that a reliable, good model should be able to perform well within the same applicability domain over several different datasets. I work further to make the applicability domain more precise with reagents curation and purification filtering of the data, which I believe would help to make models more robust and precise. So, I had two hypotheses in mind that could be summarized as: a reliable, useful model should perform well in a selected applicability domain that I define as a reaction belonging to a selected reaction type.

6.2.1 Reaction classes utilized

I selected specific reaction classes to ensure transferability between datasets and model-challenging reactions that were previously researched. Only two reaction types were present in the Enamine dataset: $\mathrm{S_nAr}$ and amide coupling. These were also selected from ELN, Reaxys, and USPTO. Additionally, I chose Suzuki and Buchwald-Hartwig couplings, as they are currently considered challenging reactions in the field[224]. Also, I considered reductive amination reactions in my studies. These classes were chosen to leverage transferability across different datasets and address complex reactions. I illustrate schemes of these reactions in Figure 6.1.

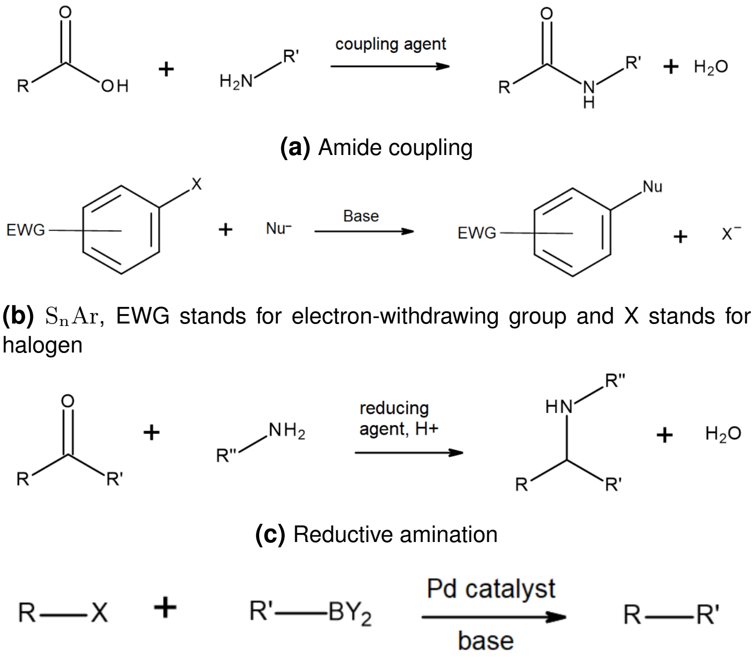
As a rough approximation of reaction classes, I used NextMove software classification[158] of reductive amination, Buchwald-Hartwig couplings, Suzuki couplings, $\mathrm{S_nAr}$, and amide coupling reactions. Although the NextMove reaction type tagging is very good already, it still misclassifies some reactions; for example, the most misclassified instances are reactions tagged as $\mathrm{S_nAr}$ but having transitional metal catalysts, which should not be present in this reaction type. Further cleaning is achieved via reagent curation. The class codes and text description are in Table A.2.

In this chapter, I will describe the results and findings for *amide coupling*, and the necessary information for other reaction types can be found in the Appendix, Figures A8-A20 and Tables A3-A30. It was interesting to look closely at amide coupling reaction since it is known to be successful with a higher probability than Buchwald-Hartwig amination, for example, and predicting whether a given amide coupling reaction will be high-yielding or low-yielding is especially useful.

6.2.2 Data collection

I utilized reaction data from multiple sources: AstraZeneca's internal Electronic Laboratory Notebook (ELN), Reaxys, USPTO, and Enamine.

From AstraZeneca ELN data, I also extracted a subset of library data (Library test in the text), the experiments produced by the automated synthesis team. These libraries are HTE data but are often used directly in biological assays after an automated purification procedure. Since this work on yield prediction is partially done in collaboration with this team, this test set is important to include as it gives a possibility to estimate the usefulness of such models for the library design to exclude the poorly yielding reactions. This data was processed with ELN data, but it was held as a separate test set of high interest. All data were presented in a table format, with columns including reaction SMILES, reagents, yield, ID, data source, and more.



(a) Amide coupling

(b) $\mathrm{S_nAr}$, EWG stands for electron-withdrawing group and X stands for halogen

(c) Reductive amination

(d) Suzuki coupling

Figure 6.1 Scheme of reactions studied in this section.

I used source XML tree files to extract the data from the USPTO dataset, which I parsed to extract information on reaction SMILES, yields, reagents, and reaction procedures. I illustrate the general algorithm for extraction in the Algorithm 1. As one can see, the algorithm includes parsing XML files and extracting a number of data features regarding how the data should be written out in a table format. This also included pre-curation of yield values, as there are different values corresponding to one reaction in the original files, and I take care of it.

Data from Enamine was previously described in Chapter 5. Additionally, AstraZeneca provided data from Reaxys.

6.2.3 Yield curation

I implemented a data preprocessing pipeline to minimize noise and ensure high-quality data. As I am working with the reactions derived

Algorithm 1 Data Extraction and Curation Pipeline for USPTO data extraction

```
Input: USPTO raw data in XML files grouped by years
Output: Curated data ready for featurization and modeling
for each year folder in the dataset do
    for each XML file in the year folder do
        Traverse the XML tree structure
        Extract the following fields:
            documentId, paragraphText, paragraphNum, reactionSmiles, PERCENTYIELD, CALCULATEDPERCENTYIELD
        Perform a sanity check:
        if number of paragraphNum entries == number of reactionSmiles entries then
            Store the results in a table with columns:
                documentId, paragraphNum, reactionSmiles, PERCENTYIELD, CALCULATEDPERCENTYIELD
        end if
    end for
end for
for each record in the table do
    if PERCENTYIELD is absent then
        Use CALCULATEDPERCENTYIELD
    else if CALCULATEDPERCENTYIELD is absent then
        Use PERCENTYIELD
    else if PERCENTYIELD == CALCULATEDPERCENTYIELD then
        Use the yield value
    else if absolute difference between PERCENTYIELD and CALCULATEDPERCENTYIELD is too large then
        Discard the instance
    end if
end for
for each reaction in the curated data do
    Map reactions and separate reagents
    if reaction is not sanitizable then
        Discard the reaction
    end if
end for
The curated data is now ready for featurization and modeling with Yield-BERT and Chemprop
```

from different data sources with different information available, I need to distinguish them and treat them differently.

ELN

ELN data contains a column "conclusion phrase" that notifies the reader of the general conclusion about the experiment. There are several phrases, and the phrases that I considered as indicative of interesting data were only three: "Reaction successful," "Test reaction failed," and "Reaction failed." The two latter were especially valuable since they provided negative data points to my dataset that are negative with high certainty. I also selected reactions with a positive yield that has no conclusion phrase. Still, all other instances with no conclusion phrase and 0 yield were discarded due to high uncertainty around this value since 0 yield records with no explanation could mean different than the reaction failed to happen.

USPTO, Reaxys, Enamine

I applied a different treatment to reactions from these datasets due to the lack of conclusive phrases indicating the experiment's outcome. Specifically, I excluded all reactions reported with a 0 yield, as these datasets often use 0 in the yield field when the actual value could not be reliably parsed from text. This approach is common in sources like Reaxys and USPTO, where the 0 yield value often reflects parsing challenges rather than a true experimental outcome, introducing substantial uncertainty. Consequently, I excluded these 0-yield reactions to avoid potentially significant noise in the data. The origin of the 0-yield values in Enamine was also unclear, so these were similarly discarded.

While this decision improves data quality by removing ambiguous entries, it does, unfortunately, eliminate a considerable portion of the negative data, adding complexity to the dataset and limiting the representation of truly unsuccessful reactions.

General

The preprocessing pipelines were implemented with a set of assumptions that are described now. I also describe the algorithm of yield preprocessing in the first part of Algorithm 2.

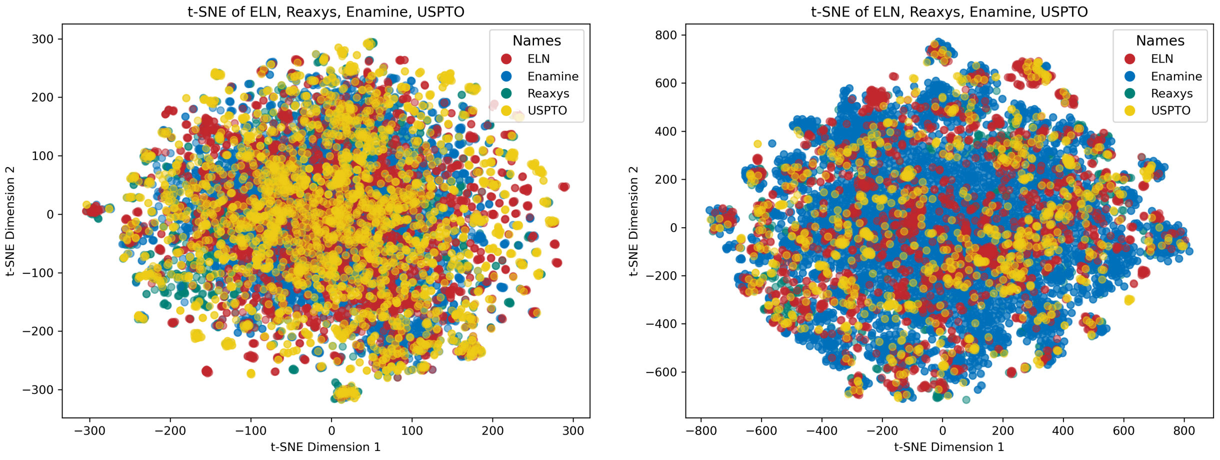
- Any reactions with yield values over 100% were discarded to ensure data accuracy.
- The 0-yielding reaction was dropped for duplicate reactions if the non-0-yielding pair was tagged as successful.
- Regular duplicates with identical yield values were dropped with the first one left.
- Duplicates with significantly different yield values (differences greater than 1%) were retained to preserve natural data variability.

6.2.4 Reagents curation

To ensure high-quality data selection, I developed a comprehensive manual dictionary for the five reactions chosen for study. This dictionary was based on the frequency of encountered reagents, categorized into solvent, base, acid, reducing agent, oxidizing agent, source of metal, source of ligand, and activator. Each reagent could fall into multiple categories. I tagged the most popular reagents with an instance count of over 30. Reactions with rarer reagents were excluded. Also, reagents could be dropped if they are unsuitable for the reactions. In Figure 6.2, I illustrate the subselection of the reagents that belong to different types of reactions but not to $\mathrm{S_nAr}$, highlighting the importance of well-rounded reaction curation and not overly relying on proprietary software tagging.

Algorithm 2 Data Preprocessing Pipeline

```
Input: Table original reaction data from AZ, Reaxys, USPTO, and Enamine
Output: Cleaned and preprocessed reaction data
Yield cleaning
for each reaction in the dataset do
    if reaction yield == 0 AND no conclusion phrase then
        Remove the reaction from the dataset
    else if source is AZ then
        if conclusion phrase indicates "Test reaction failed" then
            Remove the reaction from the dataset
        end if
    end if
    if source is Reaxys OR USPTO OR Enamine then
        if reaction yield == 0 then
            Remove the reaction from the dataset
        end if
    end if
    if reaction yield > 100 then
        Remove the reaction from the dataset
    end if
end for
for each reaction pair (r1, r2) in the dataset do
    if r1 and r2 are duplicates then
        if one reaction has yield == 0 AND the other is tagged as successful then
            Remove the 0-yielding reaction
        else if both reactions have identical yield values then
            Remove one of the duplicates
        else if difference in yield values is greater than 1% then
            Retain both reactions
        end if
    end if
end for
The cleaned data is now ready for reagent cleaning or modeling
Reagent cleaning
for each reaction in the dataset do
    if reaction reagent is NOT in essential reagents then
        Remove the reaction from the dataset
    end if
end for
The cleaned data is now ready for purification cleaning or modeling
Purification cleaning
for each reaction in the dataset do
    if reaction procedure text does NOT contain key phrases then
        Remove the reaction from the dataset
    end if
end for
The cleaned data is now ready for modeling
```



(a) ECFP-based t-SNE

(b) Kallisto-based t-SNE

Figure 6.3 The t-SNE plot depicts the distribution of reaction encodings based on ECFP and Kallisto proximity shell representations for amide coupling. I provide more details on the plot data points selection in the text. As one can see, the distribution of the feature space is uniform, and there are no distinct clusters populated by some datasets and not by others.

To refine the selection of reaction classes, I defined "essential reagents" necessary for each reaction type: an activator for amide coupling, a source of metal for Buchwald-Hartwig and Suzuki coupling, the base for $\mathrm{S_nAr}$, and a reducing agent for reductive amination, as mentioned above the reaction arrows in Figure 6.1. Reactions lacking these essential reagents were filtered out.

I used RDKit to sanitize and canonicalize the reagent molecules to avoid missing the same reagents due to ambiguity in SMILES notation. Sanitization removes molecules that don't pass sanitization checks in RDKit, which involves valency checks, aromaticity checks, and other checks for the molecule to be chemically valid. The canonicalization of a molecule is the process of outputting a unique canonical SMILES, which is important since a molecule could have many different SMILES representations, including non-canonical SMILES and tautomers. Although this approach is not ideal, I aimed to test the hypothesis that introducing filters based on reagents along with their roles would improve the performance of the models since I would eliminate the noise connected to false reaction tagging.

I also illustrate the t-SNE distribution of the data after this data filter based on ECFP and Kallisto features in Figure 6.3. Since plotting every instance on the plot is not possible, I needed to make a selection of data points to get represented on the plot. This plot illustrates the data selection based on the frequency of the essential reagents of these reactions, so the data points with more popular reagents get more representation on this plot. I selected 1000 data points from each dataset. One could see that the data is distributed pretty homogeneously, with no visible clusters populated by some data sources, not others. For other reaction classes in the

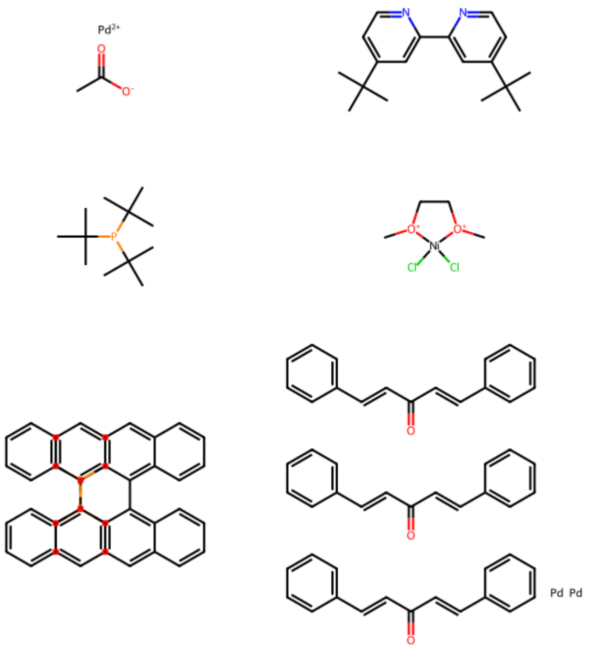
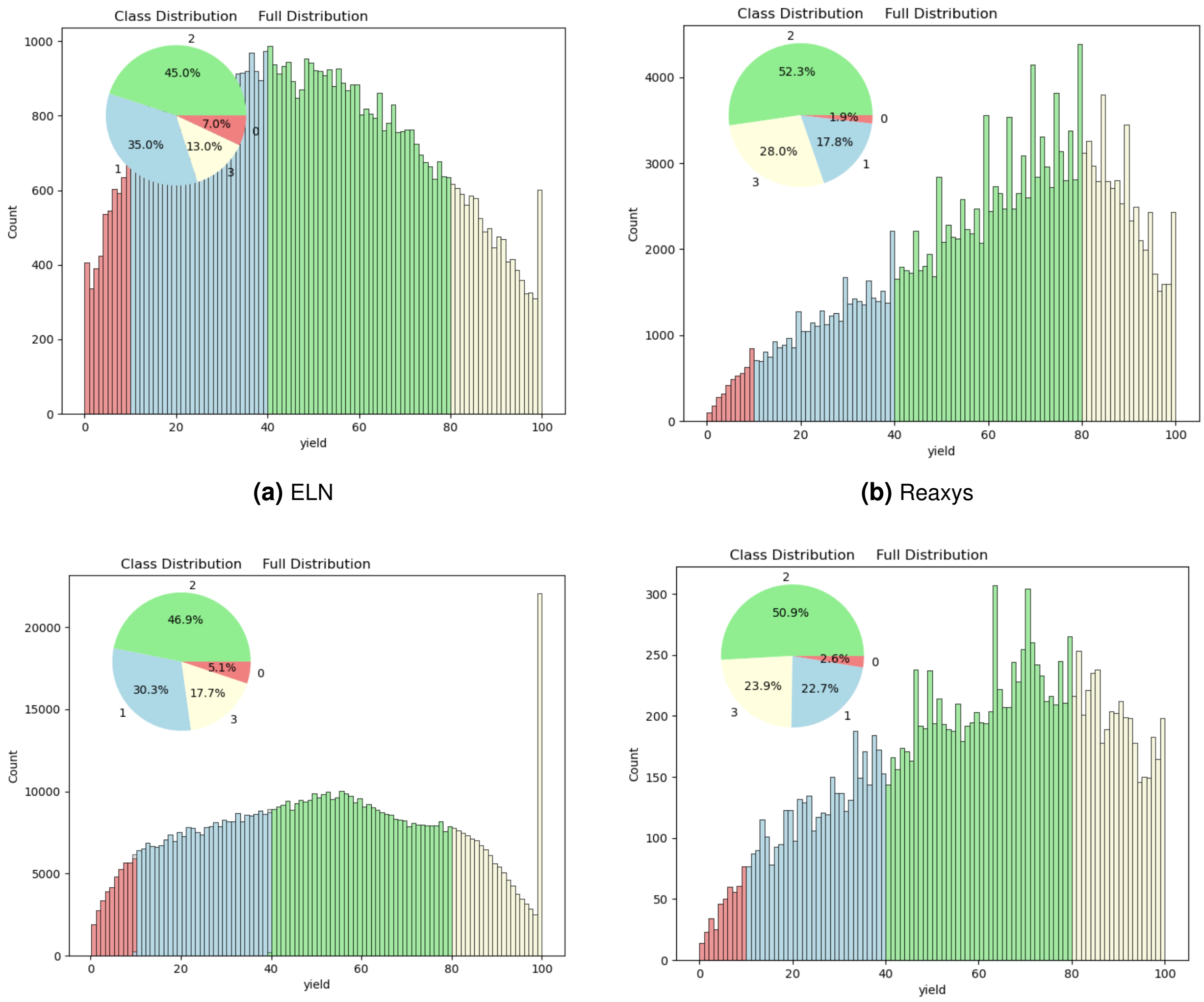


Figure 6.2 Wrong reagents were encountered during reagents tagging for the $\mathrm{S_nAr}$ reaction.



(a) ELN

(b) Reaxys

(c) Enamine

(d) USPTO

Figure 6.4 Float and class distributions of the yield after reagents filtering using thresholds determined by Optuna trials. Colors indicate the classes after applying the thresholds defined by Optuna optimization with the lowest-yielding class of under 10% in red, moderate class from 10% to 40% in blue, high moderate from 40% to 80% in green, and top class from 80% to 100% in yellow.

Appendix, one could observe some variability and formation of segregated clusters, although mostly these clusters are populated with all representative datasets.

This is valuable insight since it provides an idea that the data I am working with is within the applicability domain that I defined, and there are no significant discrepancies in the data.

6.2.5 Purification curation

I created a manual dictionary using raw text from reaction procedures to identify key phrases related to purification methods. This allowed me to filter out reactions with crude yields, retaining only those purified by column chromatography or other reliable methods, such as preparative LCMS or crystallization. I hypothesized that it could improve data quality, allowing training higher quality models on lesser data. The dictionary details can be found in the Appendix. The main point of such filtering was to filter the reactions with the crude yields, which are known to be more optimistic and only leave the purified reactions, thus showing an absolute value of yield of a given reaction.

Unfortunately, only ELN and USPTO data had the raw text of the reaction procedures, while Reaxys and Enamine did not.

I summarize the amount of data cleaned after each step of the cleaning pipelines in Tables A.3, A.4. I also present the results for the amide coupling reaction training datasets in Figure 6.5. As one can see, the quantity of data is dropping significantly, especially for USPTO, with the original selections having 104.7K reactions; after the yield cleaning, it shortened by 2.5 times to 39.8K. After cleaning the reagents, the amount was 23.5K; after purification cleaning, it was 16.8K, so it is almost tenfold less left. The amount left is around a third or quarter for other reactions after reagent cleaning, as shown in Figure6.5.

6.2.6 Data yield distribution

I illustrate the yield distribution among different datasets of amide coupling after the reagents cleaning step and divided by the classes, which I later discuss in Results, in Figure 6.4. One could see that due to the yield cleaning step that removes 0-yielding reactions for all datasets, except for ELN, lack representations of low-yielding reactions. This was contributed by the current reporting challenges in the literature, where the data from failed experiments is not represented in patents, which USPTO datasets consist of, and papers from which Reaxys extracts the data, as well as patents, included in Reaxys. One also can see a peak of 100%-yielding reactions in the Enamine dataset, which is an interesting finding supported by the fact that the company operates with many quantitative reactions and has universalized pipelines that operate on a limited number of protocols. One could also see that ELN has well-represented middle and not-so-much high-yielding reactions, compared to USPTO and Reaxys, which have a high representation of highly successful reactions.

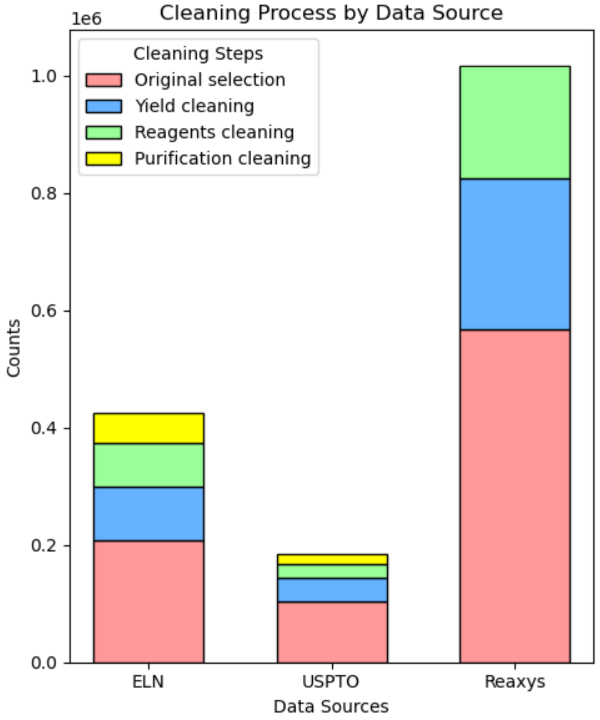


Figure 6.5 The data after the cleaning steps staked on each other shows that yield cleaning is the most demanding step, and after it, the amount of data drops but not too significantly. The illustration is for the amide coupling reaction.

6.3 Models

Since I have shown the challenges connected to different descriptors in the previous Chapter 4, I also tested a slightly different but similar set of models and fingerprints to determine whether the model can learn from high-quality data and predict data from other sources.

I used Yield-BERT[125] as the Natural Language Processing representative and Chemprop[207] as the Graph Neural Network representative.

For the classical approach using fingerprints, I used ECFP and Kallisto proximity shell fingerprints to represent the classical approach modeling. ECFP performance is a baseline for other models since it is the simplest fingerprint to calculate and use, and its approach to calculating substructures is a classic in chemoinformatics. Kallisto proximity fingerprint (abbreviated as Kallisto) is representative of a hybrid methodology, which includes some knowledge of the chemical 3D structure of the molecule and quickly calculated properties of the reactive center. The generation of Kallisto fingerprints and ECFPs was discussed in Section 2.

To get an initial guess on which classical model to use before using the classification to tune the threshold values, I used the Python package LazyPredict, which I fitted to the regression task on whole ELN data. I illustrate the findings in the Appendix Table A.5. The best-performing models were Random Forest Regressor, ExtraTreesRegressor, and XGBRegressor. I also used the literature search for current best-performing models on yield prediction and similar property prediction topics, and it supported my initial idea to use Random Forest as a model robust to overfitting, which is also known for its capabilities of finding patterns in complex features[213].

Models were trained as described in the following:

- For training and evaluation on the inner hold-out test set, the model was trained on 80% of the full data, and 20% of the data was used as a test set.
- The models were fit with the full dataset for evaluation on other datasets.
- For Library tests in ELN data, the subset of ELN data was selected as a test set using keywords, and the model was fit on the rest of the data.

6.4 Results

6.4.1 Class optimization results

In a classification approach, it is crucial to define the classes of the target variable accurately, which, in my case, is the reaction yield. This can be achieved by dividing the yield, originally a continuous numeric value, into distinct classes based on threshold values. For instance, yields can be categorized into multiple classes, with one class on each side of the threshold. I hypothesized that using 3-class or 4-class designs would provide greater detail and insight than a binary classification (go/no-go) or regression. This hypothesis is based on the observation that reaction yield deviates with a minimum RMSE of 16%, and further improvement seems unlikely with the current hand-derived dataset.

To divide the data into three classes, I would need two threshold values; to divide the data into four classes, I would need three values, correspondingly. I also hypothesized that the division would not equally represent the available data distribution. For this, I would have needed to implement a strategy for dealing with an unequal quantity of data points in each class. To perform this optimization of division values, as well as the training parameters of the Random Forest Classifier model, we would need to have two things: a good hyperparameter searching tool and good metrics to evaluate the models' performance.

There are some hyperparameter optimization frameworks available; I decided to use the Optuna framework[216] since it is easy to implement a custom objective function that would implement not only a search of models' hyperparameters but additional hyperparameters that are not related to the model, in my case, threshold values and sampling technique for representation of underrepresented classes.

I implemented several metrics based on which the model's performances have been evaluated. I used four metrics: geometric mean, precision, balanced accuracy, and MCC. The optimization procedure of a trial is as follows: Optuna selects the model's hyperparameters and thresholds, and the average metric values derived in 5-fold cross-validation are recorded. Based on the metrics values, Optuna will change the selected parameters in the next trial to maximize the metrics.

So, to summarize, the objective was to optimize the Random Forest model's hyperparameters and the threshold values for class division. The hyperparameters included the number of trees, the depth of trees, and the maximum features of trees. To address data imbalance, I evaluated the following strategies: leaving the class distribution unchanged, undersampling the most common class, oversampling the least represented class, and adjusting class weights. I mention these parameters in more detail in Section 2.

The most successful weighting approach was to balance class weights, and I used it to further model the random forest models.

I selected the most successful trials and used metric averaging to determine the best thresholds and parameters. The most successful trials are in the table 6.1. One could see that Optuna trials often proposed very low thresholds to separate low-yielding reactions. These thresholds served as preliminary orientations.

The 5-fold cross-validation results indicate that the models perform quite well across the selected thresholds. High precision and balanced accuracy scores demonstrate the models' effectiveness, particularly for the Amide coupling and Buchwald-Hartwig reactions. Although some reactions like Suzuki coupling and $\mathrm{S_nAr}$ exhibit lower balanced accuracy and MCC values, the overall metrics suggest that the models are generally robust and capable of reliable performance in most cases.

Interestingly, they aligned with "chemist-relevant" points for each class option. This alignment validated the yield divisions described in Vogel's textbook[225], where yields around 100% are "quantitative," above 90% are "excellent," above 80% are "very good," above 70% are "good," above 50% are "fair," and below 40% are "poor." Despite such coincidence, this is most likely contributed by the data yield distribution and not due to the intrinsic physics of the reaction.

I used these optimization results as rough guidelines for future modeling and class division, adopting threshold values of **[15, 65]** for 3-class classification and **[10, 40, 80]** for 4-class classification.

Table 6.1 Threshold optimization studies from Optuna trials with the best threshold results. Metrics reflect the performance of 5-fold cross-validation.

Reaction	Fingerprint	Thresholds	Precision Macro	Balanced Accuracy	MCC	Geometric Mean
Amide coupling	RF Kallisto	[2, 65]	0.671±0.013	0.605±0.006	0.430±0.014	0.582±0.007
Amide coupling	RF ECFP	[2, 63]	0.648±0.008	0.599±0.006	0.397±0.010	0.582±0.008
Suzuki coupling	RF Kallisto	[14, 67]	0.741±0.027	0.395±0.003	0.212±0.015	0.147±0.004
Suzuki coupling	RF ECFP	[15, 73]	0.739±0.015	0.383±0.001	0.194±0.006	0.143±0.007
Buchwald-Hartwig	RF Kallisto	[1, 54]	0.686±0.008	0.678±0.008	0.533±0.012	0.664±0.009
Buchwald-Hartwig	RF ECFP	[1, 62]	0.687±0.003	0.663±0.003	0.543±0.004	0.629±0.007
Reductive amination	RF Kallisto	[16, 76]	0.675±0.026	0.354±0.002	0.113±0.008	0.107±0.009
Reductive amination	RF ECFP	[24, 77]	0.676±0.025	0.369±0.004	0.130±0.005	0.118±0.008
$\mathrm{S_nAr}$	RF Kallisto	[11, 69]	0.717±0.017	0.372±0.002	0.156±0.005	0.137±0.007
$\mathrm{S_nAr}$	RF ECFP	[5, 79]	0.800±0.047	0.347±0.003	0.109±0.095	0.071± 0.011
Amide coupling	RF Kallisto	[5, 19, 61]	0.562±0.006	0.505±0.009	0.377±0.006	0.455±0.006
Amide coupling	RF ECFP	[9, 45, 79]	0.607±0.011	0.278±0.003	0.076±0.008	0.084±0.005
Suzuki coupling	RF Kallisto	[3, 18, 64]	0.516±0.003	0.555±0.004	0.372±0.003	0.539±0.003
Suzuki coupling	RF ECFP	[9, 50, 79]	0.655±0.038	0.293±0.003	0.149±0.008	0.059±0.006
Buchwald-Hartwig	RF Kallisto	[1, 41, 81]	0.563±0.007	0.568±0.006	0.475±0.010	0.541±0.008
Buchwald-Hartwig	RF ECFP	[9, 33, 62]	0.524±0.010	0.520±0.011	0.400±0.014	0.496±0.013
Reductive amination	RF Kallisto	[2, 33, 86]	0.578±0.013	0.517±0.008	0.360±0.014	0.493±0.012
Reductive amination	RF ECFP	[2, 18, 58]	0.556±0.009	0.504±0.004	0.363±0.005	0.462±0.007
$\mathrm{S_nAr}$	RF Kallisto	[3, 24, 63]	0.568±0.002	0.571±0.001	0.409±0.002	0.560±0.002
$\mathrm{S_nAr}$	RF ECFP	[2, 13, 79]	0.576±0.008	0.500±0.003	0.402±0.004	0.402±0.007

6.4.2 Classification results

To understand whether my hypothesis of making the selection of training and test data precise to the chosen reaction domain will positively influence the performance within the defined domain, I needed to train the models on every step of my developed cleaning pipeline and evaluate them on different test sets that were prepared in the same fashion. Thus, I have three subsections: the subset with only basic yield and incorrect fingerprint removal, the more advanced subset where I removed reactions with irrelevant reagents, and the final subset with only reactions with purification information.

For Chemprop, the performance in the table is based on the best-performing epoch based on the evaluation set, and that checkpoint was tested on the test set; it has an inner implemented selection process to evaluate the best epoch. For Yield-BERT, I evaluated the checkpoints of each epoch of the training on the evaluation set and selected this epoch checkpoint for further evaluation on other test sets, this model requires manual handling.

Results for only yield-cleaned data

I summarize findings for the 4 and 3-class classification based on amide coupling reaction in Tables 6.2 and 6.3. The models here are trained on the data derived from the first step of the reaction data cleaning pipeline.

The tables are organized to depict a training set on the left side of the table, and test sets are under the training sets. I sort them by the openness of the data utilized, as USPTO is a completely open-source dataset, Reaxys is purchasable, and ELN is AstraZeneca's proprietary data. For each training dataset, the table contains performance on hold-out test sets and other datasets. For amide coupling and $\mathrm{S_nAr}$ reactions, I also have test data from Enamine, which I used as merged Enamine 2M and Enamine 280K datasets described in the previous section. For ELNs, there is an additional test set: reaction libraries, which were done in AstraZeneca's automated synthesis labs (abbreviated as Libs. test).

I used BA and the MCC to show the model's performance. I put the best results in BA in bold. For random prediction, for 4 classes, BA would be 0.25, and for 3 classes, 0.33. MCC > 0.3 is a moderate correlation, and MCC around 0 is no better than a random classifier. The tables show that the models perform better than this baseline on the inner test sets but do not manage to generalize to external test sets.

Let's start with the 4-class classification: Looking at USPTO-trained models, one can see that Chemprop achieved the highest BA of 0.454 and an MCC of 0.300 on the inner test set, indicating it handles the diversity within the USPTO dataset well. On the Reaxys test set, RF ECFP had the best BA of 0.335 and an MCC of 0.107. This suggests that there is some, albeit limited, compatibility between the USPTO and Reaxys datasets. On ELN and Enamine test sets, the best BA was 0.285 for the ELN test set with RF ECFP and 0.261 for the Enamine test set with Chemprop. However, the MCC values were low, indicating poor generalization.

Looking at Reaxys-trained models, on the inner test set, RF ECFP showed the highest BA of 0.470 and an MCC of 0.271. On the USPTO test set, RF ECFP had the best BA of 0.503 and an MCC of 0.317, suggesting some overlap in the types of reactions covered by Reaxys and USPTO. Interestingly, this high performance is not observed in any other models, suggesting better generalizability capabilities of ECFP; this is also observed in other reaction types, supported by the tables in the Appendix. On ELN and Enamine test sets, Yield-BERT performed best on the ELN test set with a BA of 0.319 and on the Enamine test set with a BA of 0.257.

Finally, for the ELN-trained models, Chemprop achieved a BA of 0.548 and an MCC of 0.419 on the inner test set, indicating strong performance. Yield-BERT performed best on the libraries test set with a BA of 0.287. On USPTO, Reaxys, and Enamine test sets, RF ECFP had the highest BA on USPTO with 0.288, Yield-BERT performed best on Reaxys with 0.300, and Yield-BERT again led on Enamine with a BA of 0.258. Despite high performance in the inner test set, Chemprop showed close to random performance on other test sets, indicating the tendency of this model to strongly overfit the data.

Table 6.2 Results for 4-class models for amide coupling trained on only yield filtered datasets. Best balanced accuracy performance in bold.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.439	0.221	0.419	0.254	**0.454**	0.300	0.365	0.175
	Reaxys	**0.335**	0.107	0.274	0.109	0.278	0.057	0.257	0.028
	ELN	**0.285**	0.039	0.269	0.032	0.286	0.021	0.251	0.007
	Enamine	0.255	0.006	0.25	0.002	**0.261**	0.009	0.25	0.003
Reaxys	Inner test	**0.47**	0.271	0.449	0.323	0.464	0.360	0.419	0.274
	USPTO	**0.503**	0.317	0.292	0.193	0.268	0.032	0.251	0.014
	ELN	0.298	0.052	0.258	0.032	0.265	0.020	**0.319**	0.11
	Enamine	0.256	0.009	0.25	-0	0.253	0.003	**0.257**	0.024
ELN	Inner test	0.484	0.255	0.493	0.316	**0.548**	0.419	0.468	0.292
	Libs test	0.261	0.024	0.27	0.065	0.229	-0.023	**0.287**	0.096
	USPTO	**0.288**	0.049	0.275	0.042	0.264	0.020	0.267	0.025
	Reaxys	0.296	0.057	0.273	0.031	0.263	0.029	**0.3**	0.081
	Enamine	0.257	0.008	0.249	-0.002	0.250	-0.001	**0.258**	0.015

Next, we turn to the 3-class classification: the expected BA from a random classifier would be 0.33. The models performed better than this absolute baseline on their inner test sets, with varying degrees of success on external test sets. Starting with USPTO-trained models, on the inner test set, both RF Kallisto and Chemprop achieved the highest BA of 0.531, with Chemprop having a higher MCC of 0.342, indicating better handling of the USPTO dataset. On the Reaxys test set, RF ECFP performed best with a BA of 0.416 and an MCC of 0.143, again showing compatibility between these two datasets. On the ELN and Enamine test sets, RF Kallisto led the ELN test set with a BA of 0.387, and RF ECFP had the highest BA of 0.342 on the Enamine test set.

In Reaxys-trained models, RF ECFP had the highest BA of 0.548 and an MCC of 0.327 on the inner test set. In the USPTO test set, RF ECFP again performed best with a BA of 0.565 and an MCC of 0.341. Yield-BERT led ELN with a BA of 0.426 in the ELN test set, while RF ECFP had the highest BA of 0.346 in the Enamine test set. However, for both datasets, the MCC is low (0.189 and 0.05, respectively), indicating limited generalizability.

In ELN-trained models on the inner test set, Chemprop achieved the highest BA of 0.612 and an MCC of 0.504, indicating robust performance on the ELN dataset. Chemprop also performed best with a BA of 0.376. On the libraries test set, MCC is 0.084, indicating problems with generalizability. In USPTO, Reaxys, and Enamine test sets, RF ECFP had the highest BA on USPTO with 0.382, Yield-BERT led on Reaxys with 0.394, and RF ECFP had the best BA on Enamine with 0.343.

To summarize, on inner test sets of each dataset, Chemprop and RF ECFP generally perform well, showing good internal validation results. However, performance drops significantly on external test sets, with MCC values often close to zero, indicating poor generalization across different datasets.

In this example, no single model consistently outperforms others across all datasets. Still, if one takes a look at the performances in the Appendix, one can see that there is a clear winner for most reaction types: RF ECFP, which leads me to the conclusion that for the not-so-clean reaction data, ECFP could be the best fingerprint choice, although it has some limitations such its length, which leads the models

Table 6.3 Results from 3-class models on amide coupling trained on only yield filtered datasets. Best balanced accuracy performance in bold.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.514	0.262	0.531	0.322	**0.531**	0.342	0.44	0.235
	Reaxys	**0.416**	0.143	0.384	0.147	0.338	0.034	0.36	0.081
	ELN	0.368	0.06	**0.387**	0.118	0.337	0.014	0.345	0.063
	Enamine	**0.342**	0.015	0.331	-0.007	0.336	0.015	0.333	-0.003
Reaxys	Inner test	**0.548**	0.327	0.538	0.408	0.533	0.429	0.507	0.359
	USPTO	**0.565**	0.341	0.444	0.297	0.353	0.046	0.355	0.059
	ELN	0.392	0.108	0.389	0.115	0.338	0.014	**0.426**	0.189
	Enamine	0.346	0.024	0.33	-0.008	0.337	0.010	**0.356**	0.05
ELN	Inner test	0.571	0.31	0.564	0.358	**0.612**	0.504	0.561	0.357
	Libs test	0.352	0.02	0.33	-0.022	**0.376**	0.084	0.365	0.094
	USPTO	**0.382**	0.081	0.353	0.103	0.367	0.094	0.333	0.001
	Reaxys	0.383	0.09	0.341	0.06	0.363	0.085	**0.394**	0.168
	Enamine	**0.343**	0.011	0.333	-0.002	0.331	-0.007	0.34	0.012

to train for a longer time and it is more time-consuming to retrain model every time the new data arrives to the system.

Results for reagents-cleaned data

If one examines the performances of USPTO-trained models for both 3 and 4 classes, one can see that the best-performing model on the inner test set is Chemprop, with BA 0.441 for 4 classes and 0.538 for 3 classes. However, it is important to note that this model is not performing well on other test sets, with the performance around random. This means the predictions are not significantly better than random chance, indicating performance issues.

After Chemprop, the next best-performing model is RF ECFP with BA 0.435 for 4-class and 0.507 for 3-class. This model performs better for other test sets with BA 0.320 and 0.398 for the Reaxys test set, which leads to an observation that it could be due to partial overlap between Reaxys and USPTO data since Reaxys contains patent data, which then becomes more evident when I look at the vice versa performance of Reaxys-trained RF ECFP model which shows performance better than the inner test set, in 4-class BA is 0.469 for inner test and 0.490 for USPTO test set. That is not observed for any other fingerprint, which could communicate that ECFP is better at capturing the structural data of the reaction than other fingerprints.

Unfortunately, none of the USPTO-trained models could predict the Enamine test set with a higher degree of BA and MCC than the random. One could see that the same is true for the other datasets-trained models, with MCC being around 0.01 for all the models, which is very close to 0, indicating the complexity of the prediction.

If one looks at Reaxys-trained models, one can see that ECFP fingerprint performs best in 3 and 4-class models for the inner test and USPTO test sets, but for ELN prediction, Yield-BERT performs better than other models. For Enamine, the performance of all models is close to the random.

Table 6.4 Results from amide coupling trained reagents filtered datasets, 4 classes. On sides the training data sources.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.435	0.215	0.402	0.251	**0.441**	0.282	0.377	0.172
	Reaxys	**0.320**	0.092	0.265	0.084	0.249	0.009	0.257	0.025
	ELN	**0.278**	0.031	0.268	0.030	0.254	-0.002	0.252	0.008
	Enamine	**0.255**	0.004	0.250	-0.001	0.251	0.003	0.253	0.006
Reaxys	Inner test	**0.469**	0.269	0.455	0.328	0.447	0.328	0.446	0.306
	USPTO	**0.490**	0.298	0.287	0.181	0.248	-0.003	0.252	0.008
	ELN	0.292	0.043	0.257	0.035	0.250	-0.009	**0.319**	0.109
	Enamine	**0.257**	0.010	0.250	-0.000	0.253	-0.000	0.255	0.018
ELN	Inner test	0.473	0.236	0.470	0.301	**0.531**	0.415	0.467	0.293
	Libs test	0.247	0.013	0.269	0.065	0.249	0.006	**0.303**	0.116
	USPTO	**0.284**	0.048	0.267	0.026	0.254	0.010	0.269	0.031
	Reaxys	0.295	0.051	0.269	0.025	0.259	0.013	**0.296**	0.081
	Enamine	0.258	0.010	0.247	-0.009	**0.264**	0.015	0.263	0.028

One can see that Yield-BERT performs at its best in libraries by checking the performance of ELN-trained models. The other reactions and performance tables in the Appendix also hold this statement true. This finding is interesting, as the other models show performances much closer to the random. However, the chemical space of AZ inner data should be more homogeneous than Reaxys or USPTO. Also, for an inner set of ELN, Chemprop is performing best on all reaction classes. Unfortunately, Chemprop models show close-to-random performances on other test sets, indicating overfitting.

As one can see, for inner test sets, most models perform better than random models with a definite correlation. Still, the performance deteriorates drastically for most other datasets that models try to predict. I also could see an interesting finding about the possible overlap of data between Reaxys and USPTO since Reaxys also includes some part of patent data that could be seen in ECFP performance, and this is observed for all models trained on Reaxys and evaluated on USPTO, which shows possible superiority of ECFP fingerprints performance. Overall, observation shows that there is no clear winner in the case of amide coupling, perhaps ECFP in the 4-class classification, but not in the 3-class; in the 3-class, it's more or less spread with Yield-BERT having a slightly better performance. The performances of all models are similar on all the data, with some models having some chance of better performance. However, if one takes Enamine datasets as a general test set for all models, one could see that the difference in BA in models between the best and the worst is 0.004-0.028, which is basically within a statistical error. One could see that Yield-BERT shows some distinctly better performance for the Libraries test.

If one looks at the performances in the Appendix, there's a recurring pattern of the RF ECFP model being slightly worse on inner test performances but superior on other test sets. For the $\mathrm{S_nAr}$ reaction in 4 classes, ECFP was superior on USPTO inner, Reaxys, and Enamine test sets, with Kallisto being better on ELN with negligibly higher MCC. This is the case for 3 classes, except Kallisto performed better on the inner test and ECFP on all others. On Reaxys training set, the situation is slightly different, with ECFP being superior in USPTO and ELN test sets for 3 classes and on inner test and USPTO for 4

Table 6.5 Results from amide coupling trained reagents filtered datasets, 3 classes. On sides the training data sources.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.507	0.266	0.505	0.305	**0.538**	0.361	0.476	0.259
	Reaxys	**0.398**	0.116	0.383	0.144	0.348	0.041	0.335	0.013
	ELN	0.366	0.056	**0.388**	0.117	0.342	0.017	0.335	0.017
	Enamine	0.345	0.019	0.331	-0.008	**0.350**	0.038	0.335	0.012
Reaxys	Inner test	**0.561**	0.332	0.544	0.416	0.548	0.411	0.53	0.381
	USPTO	**0.556**	0.33	0.439	0.284	0.333	0.01	0.359	0.076
	ELN	0.387	0.098	0.39	0.116	0.355	0.062	**0.428**	0.195
	Enamine	0.347	0.027	0.329	-0.012	0.333	-0.003	**0.361**	0.068
ELN	Inner test	0.559	0.295	0.544	0.341	**0.614**	0.491	0.562	0.341
	Libs test	0.354	0.023	0.334	0.003	0.321	-0.029	**0.375**	0.104
	USPTO	**0.380**	0.073	0.354	0.106	0.333	0.005	0.334	0.009
	Reaxys	0.384	0.087	0.339	0.051	0.334	0.003	**0.390**	0.174
	Enamine	0.344	0.013	0.333	0.000	0.335	0.018	**0.348**	0.032

classes. For 3 classes, Chemprop was the best on the inner test; for the rest, Enamine in 4 classes and ELN and Enamine in 3 classes, Yield-BERT performed best. For ELN models for $\mathrm{S_nAr}$ reaction, there is variability in different models performing best. ECFP performed best in USPTO for both 3 and 4 classes, as well as for Enamine in 3 classes. For the inner test, Chemprop performed best in both cases. For the rest of the performances, Yield-BERT was superior to other models.

In Suzuki couplings, ECFP has the best performance in Reaxys and ELN test sets for USPTO-trained models in bot 3 and 4 classes cases, also it demonstrated superior perfromance on USPTO test set for Reaxys-trained models. In the case of ELN-trained models, ECFP was performing best in the USPTO test set. Kallisto performed best only on an inner test set of USPTO 4 classes. Chemprop performed best on an inner ELN and USPTO test set in both classification cases and libraries in the 3-class case. Yield-BERT performed best on an inner test set of USPTO in the 3-class case and on the ELN test set in the Reaxys-trained model. And vice versa, on Reaxys test set in ELN-trained models.

In reductive amination, the picture of performances is almost the same, with the same models performing best on the same datasets and some minor differences, such as Kallisto performing better on Reaxys and ELN for the 3-class case and Yield-BERT having the best performance on the inner test set. Also, Yield-BERT was best on the library test set in both classification cases.

For Buchwald-Hartwig amination, ECFP is dominant in performances on test sets, with only Kallisto performing better on an inner test set of USPTO in both classification cases, Chemprop having better performance in inner test sets of Reaxys for the 4-class cases and ELN for both classification cases. Yield-BERT performed better for the library test and the Reaxys test set for the model trained on ELN.

Comparing yield-cleaned results to the models trained on more clean data, which included reagent cleaning, one can see that only Yield-BERT's performance improved slightly. Other models overall decreased performance by some minor negligible differences in BA and MCC, with most decrease provided by Chemprop, which for the "dirty" mode of training was trained without reagents included, as well as Yield-BERT also didn't have reagents, only bare reaction string. For reagent-cleaned data,

Table 6.6 Results from amide coupling trained on fully filtered ELN and evaluated on the inner test set of ELN and fully cleaned USPTO.

		3 classes				4 classes			
		RF ECFP		RF Kallisto		RF ECFP		RF Kallisto	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
ELN	Inner test	**0.546**	0.273	0.532	0.319	0.445	0.214	**0.447**	0.274
	USPTO	**0.388**	0.078	0.348	0.089	**0.283**	0.046	0.272	0.046

these two models were trained with reagents; for Yield-BERT, I included the essential reagent into the reaction string, and for Chemprop, I included it as a separate reagent SMILES. Interestingly, including reagents in Chemprop is a worsening factor, meaning this information introduces more noise than is useful for the model. Other models, such as ECFP and Kallisto-trained, haven't seen any improvement compared to the less clean data, meaning that this cleaning did not help to eliminate the inherent noise of the data as I would have expected.

Despite extensive data cleaning, the improvements in model performance are modest and inconsistent across different models and test sets. This highlights the problem's complexity and suggests a missing point besides model architecture and the "purity" of the data.

Results for purification-cleaned data

In this section, I compare the results of two datasets after the full cleaning pipeline. The raw text of the reaction procedure is available in ELN and USPTO datasets, so comparing the reactions filtered by purification keyphrases is done on these. I present results on the Random Forest model with two fingerprints and only ELN-trained models.

For the 3-class case, ECFP achieved a BA of 0.546 and an MCC of 0.273; Kallisto has a BA of 0.532 and an MCC of 0.319, showing a slightly better correlation than RF ECFP within the ELN dataset. On the USPTO test set ECFP, a BA of 0.388 and an MCC of 0.078 were recorded, indicating some generalization ability but with limited predictive power. Kallisto has a BA of 0.348 and an MCC of 0.089, slightly underperforming RF ECFP in terms of balanced accuracy but having a marginally better MCC.

For the 4-class classification, both ECFP and Kallisto achieved similar BA of 0.445 and 0.447, with Kallisto having a slightly higher MCC of 0.274, compared to ECFP, 0.214, showing reasonable performance within the ELN dataset. ECFP has a BA of 0.283 and an MCC of 0.046, and Kallisto's BA of 0.272 and an MCC of 0.046; both fingerprints' performances indicate poor generalization of the USPTO test set.

To summarize the findings, it is evident that, compared to the previous steps in the cleaning pipeline, the filtering of datasets slightly diminished the performance on the internal test set while providing only marginal improvements in generalization to the USPTO dataset. Although the rationale behind this filtering was theoretically sound, it did not lead to an enhancement in predictive power as anticipated.

6.4.3 Regression models

I performed regression studies on the reagents-cleaned datasets. I show the results in the table 6.7 and other results are in the Appendix. To access regression performance, I show RMSE, which indicates how far the predicted values are from the actual values, and R^2 metrics, which indicate how well the model explains the variance in the data. As one can see, each model tends to perform best on its inner test set, similar to earlier classification assessments, indicating good performance when the test data comes from the same source as the training data. For example, RF Kallisto shows the best RMSE and R^2 on its inner test set when trained on USPTO with RMSE 21.444 and R^2 0.271 and Reaxys with

Table 6.7 Results for regression models from amide coupling trained on reagents filtered datasets.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2
USPTO	Inner test	22.186	0.223	**21.444**	0.271	21.99	0.222	24.586	0.057
	Reaxys	**23.287**	0.05	23.648	0.02	29.203	-0.493	23.933	-0.003
	ELN	27.806	-0.164	27.165	-0.114	38.693	-1.26	**25.786**	-0.001
	Enamine	27.321	-0.087	27.066	-0.021	37.338	-1.03	**26.226**	-0.002
Reaxys	Inner test	20.147	0.294	**19.298**	0.349	19.306	0.334	20.280	0.286
	USPTO	**22.467**	0.208	23.917	0.104	39.254	-1.474	24.066	-0.474
	ELN	27.784	-0.162	27.615	-0.151	32.821	-0.626	**26.053**	-0.022
	Enamine	27.758	-0.122	**27.274**	-0.037	32.039	-0.495	28.666	-0.197
ELN	Inner test	21.465	0.308	20.239	0.379	**19.827**	0.464	21.791	0.290
	Libs test	21.512	0.32	**20.431**	0.388	27.448	-0.633	21.509	-0.000
	USPTO	26.781	-0.126	26.952	-0.138	48.212	-2.732	**25.137**	0.008
	Reaxys	26.793	-0.257	27.924	-0.366	32.869	-0.892	**24.763**	-0.074
	Enamine	27.225	-0.079	28.208	-0.109	47.359	-2.266	**26.753**	-0.042

RMSE 19.298 and R^2 0.349. If one looks at performance on external test sets, it is evident that models generally perform poorly; for instance, Chemprop shows significantly higher RMSE and lower R^2 on external test sets like USPTO and Reaxys compared to its inner test set. Several negative R^2 values across different models and test sets indicate poor model fits, where the model performs worse than a simple mean-based model, for example, Chemprop on Reaxys test set with R^2 -1.474 and Yield-BERT on Reaxys test set with R^2 -0.474.

One also could notice a pattern we have seen before in classification: Reaxys-trained models perform decently on USPTO datasets with ECFP fingerprints, having R^2 0.2. This is also true for other reactions, supporting my suggestion about data overlap in Reaxys and USPTO. Another interesting finding was that Kallisto fingerprints performed best at the Libraries test with R^2 0.3 on all reactions. This is interesting since this fingerprint did not show such superiority in classification settings.

These results again show evidence that something is missing from the predictions despite the data being similarly distributed in the feature space, as seen in Figure 6.3.

6.5 Discussion

To summarize all findings, I would say that none of the DL methods beat the baseline ECFP, and the hybrid Kallisto fingerprint also did not offer superiority over the baseline, except for some limited cases, such as library predictions in the regression setting. I hypothesized that the elaborate cleaning pipeline would help eliminate the noise in the data, but it did not improve the performance.

The modeling results reveal significant transferability challenges between datasets from different origins, even when they occupy similar chemical spaces (see Figure 6.3). The similar performance of various models and their limited transferability indicate an inherent noise problem within the datasets, which current state-of-the-art models cannot resolve. This outcome challenges the hypothesis that transferability within similar applicability domains should be feasible. The diversity and potential biases within these datasets appear to be greater than anticipated, impacting model performance significantly. Each dataset seems to contain unique elements that are not easily transferable, even with elaborate preprocessing aimed at homogenizing the data as much as possible.

This finding is significant and interesting because the data within each of the mentioned datasets is also not so homogeneous, as data from Reaxys is gathered from different articles with different methods and ways of yield recording. USPTO is also collected from different patents from various companies, which could have also been reporting yield differently. AstraZeneca's ELN data is also quite non-homogeneous. This reflection on the data makes a reasonable question stand up - what's the crucial difference between these different datasets that causes models to have mediocre performance? Despite the reactions being from a similar domain of applicability, why can the models not extrapolate to other data? Do models even learn something useful? There are no ways to test it except to try to predict test reaction yield, which they render fruitless. These questions will be left open since there are numerous possibilities of what could have gone wrong.

In my reagent-cleaned experiments, Yield-BERT and Chemprop had a theoretical advantage over Random Forest ECFP and Kallisto-trained models because the former included reagent information during training. However, their performance was either worse or on par with the ECFP and Kallisto models, which did not utilize reagent information. This finding underscores the limited efficacy of sophisticated deep learning (DL) models compared to simpler Random Forest models. While DL models excel in tasks like retrosynthesis or reaction mapping[1, 8, 185], they still fail to fully capture the chemical essence of a given reaction. This is also shown in their limited ability to predict relevant reagents for reactions as well[89, 94].

The results suggest that improving DL models to better capture the underlying chemistry is crucial. Developing better reaction descriptors may be necessary if this improvement is not feasible. The mixed approach of the Kallisto fingerprint, which considers hindrance and charges around reactive atoms, did not outperform the regular substructure-based ECFP fingerprint. This suggests that our current understanding and modeling of reactions might be missing critical aspects. Reactions are inherently multidimensional, and our current methods may be too shallow to see the deeper picture.

Working with reactions produced in more controlled and reproducible environments, such as automated synthesis, could be key to improving yield prediction models. Automated synthesis eliminates many untrackable parameters associated with manual reactions, making yield prediction more feasible. However, this approach limits the application to scenarios with such controlled conditions, rendering real-world yield prediction in diverse data contexts unattainable.

7 Discussion

> Don't Panic.
>
> Douglas Adams, *The Hitchhiker's Guide to the Galaxy*

In this work, I explored the potential of applying Machine Learning (ML) methods to predict reaction yield, a crucial metric for synthesis success. I hypothesized that ML models could learn reactivity patterns when trained on diverse, high-quality curated datasets and make relevant predictions on new data. Additionally, I posited that models trained on data from one time period could extrapolate to another time period, representing another form of transferability. Data transferability is important because it could show that the models trained on differently derived data are robust and reliable for predicting yields. This hypothesis was tested throughout my thesis using different models, datasets, data preprocessing methods, and data encodings.

The results of my experiments highlight significant challenges in transferring models trained on one dataset to another. Initially, I encountered difficulties with Buchwald-Hartwig reaction datasets from different sources. Despite representing the same reaction type, these datasets did not perform well when models trained on one were tested on another due to differences in the encoding of the reagents and lack of standardization. Similarly, two time-split Enamine datasets posed challenges, and the advanced BEE model could not capture the data intricacies. Even though these datasets came from the same vendor, they failed to work interchangeably, suggesting substantial variability or hidden biases even within a single source or lack of the model's generalizability. In Chapter 6, Models did not manage to demonstrate consistent performance across different datasets containing the same reaction but sourced from various origins. Despite uniformizing the reactions and the reagents, none of the models and descriptors showed acceptable performance. This lack of consistent performance underscores chemical reactions' complex and multifaceted nature and the data representing them.

The results show the current limitation of developing any general yield prediction model for the same reaction type, even if the data is derived from the same source. I also recognize that the current methods for yield prediction are likely effective only on datasets derived from high-throughput experimentation. It is challenging to achieve meaningful predictions from models trained on real-world data due to the inconsistencies in maintaining identical conditions, such as isolation and purification processes. Both experimental variability and poor recording contribute to this. Since models and data are bound together, we need improvements from both sides to progress in the area.

To advance the field, we must reconsider the possibility of developing general yield prediction models that can utilize various available data types, such as data derived from articles, electronic notebooks, and HTE. Yield prediction is most meaningful for HTE data. At the same time, it requires understanding which recorded data is crucial for yield or reagent prediction that is likely specific to each reaction type. Some reactions may proceed well under various conditions, while others are highly sensitive and require extensive trial and error. This necessitates collaborative work between chemists, chemoinformaticians, and machine learning experts in designing new datasets, their properties, and their purposes since determining the yield is often not a priority for synthetic chemists, but the purity of the end compound and yield recording is pretty poor. At the same time, for regular experiments, it may be more practical to focus on predicting whether a reaction **will occur** to avoid missing potentially synthesizable compounds. However, improving the extraction of the data is still a valuable thing to consider since it could be helpful to refine other spheres of synthesis prediction, such as retrosynthesis and condition prediction. The field has the paradox of having limited high-quality data with maximum information available and an overwhelming amount of data with limited information. This paradox results in too little high-quality data from reliable sources and an abundance of data from unreliable sources containing minimal information. Major sources may include additional valuable information, but this data is often untrustworthy due to imperfections in parsing algorithms since the data is mostly text-mined and not recorded with the predefined structure in a database. These imperfections can significantly impact yield prediction, making it necessary to improve current reporting systems. Including additional valuable information such as spectra, making data more machine-readable, and standardizing reporting practices for academic datasets in repositories like the Open Reaction Database would be beneficial. Popularizing open-source data reporting, which is common in machine learning, could also improve data quality. Also, Large Language Models (LLMs) can extract necessary reaction data from sources such as articles and patents, helping to standardize and clean historical data for predictive modeling. These models can facilitate the extraction of crucial reaction data such as temperature, time, a complete list of reagents, and their addition order, which are vital for accurate yield prediction.

Regarding reaction encoding, it remains unclear whether advanced deep learning models offer significant advantages over simpler models with less advanced descriptors. This highlights the need for better descriptors that can more effectively capture the reactivity of reacting species, including reagents. Another modeling approach involves creating strong, limited applicability models derived from high-quality datasets and trained on advanced quantum mechanical descriptors. These models could predict various outputs, such as selectivity, go/no-go outcomes for reactions, and side product formation. Although these models would have limited applicability due to their localized nature, they would continuously refine themselves with each new data point.

Since yield prediction is a topic of narrow data availability, it is important to keep updating the current databases with the most precise data possible. Still, we should not overlook other important fields such as predicting the reagents, design of experiments, and general reaction feasibility that can utilize less precise data.

Chatbots and LLM agents represent an exciting and hot topic in chemoinformatics[226–228]. These advanced AI tools are increasingly being utilized to improve the parsing of scientific journals and extract meaningful data[229–231]. One promising application is the development of chatbots that serve as interfaces between chemists and trained models, aiding the decision-making process in experimental planning and analysis. As described earlier, small, precise reactivity models could be included as part of a larger pipeline, including a CASP interface with a user-friendly graphical interface for chemists. Additionally, a chatbot interface could make interacting with these models more user-friendly, especially for those who prefer it over command-line interfaces. Such a wrapper around models could streamline their use in everyday tasks.

LLMs pre-trained on large corpora of chemistry data have the potential to enhance their utility significantly. These models can understand and generate human-like text, making them suitable for extracting relevant information from various sources, including research articles, patents, and databases. This capability is crucial for standardizing and cleaning historical data, thus making it more suitable for predictive modeling. Augmenting regular LLMs with knowledge graph systems could further enhance their performance. Knowledge graphs enable data representation in a structured form, highlighting relationships and connections between different pieces of information. When integrated with LLMs, these systems can help the models find connections and reasonings in chemistry-relevant queries, making the AI's responses more coherent and contextually accurate.

Using chatbots to propose reaction conditions is a particularly promising area[232]. These AI-driven assistants could analyze a chemist's input, such as the desired reaction and available reagents, and suggest optimal conditions for the reaction based on the reported conditions in the text-mined literature. This could include recommendations on temperature, solvent, reaction time, and other critical parameters. Such a tool would be invaluable in high-throughput experimentation and everyday laboratory work, where quick and reliable suggestions can save time and resources.

In addition to suggesting reaction conditions, chatbots could help design new experiments by predicting possible outcomes and highlighting potential pitfalls. They could access and integrate data from previous experiments, ongoing research, and theoretical models to provide comprehensive guidance. This level of support would be especially beneficial in complex synthesis tasks and novel reaction explorations. Despite these advancements, significant challenges remain. Current chatbots and LLMs can struggle with the nuanced and complex nature of chemical data, and there is still a long way to go before they can fully understand and predict intricate chemical reactions as accurately as a human expert. However, ongoing research and development in this area hold great promise for the future of chemoinformatics, potentially transforming how chemists design and execute experiments.

In conclusion, while we are still far from asking a chatbot a vague question and receiving an answer corresponding to our deepest chemical thoughts and desires, the integration of advanced AI tools like LLMs and chatbots into chemoinformatics represents a promising frontier for enhancing the efficiency and accuracy of chemical synthesis.

8 Conclusion

Jigsaw falling into place
so there is nothing to explain.

Radiohead, *Jigsaw Falling into Place*

Despite the lack of significant breakthroughs in yield prediction, this work provides valuable contributions by comparing various datasets, proposing standardization pipelines for handling reaction data, and emphasizing the critical need for consistent and transparent reporting of reaction yields.

Although the results did not support the hypothesis on transferability between different datasets, the study demonstrates that even with meticulously cleaned data, current methods are insufficient for accurate yield prediction. This suggests that the challenges lie not within the models or descriptors but within the inherent complexity and the quality of the data itself. The findings highlight the need for a deeper understanding of what factors must be encoded to achieve reliable predictions. This indicates that the field requires innovative approaches and perhaps entirely new methodologies to address the multi-dimensional nature of chemical reaction yields.

The field of yield prediction remains highly challenging with limited advances. Several factors contribute to these difficulties, including biases in reporting, poor reporting quality, and inconsistencies in how data is recorded. Unlike the relatively straightforward prediction of qualitative metrics, such as biological activity for compounds with standardized reporting, yield reports lack such coherence. This inconsistency hampers the effectiveness of chemoinformaticians' efforts and underscores the need for improved data quality and standardization.

9 Acknowledgements

> [they] forced you to stand still
> stop moving
> to vegetate
> but you always wanted to dance
>
> Lunatic Soul, *Summoning Dance*

This journey was probably worse than an average PhD student's journey since you cannot stay calm and composed when your mother country is being wrenched by the war with russia. I would say that I would not wish anyone to experience PhD journey with that amount of shame, guilt, feeling imposter and lonely that I had, especially in the first half of this term, before arriving at AstraZeneca (please, fix your mental health *before* entering PhD journey and not during). In AstraZeneca, I met some people who facilitated my healing journey. And I am grateful to all of them simply for being there, showing that existence could be not painful, and thus, I could learn from their example.

But despite all the bad things, many good things happened, and I don't regret meeting all the amazing people on the way. I am still there. And I wrote a whole thesis, and I am somewhat sane and somewhat happy due to the extensive work of the next list of people: **Samuel Genheden**, my dearest supervisor and mentor, to whom I could write a thousand words of thanks for how meaningful and supportive our relationship was. Thanks to **Igor Tetko** for allowing me to join the AIDD project and meet all the amazing people on the way, for supervising the first half of my PhD, and for all the great input to my project. Thanks to **Dmytro Dudenko** for your supervision and helpful discussion. Thanks to **Michael Sattler** for his recommendations and support. Thanks to **Ola Engkvist** for accepting me as a "refugee" student into AstraZeneca. **Peter Hartog**, my fellow PhD student who was a great support in the war's first months, accepted my misery and still being my friend. **Paula Torren Perraire**, also my fellow PhD student who was a great support and a great friend. **Mikhail Kabeshov** who opened up my scientific spirit of endless experimentation for the sake of finding out something cool. **Thomas Löhr** - a person who interprets reality similarly to me and is my friend (also thanks for all the bouldering+coffee sessions:). My best friend, **Dmytro Mitichkin**, for all his support. My psychotherapist, **Viktoria**, helped me not to drown in myself. My **mom and dad** were a great support to heal my imposter syndrome and made me believe I was worth something. Last but not least, **Ivan Oleksiyuk**, my dearest husband who was in the same boat as I am with our reciprocality in PhD journey. You are my cloudy sun. I also thank all my colleagues for all the amazing coffee breaks, talks, and after work. I also thank all the PhDs in the AIDD and AiChemist program for our shared frustration. I also thank the **Armed Forces of Ukraine** that my parents, relatives, and friends are alive.

I thank the proofreaders of this thesis, Lukas Sigmund, Mikhail Kabeshov, and Thomas Löhr.

Some honorable mentions include the Munich Botanical Garden and the Gothenburg Botanical Garden. These places helped me to balance myself, and observing beautiful plants gave me peace and grace.

I also thank the European Union for allowing me to study this doctoral project under grant agreement No 956832 in Horizon 2020 Marie Skłodowska-Curie Innovative Training Network - European Industrial Doctorate.

The project leading to this thesis has 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.

10 List of publications

Voinarovska, V., Kabeshov, M., Dudenko, D., Genheden, S., and Tetko, I. V. (2023). When yield prediction does not yield prediction: An overview of the current challenges. Journal of Chemical Information and Modeling, 64(1), 42-56. https://doi.org/10.1021/acs.jcim.3c01524

Andronov, M., Voinarovska, V., Andronova, N., Wand, M., Clevert, D.-A., and Schmidhuber, J. (2023). Reagent prediction with a molecular transformer improves reaction data quality. Chemical Science, 14(12), 3235-3246. https://doi.org/10.1039/d2sc06798f

11 Code availability

GitHub for the Chapter 4:
`https://github.com/v-in-cube/YieldnotYield`
GitHub for the Chapter 6:
`https://github.com/v-in-cube/multi_class_yield`

Bibliography

(1) Schwaller, P.; Hoover, B.; Reymond, J.-L.; Strobelt, H.; Laino, T. Extraction of organic chemistry grammar from unsupervised learning of chemical reactions. *Sci. Adv.* **2021**, *7*, DOI: 10.1126/sciadv.abe4166.

(2) Lowe, D. Chemical reactions from US patents (1976-Sep2016), Artwork Size: 1494665893 Bytes Pages: 1494665893 Bytes Type: dataset, 2017, DOI: 10.6084/M9.FIGSHARE.5104873.V1.

(3) Federal Drug Administration, https://www.fda.gov/drugs/development-approval-process-drugs/drug-approvals-and-databases, accessed on 02.05.2024.

(4) Gupta, R.; Srivastava, D.; Sahu, M.; Tiwari, S.; Ambasta, R. K.; Kumar, P. Artificial intelligence to deep learning: machine intelligence approach for drug discovery. *Molecular Diversity* **2021**, *25*, 1315–1360, DOI: 10.1007/s11030-021-10217-3.

(5) Brown, D. G.; Boström, J. Analysis of Past and Present Synthetic Methodologies on Medicinal Chemistry: Where Have All the New Reactions Gone?: Miniperspective. *Journal of Medicinal Chemistry* **2015**, *59*, 4443–4458, DOI: 10.1021/acs.jmedchem.5b01409.

(6) Rosales, A. R.; Quinn, T. R.; Wahlers, J.; Tomberg, A.; Zhang, X.; Helquist, P.; Wiest, O.; Norrby, P.-O. Application of Q2MM to predictions in stereoselective synthesis. *Chemical Communications* **2018**, *54*, 8294–8311, DOI: 10.1039/c8cc03695k.

(7) Tu, Z.; Stuyver, T.; Coley, C. W. Predictive chemistry: machine learning for reaction deployment, reaction development, and reaction discovery. *Chemical Science* **2023**, *14*, 226–244, DOI: 10.1039/d2sc05089g.

(8) Schwaller, P.; Vaucher, A. C.; Laplaza, R.; Bunne, C.; Krause, A.; Corminboeuf, C.; Laino, T. Machine intelligence for chemical reaction space. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2022**, *12*, DOI: 10.1002/wcms.1604.

(9) Meyers, J.; Fabian, B.; Brown, N. De novo molecular design and generative models. *Drug Discovery Today* **2021**, *26*, 2707–2715, DOI: 10.1016/j.drudis.2021.05.019.

(10) Blaschke, T.; Arús-Pous, J.; Chen, H.; Margreitter, C.; Tyrchan, C.; Engkvist, O.; Papadopoulos, K.; Patronov, A. REINVENT 2.0: An AI Tool for De Novo Drug Design. *Journal of Chemical Information and Modeling* **2020**, *60*, 5918–5922, DOI: 10.1021/acs.jcim.0c00915.

(11) Ståhl, N.; Falkman, G.; Karlsson, A.; Mathiason, G.; Boström, J. Deep Reinforcement Learning for Multiparameter Optimization in de novo Drug Design. *Journal of Chemical Information and Modeling* **2019**, *59*, 3166–3176, DOI: 10.1021/acs.jcim.9b00325.

(12) Loeffler, H. H.; He, J.; Tibo, A.; Janet, J. P.; Voronov, A.; Mervin, L. H.; Engkvist, O. Reinvent 4: Modern AI–driven generative molecule design. *Journal of Cheminformatics* **2024**, *16*, DOI: 10.1186/s13321-024-00812-5.

(13) Boström, J.; Brown, D. G.; Young, R. J.; Keserü, G. M. Expanding the medicinal chemistry synthetic toolbox. *Nature Reviews Drug Discovery* **2018**, *17*, 709–727, DOI: 10.1038/nrd.2018.116.

(14) Jiang, Y.; Yu, Y.; Kong, M.; Mei, Y.; Yuan, L.; Huang, Z.; Kuang, K.; Wang, Z.; Yao, H.; Zou, J.; Coley, C. W.; Wei, Y. Artificial Intelligence for Retrosynthesis Prediction. *Engineering* **2023**, *25*, 32–50, DOI: 10.1016/j.eng.2022.04.021.

(15) Coley, C. W.; Barzilay, R.; Jaakkola, T. S.; Green, W. H.; Jensen, K. F. Prediction of Organic Reaction Outcomes Using Machine Learning. *ACS Central Science* **2017**, *3*, 434–443, DOI: 10.1021/acscentsci.7b00064.

(16) Watson, I. A.; Wang, J.; Nicolaou, C. A. A retrosynthetic analysis algorithm implementation. *Journal of Cheminformatics* **2019**, *11*, DOI: 10.1186/s13321-018-0323-6.

(17) 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. *Journal of Cheminformatics* **2020**, *12*, DOI: 10.1186/s13321-020-00472-1.

(18) Malo, N.; Hanley, J. A.; Cerquozzi, S.; Pelletier, J.; Nadon, R. Statistical practice in high-throughput screening data analysis. *Nature Biotechnology* **2006**, *24*, 167–175, DOI: 10.1038/nbt1186.

(19) Obrezanova, O.; Martinsson, A.; Whitehead, T.; Mahmoud, S.; Bender, A.; Miljković, F.; Grabowski, P.; Irwin, B.; Oprisiu, I.; Conduit, G.; Segall, M.; Smith, G. F.; Williamson, B.; Winiwarter, S.; Greene, N. Prediction of In Vivo Pharmacokinetic Parameters and Time–Exposure Curves in Rats Using Machine Learning from the Chemical Structure. *Molecular Pharmaceutics* **2022**, *19*, 1488–1504, DOI: 10.1021/acs.molpharmaceut.2c00027.

(20) Colclough, N.; Wenlock, M. C. Interpreting physicochemical experimental data sets. *Journal of Computer-Aided Molecular Design* **2015**, *29*, 779–794, DOI: 10.1007/s10822-015-9850-7.

(21) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; Polosukhin, I. In *Adv. Neural Inf. Process. Syst.* Ed. by Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; Garnett, R., Curran Associates, Inc.: 2017; Vol. 30.

(22) Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P. J. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, 2019, DOI: 10.48550/ARXIV.1910.10683.

(23) Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2018, DOI: 10.48550/ARXIV.1810.04805.

(24) Reimers, N.; Gurevych, I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, 2019, DOI: 10.48550/ARXIV.1908.10084.

(25) Liu, P. J.; Saleh, M.; Pot, E.; Goodrich, B.; Sepassi, R.; Kaiser, L.; Shazeer, N. Generating Wikipedia by Summarizing Long Sequences, 2018, DOI: 10.48550/ARXIV.1801.10198.

(26) Kryscinski, W.; Keskar, N. S.; McCann, B.; Xiong, C.; Socher, R. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Association for Computational Linguistics: 2019, DOI: 10.18653/v1/d19-1051.

(27) OpenAI; Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; Avila, R.; Babuschkin, I.; Balaji, S.; Balcom, V.; Baltescu, P.; Bao, H.; Bavarian, M.; Belgum, J.; Bello, I., et al. GPT-4 Technical Report, 2023, DOI: 10.48550/ARXIV.2303.08774.

(28) Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Žídek, A.; Potapenko, A.; Bridgland, A.; Meyer, C.; Kohl, S. A. A.; Ballard, A. J.; Cowie, A.; Romera-Paredes, B.; Nikolov, S.; Jain, R.; Adler, J.; Back, T., et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583–589, DOI: 10.1038/s41586-021-03819-2.

(29) 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, DOI: 10.1021/acscentsci.9b00576.

(30) Kipf, T. N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks, 2016, DOI: 10.48550/ARXIV.1609.02907.

(31) Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. In *Proceedings of the 34th International Conference on Machine Learning*, ed. by Precup, D.; Teh, Y. W., PMLR: 2017; Vol. 70, pp 1263–1272.

(32) Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph Attention Networks, 2017, DOI: 10.48550/ARXIV.1710.10903.

(33) Duvenaud, D.; Maclaurin, D.; Aguilera-Iparraguirre, J.; Gómez-Bombarelli, R.; Hirzel, T.; Aspuru-Guzik, A.; Adams, R. P. Convolutional Networks on Graphs for Learning Molecular Fingerprints, 2015, DOI: 10.48550/ARXIV.1509.09292.

(34) Sutton, R.; Barto, A. Reinforcement Learning: An Introduction. *IEEE Transactions on Neural Networks* **1998**, *9*, 1054–1054, DOI: 10.1109/tnn.1998.712192.

(35) Olivecrona, M.; Blaschke, T.; Engkvist, O.; Chen, H. Molecular de-novo design through deep reinforcement learning. *Journal of Cheminformatics* **2017**, *9*, DOI: 10.1186/s13321-017-0235-x.

(36) Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning* **1992**, *8*, 229–256, DOI: 10.1007/bf00992696.

(37) Snoek, J.; Larochelle, H.; Adams, R. P. Practical Bayesian Optimization of Machine Learning Algorithms, 2012, DOI: 10.48550/ARXIV.1206.2944.

(38) Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R. P.; de Freitas, N. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proceedings of the IEEE* **2016**, *104*, 148–175, DOI: 10.1109/jproc.2015.2494218.

(39) Shields, B. J.; Stevens, J.; Li, J.; Parasram, M.; Damani, F.; Alvarado, J. I. M.; Janey, J. M.; Adams, R. P.; Doyle, A. G. Bayesian reaction optimization as a tool for chemical synthesis. *Nature* **2021**, *590*, 89–96, DOI: 10.1038/s41586-021-03213-y.

(40) Corey, E. J.; Long, A. K.; Rubenstein, S. D. Computer-Assisted Analysis in Organic Synthesis. *Science* **1985**, *228*, 408–418, DOI: 10.1126/science.3838594.

(41) Corey, E. J. General methods for the construction of complex molecules. *Pure and Applied Chemistry* **1967**, *14*, 19–38, DOI: 10.1351/pac196714010019.

(42) Corey, E. J.; Wipke, W. T.; Cramer, R. D.; Howe, W. J. Computer-assisted synthetic analysis. Facile man-machine communication of chemical structure by interactive computer graphics. *Journal of the American Chemical Society* **1972**, *94*, 421–430, DOI: 10.1021/ja00757a020.

(43) Plehiers, P. P.; Coley, C. W.; Gao, H.; Vermeire, F. H.; Dobbelaere, M. R.; Stevens, C. V.; Van Geem, K. M.; Green, W. H. Artificial Intelligence for Computer-Aided Synthesis In Flow: Analysis and Selection of Reaction Components. *Frontiers in Chemical Engineering* **2020**, *2*, DOI: 10.3389/fceng.2020.00005.

(44) Thakkar, A.; Johansson, S.; Jorner, K.; Buttar, D.; Reymond, J.-L.; Engkvist, O. Artificial intelligence and automation in computer aided synthesis planning. *Reaction Chemistry and Engineering* **2021**, *6*, 27–51, DOI: 10.1039/d0re00340a.

(45) Szymkuć, S.; Gajewska, E. P.; Klucznik, T.; Molga, K.; Dittwald, P.; Startek, M.; Bajczyk, M.; Grzybowski, B. A. Computer-Assisted Synthetic Planning: The End of the Beginning. *Angewandte Chemie International Edition* **2016**, *55*, 5904–5937, DOI: 10.1002/anie.201506101.

(46) Klucznik, T.; Mikulak-Klucznik, B.; McCormack, M. P.; Lima, H.; Szymkuć, S.; Bhowmick, M.; Molga, K.; Zhou, Y.; Rickershauser, L.; Gajewska, E. P.; Toutchkine, A.; Dittwald, P.; Startek, M. P.; Kirkovits, G. J.; Roszak, R.; Adamski, A.; Sieredzińska, B.; Mrksich, M.; Trice, S. L.; Grzybowski, B. A. Efficient Syntheses of Diverse, Medicinally Relevant Targets Planned by Computer and Executed in the Laboratory. *Chem* **2018**, *4*, 522–532, DOI: 10.1016/j.chempr.2018.02.002.

(47) Molga, K.; Szymkuć, S.; Grzybowski, B. A. Chemist Ex Machina: Advanced Synthesis Planning by Computers. *Accounts of Chemical Research* **2021**, *54*, 1094–1106, DOI: 10.1021/acs.accounts.0c00714.

(48) SMIRKS, https://www.daylight.com/dayhtml/doc/theory/theory.smirks.html, accessed on 02.08.2022.

(49) Dong, J.; Zhao, M.; Liu, Y.; Su, Y.; Zeng, X. Deep learning in retrosynthesis planning: datasets, models and tools. *Briefings in Bioinformatics* **2021**, *23*, DOI: 10.1093/bib/bbab391.

(50) Segler, M. H. S.; Kogej, T.; Tyrchan, C.; Waller, M. P. Generating Focused Molecule Libraries for Drug Discovery with Recurrent Neural Networks. *ACS Central Science* **2017**, *4*, 120–131, DOI: 10.1021/acscentsci.7b00512.

(51) Struble, T. J.; Alvarez, J. C.; Brown, S. P.; Chytil, M.; Cisar, J.; DesJarlais, R. L.; Engkvist, O.; Frank, S. A.; Greve, D. R.; Griffin, D. J.; Hou, X.; Johannes, J. W.; Kreatsoulas, C.; Lahue, B.; Mathea, M.; Mogk, G.; Nicolaou, C. A.; Palmer, A. D.; Price, D. J.; Robinson, R. I., et al. Current and Future Roles of Artificial Intelligence in Medicinal Chemistry Synthesis. *Journal of Medicinal Chemistry* **2020**, *63*, 8667–8682, DOI: 10.1021/acs.jmedchem.9b02120.

(52) Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Model.* **1988**, *28*, 31–36, DOI: 10.1021/ci00057a005.

(53) Liu, B.; Ramsundar, B.; Kawthekar, P.; Shi, J.; Gomes, J.; Luu Nguyen, Q.; Ho, S.; Sloane, J.; Wender, P.; Pande, V. Retrosynthetic Reaction Prediction Using Neural Sequence-to-Sequence Models. *ACS Central Science* **2017**, *3*, 1103–1113, DOI: 10.1021/acscentsci.7b00303.

(54) Karpov, P.; Godin, G.; Tetko, I. V. In *Artificial Neural Networks and Machine Learning – ICANN 2019: Workshop and Special Sessions*; Springer International Publishing: 2019, pp 817–830, DOI: 10.1007/978-3-030-30493-5_78.

(55) Ivanenkov, Y. A.; Zhebrak, A.; Bezrukov, D.; Zagribelnyy, B.; Aladinskiy, V.; Polykovskiy, D.; Putin, E.; Kamya, P.; Aliper, A.; Zhavoronkov, A. Chemistry42: An AI-based platform for de novo molecular design, 2021, DOI: 10.48550/ARXIV.2101.09050.

(56) Coley, C. W.; Rogers, L.; Green, W. H.; Jensen, K. F. Computer-Assisted Retrosynthesis Based on Molecular Similarity. *ACS Central Science* **2017**, *3*, 1237–1245, DOI: 10.1021/acscentsci.7b00355.

(57) Schwaller, P.; Gaudin, T.; Lányi, D.; Bekas, C.; Laino, T. “Found in Translation”: predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. *Chemical Science* **2018**, *9*, 6091–6098, DOI: 10.1039/c8sc02339e.

(58) Segler, M. H. S.; Preuss, M.; Waller, M. P. Planning chemical syntheses with deep neural networks and symbolic AI. *Nature* **2018**, *555*, 604–610, DOI: 10.1038/nature25978.

(59) Lin, A.; Dyubankova, N.; Madzhidov, T. I.; Nugmanov, R. I.; Verhoeven, J.; Gimadiev, T. R.; Afonina, V. A.; Ibragimova, Z.; Rakhimbekova, A.; Sidorov, P.; Gedich, A.; Suleymanov, R.; Mukhametgaleev, R.; Wegner, J.; Ceulemans, H.; Varnek, A. Atom-to-atom Mapping: A Benchmarking Study of Popular Mapping Algorithms and Consensus Strategies. *Molecular Informatics* **2021**, *41*, DOI: 10.1002/minf.202100138.

(60) Schwaller, P.; Petraglia, R.; Zullo, V.; Nair, V. H.; Haeuselmann, R. A.; Pisoni, R.; Bekas, C.; Iuliano, A.; Laino, T. Predicting retrosynthetic pathways using transformer-based models and a hyper-graph exploration strategy. *Chemical Science* **2020**, *11*, 3316–3325, DOI: 10.1039/c9sc05704h.

(61) Hassen, A. K.; Torren-Peraire, P.; Genheden, S.; Verhoeven, J.; Preuss, M.; Tetko, I. Mind the Retrosynthesis Gap: Bridging the divide between Single-step and Multi-step Retrosynthesis Prediction, 2022, DOI: 10.48550/ARXIV.2212.11809.

(62) Salatin, T. D.; Jorgensen, W. L. Computer-assisted mechanistic evaluation of organic reactions. 1. Overview. *The Journal of Organic Chemistry* **1980**, *45*, 2043–2051, DOI: 10.1021/jo01299a001.

(63) Jorgensen, W. L.; Laird, E. R.; Gushurst, A. J.; Fleischer, J. M.; Gothe, S. A.; Helson, H. E.; Paderes, G. D.; Sinclair, S. CAMEO: a program for the logical prediction of the products of organic reactions. *Pure and Applied Chemistry* **1990**, *62*, 1921–1932, DOI: 10.1351/pac199062101921.

(64) Ugi, I.; Bauer, J.; Baumgartner, R.; Fontain, E.; Forstmeyer, D.; Lohberger, S. Computer assistance in the design of syntheses and a new generation of computer programs for the solution of chemical problems by molecular logic. *Pure and Applied Chemistry* **1988**, *60*, 1573–1586, DOI: 10.1351/pac198860111573.

(65) Bauer, J.; Fontain, E.; Forstmeyer, D.; Ugi, I. Interactive generation of organic reactions by IGOR 2 and the PC-assisted discovery of a new reaction. *Tetrahedron Computer Methodology* **1988**, *1*, 129–132, DOI: 10.1016/0898-5529(88)90017-6.

(66) Bauer, J. IGOR2: a PC-program for generating new reactions and molecular structures. *Tetrahedron Computer Methodology* **1989**, *2*, 269–280, DOI: 10.1016/0898-5529(89)90034-1.

(67) Gasteiger, J.; Ihlenfeldt, W. D.; Röse, P. A collection of computer methods for synthesis design and reaction prediction. *Recueil des Travaux Chimiques des Pays-Bas* **1992**, *111*, 270–290, DOI: 10.1002/recl.19921110605.

(68) Röse, P.; Gasteiger, J. Automated derivation of reaction rules for the EROS 6.0 system for reaction prediction. *Analytica Chimica Acta* **1990**, *235*, 163–168, DOI: 10.1016/s0003-2670(00)82071-1.

(69) Zefirov, N. S.; Baskin, I. I.; Palyulin, V. A. SYMBEQ Program and Its Application in Computer-Assisted Reaction Design. *Journal of Chemical Information and Computer Sciences* **1994**, *34*, 994–999, DOI: 10.1021/ci00020a038.

(70) Chen, J. H.; Baldi, P. No Electron Left Behind: A Rule-Based Expert System To Predict Chemical Reactions and Reaction Mechanisms. *Journal of Chemical Information and Modeling* **2009**, *49*, 2034–2043, DOI: 10.1021/ci900157k.

(71) Kayala, M. A.; Azencott, C.-A.; Chen, J. H.; Baldi, P. Learning to Predict Chemical Reactions. *Journal of Chemical Information and Modeling* **2011**, *51*, 2209–2222, DOI: 10.1021/ci200207y.

(72) Kayala, M. A.; Baldi, P. ReactionPredictor: Prediction of Complex Chemical Reactions at the Mechanistic Level Using Machine Learning. *Journal of Chemical Information and Modeling* **2012**, *52*, 2526–2540, DOI: 10.1021/ci3003039.

(73) Wei, J. N.; Duvenaud, D.; Aspuru-Guzik, A. Neural Networks for the Prediction of Organic Chemistry Reactions. *ACS Central Science* **2016**, *2*, 725–732, DOI: 10.1021/acscentsci.6b00219.

(74) Segler, M. H. S.; Waller, M. P. Neural-Symbolic Machine Learning for Retrosynthesis and Reaction Prediction. *Chem. - Eur. J.* **2017**, *23*, 5966–5971, DOI: 10.1002/chem.201605499.

(75) Zaheer, M.; Kottur, S.; Ravanbakhsh, S.; Poczos, B.; Salakhutdinov, R. R.; Smola, A. J. Deep sets. *Advances in neural information processing systems* **2017**, *30*.

(76) Nikitin, F.; Isayev, O.; Strijov, V. DRACON: disconnected graph neural network for atom mapping in chemical reactions. *Physical Chemistry Chemical Physics* **2020**, *22*, 26478–26486, DOI: 10.1039/d0cp04748a.

(77) Coley, C. W.; Jin, W.; Rogers, L.; Jamison, T. F.; Jaakkola, T. S.; Green, W. H.; Barzilay, R.; Jensen, K. F. A graph-convolutional neural network model for the prediction of chemical reactivity. *Chem. Sci.* **2019**, *10*, 370–377, DOI: 10.1039/c8sc04228d.

(78) Bradshaw, J.; Kusner, M. J.; Paige, B.; Segler, M. H.; Hernández-Lobato, J. M. A generative model for electron paths. *arXiv preprint arXiv:1805.10970* **2018**.

(79) Do, K.; Tran, T.; Venkatesh, S. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp 750–760.

(80) Nam, J.; Kim, J. Linking the neural machine translation and the prediction of organic chemistry reactions. *arXiv preprint arXiv:1612.09529* **2016**.

(81) Qian, W. W.; Russell, N. T.; Simons, C. L. W.; Luo, Y.; Burke, M. D.; Peng, J. Integrating Deep Neural Networks and Symbolic Inference for Organic Reactivity Prediction. **2020**, DOI: 10.26434/chemrxiv.11659563.v1.

(82) Sacha, M.; Błaż, M.; Byrski, P.; Dąbrowski-Tumański, P.; Chromiński, M.; Loska, R.; Włodarczyk-Pruszyński, P.; Jastrzębski, S. Molecule Edit Graph Attention Network: Modeling Chemical Reactions as Sequences of Graph Edits. *Journal of Chemical Information and Modeling* **2021**, *61*, 3273–3284, DOI: 10.1021/acs.jcim.1c00537.

(83) Pesciullesi, G.; Schwaller, P.; Laino, T.; Reymond, J. L. Transfer learning enables the molecular transformer to predict regio- and stereoselective reactions on carbohydrates. *Nat. Commun. 2020 11:1* **2020**, *11*, 1–8, DOI: 10.1038/s41467-020-18671-7.

(84) Wang, L.; Zhang, C.; Bai, R.; Li, J.; Duan, H. Heck reaction prediction using a transformer model based on a transfer learning strategy. *Chemical Communications* **2020**, *56*, 9368–9371, DOI: 10.1039/d0cc02657c.

(85) Wu, Y.; Zhang, C.; Wang, L.; Duan, H. A graph-convolutional neural network for addressing small-scale reaction prediction. *Chemical Communications* **2021**, *57*, 4114–4117, DOI: 10.1039/d1cc00586c.

(86) Seidl, P.; Renz, P.; Dyubankova, N.; Neves, P.; Verhoeven, J.; Segler, M.; Wegner, J. K.; Hochreiter, S.; Klambauer, G. Modern hopfield networks for few-and zero-shot reaction template prediction. *arXiv preprint arXiv:2104.03279* **2021**.

(87) Struebing, H.; Ganase, Z.; Karamertzanis, P. G.; Siougkrou, E.; Haycock, P.; Piccione, P. M.; Armstrong, A.; Galindo, A.; Adjiman, C. S. Computer-aided molecular design of solvents for accelerated reaction kinetics. *Nature Chemistry* **2013**, *5*, 952–957, DOI: 10.1038/nchem.1755.

(88) Marcou, G.; Aires de Sousa, J.; Latino, D. A. R. S.; de Luca, A.; Horvath, D.; Rietsch, V.; Varnek, A. Expert System for Predicting Reaction Conditions: The Michael Reaction Case. *Journal of Chemical Information and Modeling* **2015**, *55*, 239–250, DOI: 10.1021/ci500698a.

(89) Gao, H.; Struble, T. J.; Coley, C. W.; Wang, Y.; Green, W. H.; Jensen, K. F. Using Machine Learning To Predict Suitable Conditions for Organic Reactions. *ACS Cent. Sci.* **2018**, *4*, 1465–1476, DOI: 10.1021/acscentsci.8b00357.

(90) Reaxys, https://www.reaxys.com/, accessed on 02.08.2022.

(91) Walker, E.; Kammeraad, J.; Goetz, J.; Robo, M. T.; Tewari, A.; Zimmerman, P. M. Learning To Predict Reaction Conditions: Relationships between Solvent, Molecular Structure, and Catalyst. *Journal of Chemical Information and Modeling* **2019**, *59*, 3645–3654, DOI: 10.1021/acs.jcim.9b00313.

(92) Maser, M. R.; Cui, A. Y.; Ryou, S.; DeLano, T. J.; Yue, Y.; Reisman, S. E. Multilabel Classification Models for the Prediction of Cross-Coupling Reaction Conditions. *Journal of Chemical Information and Modeling* **2021**, *61*, 156–166, DOI: 10.1021/acs.jcim.0c01234.

(93) Vaucher, A. C.; Schwaller, P.; Geluykens, J.; Nair, V. H.; Iuliano, A.; Laino, T. Inferring experimental procedures from text-based representations of chemical reactions. *Nature Communications* **2021**, *12*, DOI: 10.1038/s41467-021-22951-1.

(94) Andronov, M.; Voinarovska, V.; Andronova, N.; Wand, M.; Clevert, D.-A.; Schmidhuber, J. Reagent prediction with a molecular transformer improves reaction data quality. *Chemical Science* **2023**, *14*, 3235–3246, DOI: 10.1039/d2sc06798f.

(95) McMullen, J. P.; Jensen, K. F. Integrated Microreactors for Reaction Automation: New Approaches to Reaction Development. *Annual Review of Analytical Chemistry* **2010**, *3*, 19–42, DOI: 10.1146/annurev.anchem.111808.073718.

(96) Fabry, D. C.; Sugiono, E.; Rueping, M. Self-Optimizing Reactor Systems: Algorithms, On-line Analytics, Setups, and Strategies for Accelerating Continuous Flow Process Optimization. *Israel Journal of Chemistry* **2013**, *54*, 341–350, DOI: 10.1002/ijch.201300080.

(97) McMullen, J. P.; Stone, M. T.; Buchwald, S. L.; Jensen, K. F. An Integrated Microreactor System for Self-Optimization of a Heck Reaction: From Micro- to Mesoscale Flow Systems. *Angewandte Chemie International Edition* **2010**, *49*, 7076–7080, DOI: 10.1002/anie.201002590.

(98) Parrott, A. J.; Bourne, R. A.; Akien, G. R.; Irvine, D. J.; Poliakoff, M. Self-Optimizing Continuous Reactions in Supercritical Carbon Dioxide. *Angewandte Chemie International Edition* **2011**, *50*, 3788–3792, DOI: 10.1002/anie.201100412.

(99) Reizman, B. J.; Wang, Y.-M.; Buchwald, S. L.; Jensen, K. F. Suzuki–Miyaura cross-coupling optimization enabled by automated feedback. *Reaction Chemistry and Engineering* **2016**, *1*, 658–666, DOI: 10.1039/c6re00153j.

(100) Plutschack, M. B.; Pieber, B.; Gilmore, K.; Seeberger, P. H. The Hitchhiker's Guide to Flow Chemistry. *Chemical Reviews* **2017**, *117*, 11796–11893, DOI: 10.1021/acs.chemrev.7b00183.

(101) Echtermeyer, A.; Amar, Y.; Zakrzewski, J.; Lapkin, A. Self-optimisation and model-based design of experiments for developing a C–H activation flow process. *Beilstein Journal of Organic Chemistry* **2017**, *13*, 150–163, DOI: 10.3762/bjoc.13.18.

(102) Fitzpatrick, D. E.; Battilocchio, C.; Ley, S. V. A Novel Internet-Based Reaction Monitoring, Control and Autonomous Self-Optimization Platform for Chemical Synthesis. *Organic Process Research and Development* **2015**, *20*, 386–394, DOI: 10.1021/acs.oprd.5b00313.

(103) Holmes, N.; Akien, G. R.; Blacker, A. J.; Woodward, R. L.; Meadows, R. E.; Bourne, R. A. Self-optimisation of the final stage in the synthesis of EGFR kinase inhibitor AZD9291 using an automated flow reactor. *Reaction Chemistry and Engineering* **2016**, *1*, 366–371, DOI: 10.1039/c6re00059b.

(104) Krishnadasan, S.; Brown, R. J. C.; deMello, A. J.; deMello, J. C. Intelligent routes to the controlled synthesis of nanoparticles. *Lab on a Chip* **2007**, *7*, 1434, DOI: 10.1039/b711412e.

(105) Sans, V.; Porwol, L.; Dragone, V.; Cronin, L. A self optimizing synthetic organic reactor system using real-time in-line NMR spectroscopy. *Chemical Science* **2015**, *6*, 1258–1264, DOI: 10.1039/c4sc03075c.

(106) Močkus, J. In *Lecture Notes in Computer Science*; Springer Berlin Heidelberg: 1975, pp 400–404, DOI: 10.1007/3-540-07165-2_55.

(107) Liang, R.; Duan, X.; Zhang, J.; Yuan, Z. Bayesian based reaction optimization for complex continuous gas–liquid–solid reactions. *Reaction Chemistry and Engineering* **2022**, *7*, 590–598, DOI: 10.1039/d1re00397f.

(108) Karan, D.; Chen, G.; Jose, N.; Bai, J.; McDaid, P.; Lapkin, A. A. A machine learning-enabled process optimization of ultra-fast flow chemistry with multiple reaction metrics. *Reaction Chemistry and Engineering* **2024**, *9*, 619–629, DOI: 10.1039/d3re00539a.

(109) Zhang, J.; Sugisawa, N.; Felton, K. C.; Fuse, S.; Lapkin, A. A. Multi-objective Bayesian optimisation using q-noisy expected hypervolume improvement (qNEHVI) for the Schotten–Baumann reaction. *Reaction Chemistry and Engineering* **2024**, *9*, 706–712, DOI: 10.1039/d3re00502j.

(110) Zhou, Z.; Li, X.; Zare, R. N. Optimizing Chemical Reactions with Deep Reinforcement Learning. *ACS Central Science* **2017**, *3*, 1337–1344, DOI: 10.1021/acscentsci.7b00492.

(111) Guimaraes, G. L.; Sanchez-Lengeling, B.; Outeiral, C.; Farias, P. L. C.; Aspuru-Guzik, A. Objective-Reinforced Generative Adversarial Networks (ORGAN) for Sequence Generation Models, 2017, DOI: 10.48550/ARXIV.1705.10843.

(112) Sanchez-Lengeling, B.; Outeiral, C.; Guimaraes, G. L.; Aspuru-Guzik, A. Optimizing distributions over molecular space. An Objective-Reinforced Generative Adversarial Network for Inverse-design Chemistry (ORGANIC). **2017**, DOI: 10.26434/chemrxiv.5309668.v3.

(113) Yu, L.; Zhang, W.; Wang, J.; Yu, Y. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. *Proceedings of the AAAI Conference on Artificial Intelligence* **2017**, *31*, DOI: 10.1609/aaai.v31i1.10804.

(114) You, J.; Liu, B.; Ying, R.; Pande, V.; Leskovec, J. Graph Convolutional Policy Network for Goal-Directed Molecular Graph Generation, 2018, DOI: 10.48550/ARXIV.1806.02473.

(115) Voinarovska, V.; Kabeshov, M.; Dudenko, D.; Genheden, S.; Tetko, I. V. When Yield Prediction Does Not Yield Prediction: An Overview of the Current Challenges. *Journal of Chemical Information and Modeling* **2023**, *64*, 42–56, DOI: 10.1021/acs.jcim.3c01524.

(116) Brönsted, J. N.; Pedersen, K. Die katalytische Zersetzung des Nitramids und ihre physikalisch-chemische Bedeutung. *Zeitschrift für Physikalische Chemie* **1924**, *108U*, 185–235, DOI: 10.1515/zpch-1924-10814.

(117) Hammett, L. P. Some Relations between Reaction Rates and Equilibrium Constants. *Chem. Rev.* **1935**, *17*, 125–136, DOI: 10.1021/cr60056a010.

(118) Aoyama, T.; Ichikawa, H. Neural networks as nonlinear structure-activity relationship analyzers. Useful functions of the partial derivative method in multilayer neural networks. *J. Chem. Inf. Model.* **1992**, *32*, 492–500, DOI: 10.1021/ci00009a015.

(119) Zheng, W.; Tropsha, A. Novel Variable Selection Quantitative Structure-Property Relationship Approach Based on the ik/i-Nearest-Neighbor Principle. *J. Chem. Inf. Model.* **1999**, *40*, 185–194, DOI: 10.1021/ci980033m.

(120) Liu, H. X.; Zhang, R. S.; Yao, X. J.; Liu, M. C.; Hu, Z. D.; Fan, B. T. QSAR Study of Ethyl 2-[(3-Methyl-2, 5-dioxo(3-pyrrolinyl))amino]-4-(trifluoromethyl) pyrimidine-5-carboxylate: An Inhibitor of AP-1 and NF-κB Mediated Gene Expression Based on Support Vector Machines. *J. Chem. Inf. Model.* **2003**, *43*, 1288–1296, DOI: 10.1021/ci0340355.

(121) Svetnik, V.; Liaw, A.; Tong, C.; Culberson, J. C.; Sheridan, R. P.; Feuston, B. P. Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *J. Chem. Inf. Model.* **2003**, *43*, 1947–1958, DOI: 10.1021/ci034160g.

(122) Williams, W. L.; Zeng, L.; Gensch, T.; Sigman, M. S.; Doyle, A. G.; Anslyn, E. V. The Evolution of Data-Driven Modeling in Organic Chemistry. *ACS Cent. Sci.* **2021**, *7*, 1622–1637, DOI: 10.1021/acscentsci.1c00535.

(123) Emami, F. S.; Vahid, A.; Wylie, E. K.; Szymkuć, S.; Dittwald, P.; Molga, K.; Grzybowski, B. A. A Priori Estimation of Organic Reaction Yields. *Angew. Chem., Int. Ed.* **2015**, *54*, 10797–10801, DOI: 10.1002/anie.201503890.

(124) Raccuglia, P.; Elbert, K. C.; Adler, P. D. F.; Falk, C.; Wenny, M. B.; Mollo, A.; Zeller, M.; Friedler, S. A.; Schrier, J.; Norquist, A. J. Machine-learning-assisted materials discovery using failed experiments. *Nature* **2016**, *533*, 73–76, DOI: 10.1038/nature17439.

(125) Schwaller, P.; Vaucher, A. C.; Laino, T.; Reymond, J.-L. Prediction of chemical reaction yields using deep learning. *Mach. learn.: sci. technol.* **2021**, *2*, 015016, DOI: 10.1088/2632-2153/abc81d.

(126) Jiang, S.; Zhang, Z.; Zhao, H.; Li, J.; Yang, Y.; Lu, B.-L.; Xia, N. When SMILES Smiles, Practicality Judgment and Yield Prediction of Chemical Reaction via Deep Chemical Language Processing. *IEEE Access* **2021**, *9*, 85071–85083, DOI: 10.1109/access.2021.3083838.

(127) Ahneman, D. T.; Estrada, J. G.; Lin, S.; Dreher, S. D.; Doyle, A. G. Predicting reaction performance in C–N cross-coupling using machine learning. *Science* **2018**, *360*, 186–190, DOI: 10.1126/science.aar5169.

(128) Chuang, K. V.; Keiser, M. J. Comment on “Predicting reaction performance in C–N cross-coupling using machine learning”. *Science* **2018**, *362*, DOI: 10.1126/science.aat8603.

(129) Żurański, A. M.; Alvarado, J. I. M.; Shields, B. J.; Doyle, A. G. Predicting Reaction Yields via Supervised Learning. *Acc. Chem. Res.* **2021**, *54*, 1856–1865, DOI: 10.1021/acs.accounts.0c00770.

(130) Sandfort, F.; Strieth-Kalthoff, F.; Kühnemund, M.; Beecks, C.; Glorius, F. A Structure-Based Platform for Predicting Chemical Reactivity. *Chem* **2020**, *6*, 1379–1390, DOI: 10.1016/j.chempr.2020.02.017.

(131) Dong, J.; Peng, L.; Yang, X.; Zhang, Z.; Zhang, P. scpXGBoost-based/scp intelligence yield prediction and reaction factors analysis of amination reaction. *J. Comput. Chem.* **2021**, *43*, 289–302, DOI: 10.1002/jcc.26791.

(132) Johansson, S. V.; Svensson, H. G.; Bjerrum, E.; Schliep, A.; Chehreghani, M. H.; Tyrchan, C.; Engkvist, O. Using Active Learning to Develop Machine Learning Models for Reaction Yield Prediction. *Mol. Inf.* **2022**, *41*, 2200043, DOI: 10.1002/minf.202200043.

(133) Eyke, N. S.; Green, W. H.; Jensen, K. F. Iterative experimental design based on active machine learning reduces the experimental burden associated with reaction screening. *React. Chem. Eng.* **2020**, *5*, 1963–1972, DOI: 10.1039/d0re00232a.

(134) Chen, K.; Chen, G.; Li, J.; Huang, Y.; Wang, E.; Hou, T.; Heng, P.-A. MetaRF: attention-based random forest for reaction yield prediction with a few trails. *J. Cheminf.* **2023**, *15*, DOI: 10.1186/s13321-023-00715-x.

(135) Haywood, A. L.; Redshaw, J.; Hanson-Heine, M. W. D.; Taylor, A.; Brown, A.; Mason, A. M.; Gärtner, T.; Hirst, J. D. Kernel Methods for Predicting Yields of Chemical Reactions. *J. Chem. Inf. Model.* **2021**, *62*, 2077–2092, DOI: 10.1021/acs.jcim.1c00699.

(136) Ranković, B.; Griffiths, R.-R.; Moss, H. B.; Schwaller, P. Bayesian optimisation for additive screening and yield improvements in chemical reactions – beyond one-hot encoding. *ChemRxiv* **2023**, DOI: 10.26434/chemrxiv-2022-nll2j-v3.

(137) Fitzner, M.; Wuitschik, G.; Koller, R.; Adam, J.-M.; Schindler, T. Machine Learning C–N Couplings: Obstacles for a General-Purpose Reaction Yield Prediction. *ACS Omega* **2023**, *8*, 3017–3025, DOI: 10.1021/acsomega.2c05546.

(138) Reker, D.; Hoyt, E. A.; Bernardes, G. J.; Rodrigues, T. Adaptive Optimization of Chemical Reactions with Minimal Experimental Information. *Cell Rep. Phys. Sci.* **2020**, *1*, 100247, DOI: 10.1016/j.xcrp.2020.100247.

(139) Chithrananda, S.; Grand, G.; Ramsundar, B. ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction, 2020, DOI: 10.48550/ARXIV.2010.09885.

(140) Honda, S.; Shi, S.; Ueda, H. R. SMILES Transformer: Pre-trained Molecular Fingerprint for Low Data Drug Discovery, 2019, DOI: 10.48550/ARXIV.1911.04738.

(141) Baraka, S.; Kerdawy, A. M. E. Multimodal Transformer-based Model for Buchwald-Hartwig and Suzuki-Miyaura Reaction Yield Prediction, 2022, DOI: 10.48550/ARXIV.2204.14062.

(142) Sagawa, T.; Kojima, R. ReactionT5: a large-scale pre-trained model towards application of limited reaction data, 2023, DOI: 10.48550/ARXIV.2311.06708.

(143) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular graph convolutions: moving beyond fingerprints. *Journal of Computer-Aided Molecular Design* **2016**, *30*, 595–608, DOI: 10.1007/s10822-016-9938-8.

(144) Sato, A.; Miyao, T.; Funatsu, K. Prediction of Reaction Yield for Buchwald-Hartwig Cross-coupling Reactions Using Deep Learning. *Mol. Inf.* **2021**, *41*, 2100156, DOI: 10.1002/minf.202100156.

(145) Jaeger, S.; Fulle, S.; Turk, S. Mol2vec: Unsupervised Machine Learning Approach with Chemical Intuition. *J. Chem. Inf. Model.* **2018**, *58*, 27–35, DOI: 10.1021/acs.jcim.7b00616.

(146) Kwon, Y.; Lee, D.; Choi, Y.-S.; Kang, S. Uncertainty-aware prediction of chemical reaction yields with graph neural networks. *J. Cheminf.* **2022**, *14*, DOI: 10.1186/s13321-021-00579-z.

(147) Saebi, M.; Nan, B.; Herr, J. E.; Wahlers, J.; Guo, Z.; Zurański, A. M.; Kogej, T.; Norrby, P.-O.; Doyle, A. G.; Chawla, N. V.; Wiest, O. On the use of real-world datasets for reaction yield prediction. *Chem. Sci.* **2023**, *14*, 4997–5005, DOI: 10.1039/d2sc06041h.

(148) Neves, P.; McClure, K.; Verhoeven, J.; Dyubankova, N.; Nugmanov, R.; Gedich, A.; Menon, S.; Shi, Z.; Wegner, J. K. Global reactivity models are impactful in industrial synthesis applications. *J. Cheminf.* **2023**, *15*, DOI: 10.1186/s13321-023-00685-0.

(149) Yarish, D.; Garkot, S.; Grygorenko, O. O.; Radchenko, D. S.; Moroz, Y. S.; Gurbych, O. Advancing molecular graphs with descriptors for the prediction of chemical reaction yields. *J. Comput. Chem.* **2022**, *44*, 76–92, DOI: 10.1002/jcc.27016.

(150) Ley, S. V.; Fitzpatrick, D. E.; Ingham, R. J.; Myers, R. M. Organic Synthesis: March of the Machines. *Angew. Chem., Int. Ed.* **2015**, *54*, 3449–3464, DOI: 10.1002/anie.201410744.

(151) Schneider, G. Automating drug discovery. *Nature Reviews Drug Discovery* **2017**, *17*, 97–113, DOI: 10.1038/nrd.2017.232.

(152) Perera, D.; Tucker, J. W.; Brahmbhatt, S.; Helal, C. J.; Chong, A.; Farrell, W.; Richardson, P.; Sach, N. W. A platform for automated nanomole-scale reaction screening and micromole-scale synthesis in flow. *Science* **2018**, *359*, 429–434, DOI: 10.1126/science.aap9112.

(153) Li, J.; Ballmer, S. G.; Gillis, E. P.; Fujii, S.; Schmidt, M. J.; Palazzolo, A. M. E.; Lehmann, J. W.; Morehouse, G. F.; Burke, M. D. Synthesis of many different types of organic small molecules using one automated process. *Science* **2015**, *347*, 1221–1226, DOI: 10.1126/science.aaa5414.

(154) Santanilla, A. B.; Regalado, E. L.; Pereira, T.; Shevlin, M.; Bateman, K.; Campeau, L.-C.; Schneeweis, J.; Berritt, S.; Shi, Z.-C.; Nantermet, P.; Liu, Y.; Helmy, R.; Welch, C. J.; Vachal, P.; Davies, I. W.; Cernak, T.; Dreher, S. D. Nanomole-scale high-throughput chemistry for the synthesis of complex molecules. *Science* **2015**, *347*, 49–53, DOI: 10.1126/science.1259203.

(155) Ahn, G.-N.; Sharma, B. M.; Lahore, S.; Yim, S.-J.; Vidyacharan, S.; Kim, D.-P. Flow parallel synthesizer for multiplex synthesis of aryl diazonium libraries via efficient parameter screening. *Commun. Chem.* **2021**, *4*, DOI: 10.1038/s42004-021-00490-6.

(156) Christensen, M.; Yunker, L. P. E.; Adedeji, F.; Häse, F.; Roch, L. M.; Gensch, T.; dos Passos Gomes, G.; Zepel, T.; Sigman, M. S.; Aspuru-Guzik, A.; Hein, J. E. Data-science driven autonomous process optimization. *Commun. Chem.* **2021**, *4*, DOI: 10.1038/s42004-021-00550-x.

(157) Gabrielson, S. W. SciFinder. *Journal of the Medical Library Association* **2018**, *106*, DOI: 10.5195/jmla.2018.515.

(158) NextMove, https://nextmovesoftware.com/, accessed on 03.07.2022.

(159) Kearnes, S. M.; Maser, M. R.; Wleklinski, M.; Kast, A.; Doyle, A. G.; Dreher, S. D.; Hawkins, J. M.; Jensen, K. F.; Coley, C. W. The Open Reaction Database. *J. Am. Chem. Soc.* **2021**, *143*, 18820–18826, DOI: 10.1021/jacs.1c09820.

(160) King-Smith, E.; Berritt, S.; Bernier, L.; Hou, X.; Klug-McLeod, J. L.; Mustakis, J.; Sach, N. W.; Tucker, J. W.; Yang, Q.; Howard, R. M.; Lee, A. A. Probing the chemical 'reactome' with high-throughput experimentation data. *Nature Chemistry* **2024**, *16*, 633–643, DOI: 10.1038/s41557-023-01393-w.

(161) Eyke, N. S.; Koscher, B. A.; Jensen, K. F. Toward Machine Learning-Enhanced High-Throughput Experimentation. *Trends Chem.* **2021**, *3*, 120–132, DOI: 10.1016/j.trechm.2020.12.001.

(162) Huffman, M. A.; Fryszkowska, A.; Alvizo, O.; Borra-Garske, M.; Campos, K. R.; Canada, K. A.; Devine, P. N.; Duan, D.; Forstater, J. H.; Grosser, S. T.; Halsey, H. M.; Hughes, G. J.; Jo, J.; Joyce, L. A.; Kolev, J. N.; Liang, J.; Maloney, K. M.; Mann, B. F.; Marshall, N. M.; McLaughlin, M., et al. Design of an in vitro biocatalytic cascade for the manufacture of islatravir. *Science* **2019**, *366*, 1255–1259, DOI: 10.1126/science.aay8484.

(163) Liu, R. Copper-Catalyzed Enantioselective Hydroamination of Alkenes. *Org. Synth.* **2018**, *95*, 80–96, DOI: 10.15227/orgsyn.095.0080.

(164) Christensen, M.; Adedeji, F.; Grosser, S.; Zawatzky, K.; Ji, Y.; Liu, J.; Jurica, J. A.; Naber, J. R.; Hein, J. E. Development of an automated kinetic profiling system with online HPLC for reaction optimization. *React. Chem. Eng.* **2019**, *4*, 1555–1558, DOI: 10.1039/c9re00086k.

(165) Zuo, Z.; Ahneman, D. T.; Chu, L.; Terrett, J. A.; Doyle, A. G.; MacMillan, D. W. C. Merging photoredox with nickel catalysis: Coupling of α-carboxyl sp sup3/sup -carbons with aryl halides. *Science* **2014**, *345*, 437–440, DOI: 10.1126/science.1255525.

(166) Gioiello, A.; Rosatelli, E.; Teofrasti, M.; Filipponi, P.; Pellicciari, R. Building a Sulfonamide Library by Eco-Friendly Flow Synthesis. *ACS Comb. Sci.* **2013**, *15*, 235–239, DOI: 10.1021/co400012m.

(167) Stadler, A.; Kappe, C. O. Automated Library Generation Using Sequential Microwave-Assisted Chemistry. Application toward the Biginelli Multicomponent Condensation. *J. Comb. Chem.* **2001**, *3*, 624–630, DOI: 10.1021/cc010044j.

(168) Nielsen, M. K.; Ahneman, D. T.; Riera, O.; Doyle, A. G. Deoxyfluorination with Sulfonyl Fluorides: Navigating Reaction Space with Machine Learning. *J. Am. Chem. Soc.* **2018**, *140*, 5004–5008, DOI: 10.1021/jacs.8b01523.

(169) Kutchukian, P. S.; Dropinski, J. F.; Dykstra, K. D.; Li, B.; DiRocco, D. A.; Streckfuss, E. C.; Campeau, L.-C.; Cernak, T.; Vachal, P.; Davies, I. W.; Krska, S. W.; Dreher, S. D. Chemistry informer libraries: a chemoinformatics enabled approach to evaluate and advance synthetic methods. *Chem. Sci.* **2016**, *7*, 2604–2613, DOI: 10.1039/c5sc04751j.

(170) Schwärzer, K.; Rout, S. K.; Bessinger, D.; Lima, F.; Brocklehurst, C. E.; Karaghiosoff, K.; Bein, T.; Knochel, P. Selective functionalization of the 1iH/i-imidazo[1, 2-ib/i]pyrazole scaffold. A new potential non-classical isostere of indole and a precursor of push–pull dyes. *Chem. Sci.* **2021**, *12*, 12993–13000, DOI: 10.1039/d1sc04155j.

(171) Newman-Stonebraker, S.; Smith, S.; Borowski, J.; Peters, E.; Gensch, T.; Johnson, H.; Sigman, M.; Doyle, A. Linking Mechanistic Analysis of Catalytic Reactivity Cliffs to Ligand Classification. *ChemRxiv* **2021**, DOI: 10.26434/chemrxiv.14388557.v1.

(172) Mdluli, V.; Diluzio, S.; Lewis, J.; Kowalewski, J. F.; Connell, T. U.; Yaron, D.; Kowalewski, T.; Bernhard, S. High-throughput Synthesis and Screening of Iridium(III) Photocatalysts for the Fast and Chemoselective Dehalogenation of Aryl Bromides. *ACS Catal.* **2020**, *10*, 6977–6987, DOI: 10.1021/acscatal.0c02247.

(173) Dreher, S. D.; Krska, S. W. Chemistry Informer Libraries: Conception, Early Experience, and Role in the Future of Cheminformatics. *Acc. Chem. Res.* **2021**, *54*, 1586–1596, DOI: 10.1021/acs.accounts.0c00760.

(174) Schleinitz, J.; Langevin, M.; Smail, Y.; Wehnert, B.; Grimaud, L.; Vuilleumier, R. Machine Learning Yield Prediction from NiCOlit, a Small-Size Literature Data Set of Nickel Catalyzed C–O Couplings. *J. Am. Chem. Soc.* **2022**, *144*, 14722–14730, DOI: 10.1021/jacs.2c05302.

(175) Murray, P. M.; Tyler, S. N. G.; Moseley, J. D. Beyond the Numbers: Charting Chemical Reaction Space. *Org. Process Res. Dev.* **2013**, *17*, 40–46, DOI: 10.1021/op300275p.

(176) Fitzner, M.; Wuitschik, G.; Koller, R. J.; Adam, J.-M.; Schindler, T.; Reymond, J.-L. What can reaction databases teach us about Buchwald–Hartwig cross-couplings? *Chem. Sci.* **2020**, *11*, 13085–13093, DOI: 10.1039/d0sc04074f.

(177) Strieth-Kalthoff, F.; Sandfort, F.; Kühnemund, M.; Schäfer, F. R.; Kuchen, H.; Glorius, F. Machine Learning for Chemical Reactivity: The Importance of Failed Experiments. *Angew. Chem., Int. Ed.* **2022**, *61*, DOI: 10.1002/anie.202204647.

(178) Maloney, M. P.; Coley, C. W.; Genheden, S.; Carson, N.; Helquist, P.; Norrby, P.-O.; Wiest, O. Negative Data in Data Sets for Machine Learning Training. *Org. Lett.* **2023**, *25*, 2945–2947, DOI: 10.1021/acs.orglett.3c01282.

(179) Jablonka, K. M.; Patiny, L.; Smit, B. Making the collective knowledge of chemistry open and machine actionable. *Nat. Chem.* **2022**, *14*, 365–376, DOI: 10.1038/s41557-022-00910-7.

(180) Mehr, S. H. M.; Craven, M.; Leonov, A. I.; Keenan, G.; Cronin, L. A universal system for digitization and automatic execution of the chemical synthesis literature. *Science* **2020**, *370*, 101–108, DOI: 10.1126/science.abc2986.

(181) Qian, Y.; Guo, J.; Tu, Z.; Coley, C. W.; Barzilay, R. RxnScribe: A Sequence Generation Model for Reaction Diagram Parsing, 2023, DOI: 10.48550/ARXIV.2305.11845.

(182) Wilary, D. M.; Cole, J. M. ReactionDataExtractor: A Tool for Automated Extraction of Information from Chemical Reaction Schemes. *J. Chem. Inf. Model.* **2021**, *61*, 4962–4974, DOI: 10.1021/acs.jcim.1c01017.

(183) Fourches, D.; Muratov, E.; Tropsha, A. Trust, But Verify: On the Importance of Chemical Structure Curation in Cheminformatics and QSAR Modeling Research. *Journal of Chemical Information and Modeling* **2010**, *50*, 1189–1204, DOI: 10.1021/ci100176x.

(184) Fourches, D.; Muratov, E.; Tropsha, A. Trust, but Verify II: A Practical Guide to Chemogenomics Data Curation. *Journal of Chemical Information and Modeling* **2016**, *56*, 1243–1252, DOI: 10.1021/acs.jcim.6b00129.

(185) Gimadiev, T. R.; Lin, A.; Afonina, V. A.; Batyrshin, D.; Nugmanov, R. I.; Akhmetshin, T.; Sidorov, P.; Duybankova, N.; Verhoeven, J.; Wegner, J.; Ceulemans, H.; Gedich, A.; Madzhidov, T. I.; Varnek, A. Reaction Data Curation I: Chemical Structures and Transformations Standardization. *Molecular Informatics* **2021**, *40*, DOI: 10.1002/minf.202100119.

(186) Kannas, C.; Thakkar, A.; Bjerrum, E.; Genheden, S. rxnutils – A Cheminformatics Python Library for Manipulating Chemical Reaction Data. **2022**, DOI: 10.26434/chemrxiv-2022-wt440-v2.

(187) Raghavan, P.; Haas, B. C.; Ruos, M. E.; Schleinitz, J.; Doyle, A. G.; Reisman, S. E.; Sigman, M. S.; Coley, C. W. Dataset Design for Building Models of Chemical Reactivity. *ACS Central Science* **2023**, *9*, 2196–2204, DOI: 10.1021/acscentsci.3c01163.

(188) Wigh, D. S.; Arrowsmith, J.; Pomberger, A.; Felton, K. C.; Lapkin, A. A. ORDerly: Data Sets and Benchmarks for Chemical Reaction Data. *Journal of Chemical Information and Modeling* **2024**, *64*, 3790–3798, DOI: 10.1021/acs.jcim.4c00292.

(189) Wigh, D. S.; Goodman, J. M.; Lapkin, A. A. A review of molecular representation in the age of machine learning. *WIREs Computational Molecular Science* **2022**, *12*, DOI: 10.1002/wcms.1603.

(190) SMARTS, https://www.daylight.com/dayhtml/doc/theory/theory.smarts.html, accessed on 02.08.2022.

(191) Krenn, M.; Häse, F.; Nigam, A.; Friederich, P.; Aspuru-Guzik, A. Self-referencing embedded strings (SELFIES): A 100% robust molecular string representation. *Mach. learn.: sci. technol.* **2020**, *1*, 045024, DOI: 10.1088/2632-2153/aba947.

(192) Krenn, M.; Ai, Q.; Barthel, S.; Carson, N.; Frei, A.; Frey, N. C.; Friederich, P.; Gaudin, T.; Gayle, A. A.; Jablonka, K. M.; Lameiro, R. F.; Lemm, D.; Lo, A.; Moosavi, S. M.; Nápoles-Duarte, J. M.; Nigam, A.; Pollice, R.; Rajan, K.; Schatzschneider, U.; Schwaller, P., et al. SELFIES and the future of molecular string representations. *Patterns* **2022**, *3*, 100588, DOI: 10.1016/j.patter.2022.100588.

(193) O'Boyle, N.; Dalke, A. DeepSMILES: An Adaptation of SMILES for Use in Machine-Learning of Chemical Structures. **2018**, DOI: 10.26434/chemrxiv.7097960.v1.

(194) RDKit, https://www.rdkit.org, accessed on 13.07.2022.

(195) Quirós, M.; Gražulis, S.; Girdzijauskaitė, S.; Merkys, A.; Vaitkus, A. Using SMILES strings for the description of chemical connectivity in the Crystallography Open Database. *J. Cheminf.* **2018**, *10*, DOI: 10.1186/s13321-018-0279-6.

(196) Behler, J.; Parrinello, M. Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces. *Physical Review Letters* **2007**, *98*, DOI: 10.1103/physrevlett.98.146401.

(197) Schütt, K. T.; Gastegger, M.; Tkatchenko, A.; Müller, K.-R.; Maurer, R. J. Unifying machine learning and quantum chemistry with a deep neural network for molecular wavefunctions. *Nature Communications* **2019**, *10*, DOI: 10.1038/s41467-019-12875-2.

(198) Schütt, K. T.; Kindermans, P.-J.; Sauceda, H. E.; Chmiela, S.; Tkatchenko, A.; Müller, K.-R. SchNet: A continuous-filter convolutional neural network for modeling quantum interactions. **2017**, DOI: 10.48550/ARXIV.1706.08566.

(199) Morgan, H. L. The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service. *J. Chem. Doc.* **1965**, *5*, 107–113, DOI: 10.1021/c160017a018.

(200) Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints. *J. Chem. Inf. Model.* **2010**, *50*, 742–754, DOI: 10.1021/ci100050t.

(201) Probst, D.; Schwaller, P.; Reymond, J.-L. Reaction classification and yield prediction using the differential reaction fingerprint DRFP. *Digital Discovery* **2022**, *1*, 91–97, DOI: 10.1039/d1dd00006c.

(202) Probst, D.; Reymond, J.-L. A probabilistic molecular fingerprint for big data settings. *Journal of Cheminformatics* **2018**, *10*, DOI: 10.1186/s13321-018-0321-8.

(203) Caldeweyher, E. kallisto: A command-line interface to simplify computational modelling and the generation of atomic features. *J. Open Source Softw.* **2021**, *6*, 3050, DOI: 10.21105/joss.03050.

(204) Varnek, A.; Fourches, D.; Hoonakker, F.; Solov'ev, V. P. Substructural fragments: an universal language to encode reactions, molecular and supramolecular structures. *J. Comput.-Aided Mol. Des.* **2005**, *19*, 693–703, DOI: 10.1007/s10822-005-9008-0.

(205) Fujita, S. Description of organic reactions based on imaginary transition structures. 1. Introduction of new concepts. *J. Chem. Inf. Model.* **1986**, *26*, 205–212, DOI: 10.1021/ci00052a009.

(206) Nugmanov, R. I.; Mukhametgaleev, R. N.; Akhmetshin, T.; Gimadiev, T. R.; Afonina, V. A.; Madzhidov, T. I.; Varnek, A. CGRtools: Python Library for Molecule, Reaction, and Condensed Graph of Reaction Processing. *J. Chem. Inf. Model.* **2019**, *59*, 2516–2521, DOI: 10.1021/acs.jcim.9b00102.

(207) Heid, E.; Greenman, K. P.; Chung, Y.; Li, S.-C.; Graff, D. E.; Vermeire, F. H.; Wu, H.; Green, W. H.; McGill, C. J. Chemprop: A Machine Learning Package for Chemical Property Prediction. *Journal of Chemical Information and Modeling* **2023**, *64*, 9–17, DOI: 10.1021/acs.jcim.3c01250.

(208) Winter, R.; Montanari, F.; Noé, F.; Clevert, D.-A. Learning continuous and data-driven molecular descriptors by translating equivalent chemical representations. *Chem. Sci.* **2019**, *10*, 1692–1701, DOI: 10.1039/c8sc04175j.

(209) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; Duchesnay, E. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

(210) Lemaître, G.; Nogueira, F.; Aridas, C. K. Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *Journal of Machine Learning Research* **2017**, *18*, 1–5.

(211) Breiman, L. *Machine Learning* **2001**, *45*, 5–32, DOI: 10.1023/a:1010933404324.

(212) Ho, T. K. In *Proceedings of 3rd international conference on document analysis and recognition*, 1995; Vol. 1, pp 278–282.

(213) Louppe, G. Understanding Random Forests: From Theory to Practice, 2014, DOI: 10.48550/ARXIV.1407.7502.

(214) Friedman, J. H. Greedy function approximation: a gradient boosting machine. *Ann. Stat.* **2001**, 1189–1232.

(215) Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297.

(216) Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

(217) Hunter, J. D. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering* **2007**, *9*, 90–95, DOI: 10.1109/MCSE.2007.55.

(218) Waskom, M. L. seaborn: statistical data visualization. *Journal of Open Source Software* **2021**, *6*, 3021, DOI: 10.21105/joss.03021.

(219) Van der Maaten, L.; Hinton, G. Visualizing Data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.

(220) Chawla, N. V.; Bowyer, K. W.; Hall, L. O.; Kegelmeyer, W. P. SMOTE: Synthetic Minority Oversampling Technique. *Journal of Artificial Intelligence Research* **2002**, *16*, 321–357, DOI: 10.1613/jair.953.

(221) Indigo, https://lifescience.opensource.epam.com/indigo/, accessed on 03.07.2024.

(222) Jaworski, W.; Szymkuć, S.; Mikulak-Klucznik, B.; Piecuch, K.; Klucznik, T.; Kaźmierowski, M.; Rydzewski, J.; Gambin, A.; Grzybowski, B. A. Automatic mapping of atoms across both simple and complex chemical reactions. *Nature Communications* **2019**, *10*, DOI: 10.1038/s41467-019-09440-2.

(223) Estrada, J. G.; Ahneman, D. T.; Sheridan, R. P.; Dreher, S. D.; Doyle, A. G. Response to Comment on “Predicting reaction performance in C–N cross-coupling using machine learning”. *Science* **2018**, *362*, DOI: 10.1126/science.aat8763.

(224) Isbrandt, E.; Sullivan, R.; Newman, S. High Throughput Strategies for the Discovery and Optimization of Catalytic Reactions. *Angewandte Chemie International Edition* **2019**, *58*, DOI: 10.1002/anie.201812534.

(225) Vogel, A. I.; Tatchell, A. R.; Furnis, B. S.; Hannaford, A. J.; Greig- Smith, P. W., *Vogel's textbook of practical organic chemistry*, 5th ed.; Prentice-Hall: London, England, 1989.

(226) M. Bran, A.; Cox, S.; Schilter, O.; Baldassari, C.; White, A. D.; Schwaller, P. Augmenting large language models with chemistry tools. *Nature Machine Intelligence* **2024**, *6*, 525–535, DOI: 10.1038/s42256-024-00832-8.

(227) Mirza, A.; Alampara, N.; Kunchapu, S.; Emoekabu, B.; Krishnan, A.; Wilhelmi, M.; Okereke, M.; Eberhardt, J.; Elahi, A. M.; Greiner, M.; Holick, C. T.; Gupta, T.; Asgari, M.; Glaubitz, C.; Klepsch, L. C.; Köster, Y.; Meyer, J.; Miret, S.; Hoffmann, T.; Kreth, F. A., et al. Are large language models superhuman chemists?, 2024, DOI: 10.48550/ARXIV.2404.01475.

(228) Ramos, M. C.; Collison, C. J.; White, A. D. A Review of Large Language Models and Autonomous Agents in Chemistry, 2024, DOI: 10.48550/ARXIV.2407.01603.

(229) Ai, Q.; Meng, F.; Shi, J.; Pelkie, B.; Coley, C. W. Extracting Structured Data from Organic Synthesis Procedures Using a Fine-Tuned Large Language Model. **2024**, DOI: 10.26434/chemrxiv-2024-979fz.

(230) Leong, S. X.; Pablo-García, S.; Zhang, Z.; Aspuru-Guzik, A. Automated electrosynthesis reaction mining with multimodal large language models (MLLMs). **2024**, DOI: 10.26434/chemrxiv-2024-7fwxv.

(231) Polak, M. P.; Morgan, D. Extracting accurate materials data from research papers with conversational language models and prompt engineering. *Nature Communications* **2024**, *15*, DOI: 10.1038/s41467-024-45914-8.

(232) Zhang, Y.; Yu, R.; Zeng, K.; Li, D.; Zhu, F.; Yang, X.; Jin, Y.; Xu, Y. Text-Augmented Multimodal LLMs for Chemical Reaction Condition Recommendation, 2024, DOI: 10.48550/ARXIV.2407.15141.

A Appendix

Parameters of the Random Forest models used.

```
model = RandomForestClassifier(
        n_estimators=600,
        max_depth=None,
        max_features="sqrt",
        min_samples_leaf=2,
        min_samples_split=5,
        random_state=42,
        bootstrap=False,
        criterion='gini',
        class_weight="balanced",
    )
```

```
model = RandomForestRegressor(
        n_estimators=600,
        max_depth=None,
        max_features='sqrt',
        min_samples_leaf=2,
        min_samples_split=5,
        bootstrap=True,
        random_state=42,
    )
```

Procedure for extracting reactions that have been purified:

I designed dictionaries for a) dropping the unsuccessful reaction or reactions that proceeded with the crude product to the next stage of the reaction and b) grasping reactions that had at least some information on whether the reaction was well-documented and had information on the reaction product. The final stage of determination aimed to determine the type of purification used for the reaction and make the previous selection more precise.

Dropping the unsuccessful reactions had the next keywords:

"abandon", "no product", "not progressed", "only starting material", "discontinued", "discarded", "no reaction", "no evidence", "no indication", "not continued", "not obtained", "product was not observed", "reaction failed", "reaction trashed", "no conversion", "not isolated", "no target", "not detected".

Dropping the reactions where the crude was used in the next stage without the purification had the next keywords:

"duplicated", "use as is", "used without purification", "used in the next", "crude was used", "used directly", "without further purification", "next step", "did not set up", "didn’t set up", "used as a crude".

Selection of reaction with some information on the product:

"purif", "isolate", "to give", "obtain", "yielding", "reaction was complete", "afford", "collect", "to yield", "gave", "provid", "giving", "given", "concentrat", "evaporat", "crystal", "dried". Selection of reactions that provided more defined information on the purification procedure: "chromatograph", "HPLC", "biotage", "ISCO", "triturat", "column", "silica", "gilson", "normal phase", "preparative LCMS", "Prep-LCMS", "preparative TLC".

			Test datasets RMSE			
Trained Model	Fingerprints	Training dataset	USPTO (test)	Reaxys (test)	BH HTE	AZ ELN
Random Forest Regression	RXNFP	USPTO	23.99	25.88	32.37	33.35
		Reaxys	28.54	20.70	40.29	40.26
	ECFP	USPTO	22.22	24.13	33.90	32.97
		Reaxys	26.72	18.25	43.30	39.15
	DRFP	USPTO	22.84	25.39	30.86	34.39
		Reaxys	28.31	18.70	42.26	40.23
Gradient Boost Regression	RXNFP	USPTO	24.29	26.17	31.86	33.63
		Reaxys	28.82	20.85	41.41	41.05
	ECFP	USPTO	22.39	24.18	34.75	32.03
		Reaxys	27.12	18.72	50.6	38.19
	DRFP	USPTO	23.41	25.93	32.47	33.28
		Reaxys	28.08	18.95	38.57	39.12
Support Vector Regression	RXNFP	USPTO	24.71	26.76	32.68	33.78
		Reaxys	30.53	21.72	42.27	43.82
	ECFP	USPTO	22.94	24.01	35.04	32.42
		Reaxys	27.60	18.9	48.29	40.76
	DRFP	USPTO	23.27	26.17	36.64	32.42
		Reaxys	28.98	19.56	48.75	41.37
BERT-Yield		USPTO	24.82	38.41	42.13	45.42
		Reaxys	38.45	20.99	47.18	48.61

Table A.1 RMSE for the models employed in the research

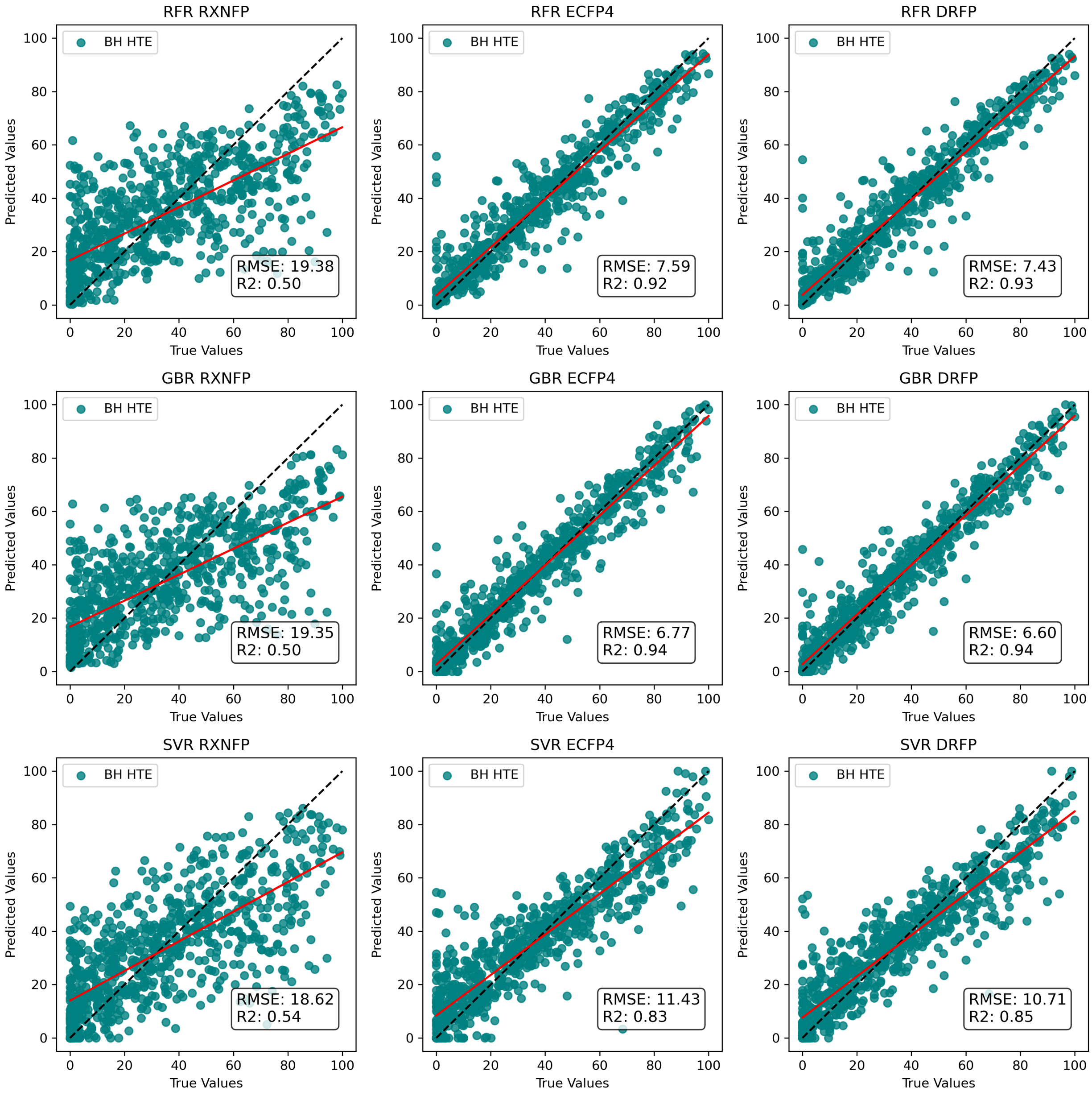


Figure A.1 Comparison of the RFR, GBR, SVR model's performance using different encodings and fingerprints, trained with a random 80:20 ratio and 5-fold Cross-Validation.

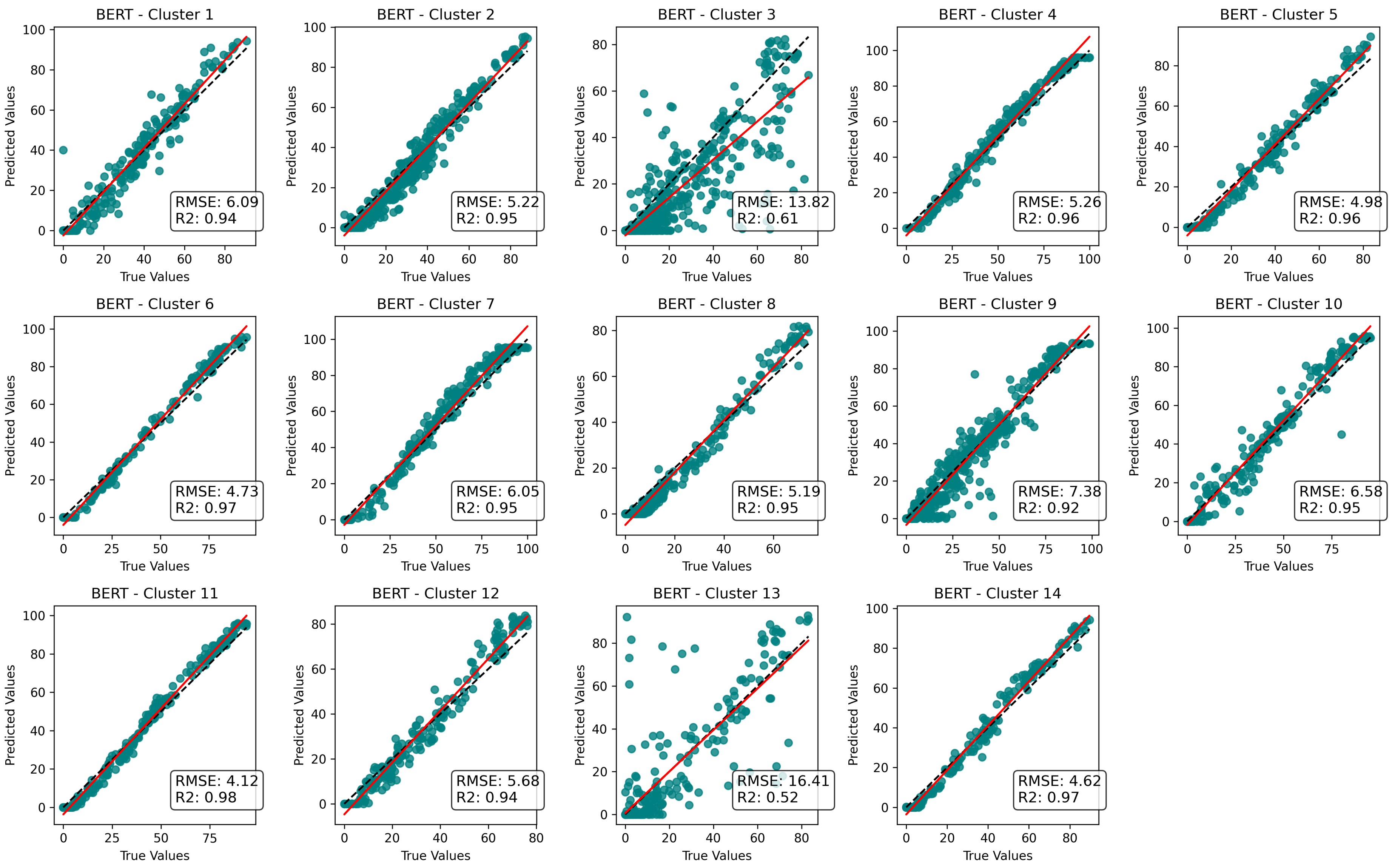


Figure A.2 Yield-BERT performance on training one DRFP cluster left out

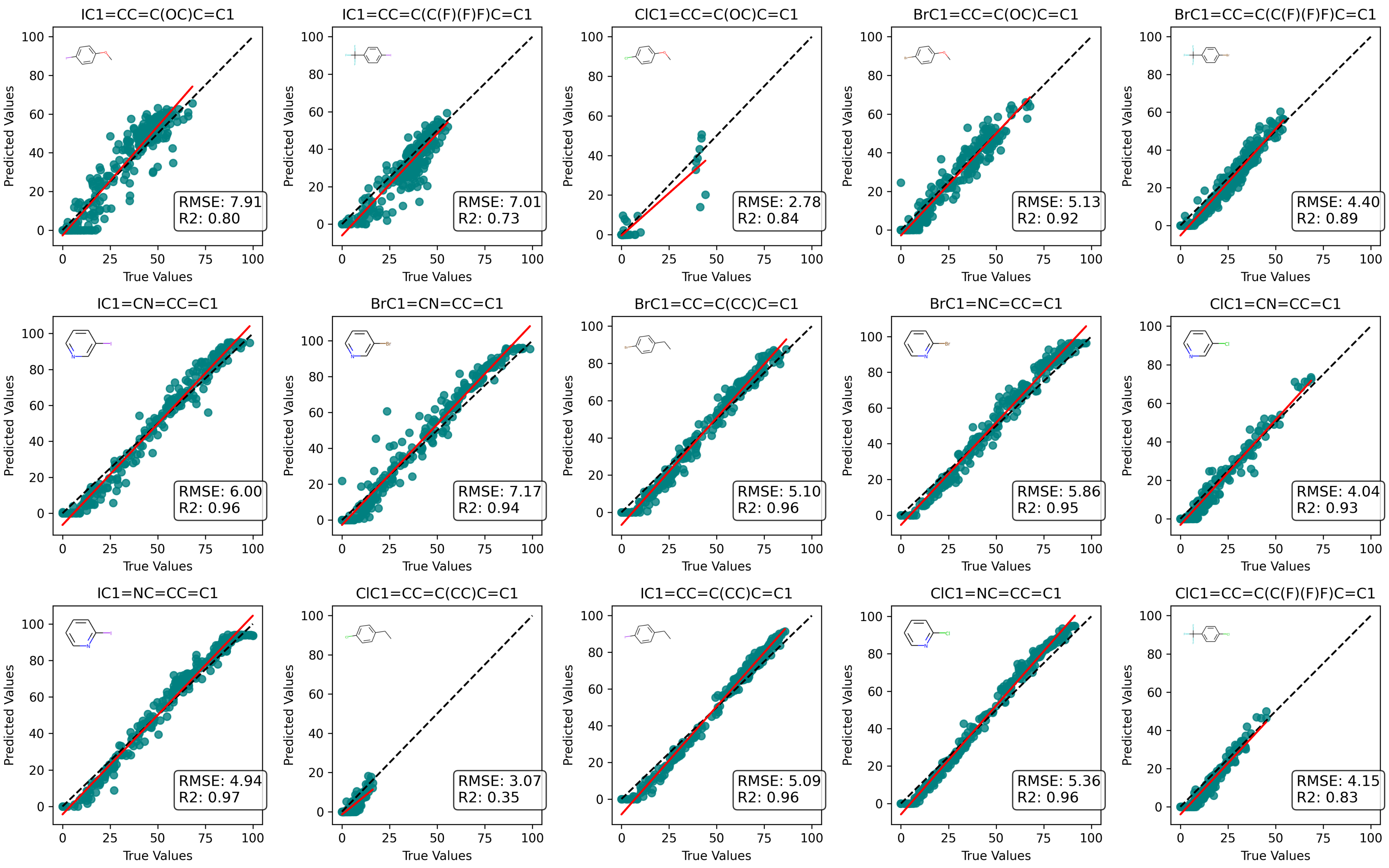


Figure A.3 Yield-BERT performance on training one aryl halide cluster left out

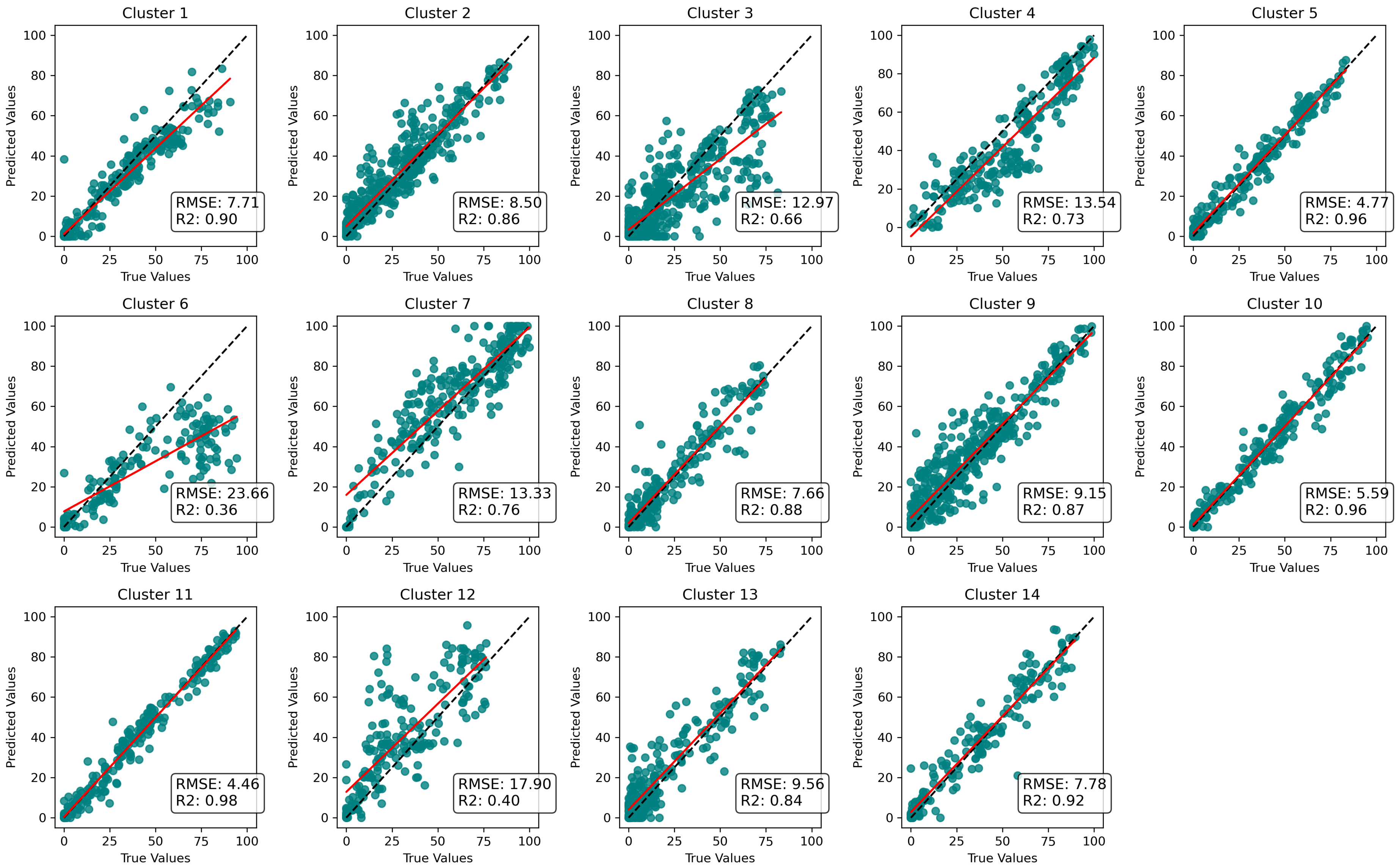


Figure A.4 GBR performance on BH HTE DRFP features training one DRFP cluster left out

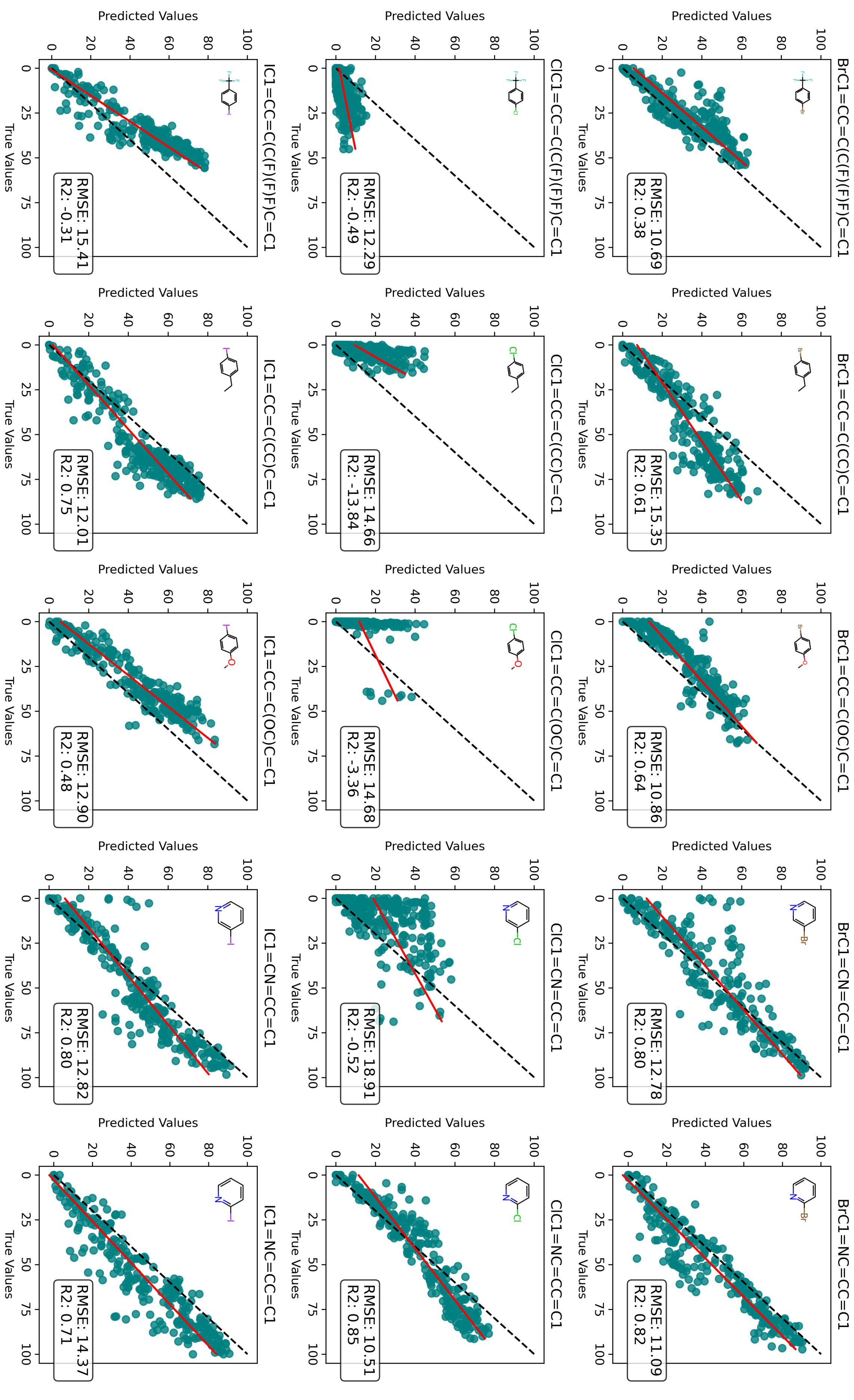


Figure A.5 The performance of GBR on BH HTE DRFP features was evaluated by training the model with one aryl halide left out as a test set. Each plot represents the reactions of the left-out aryl halide used for testing.

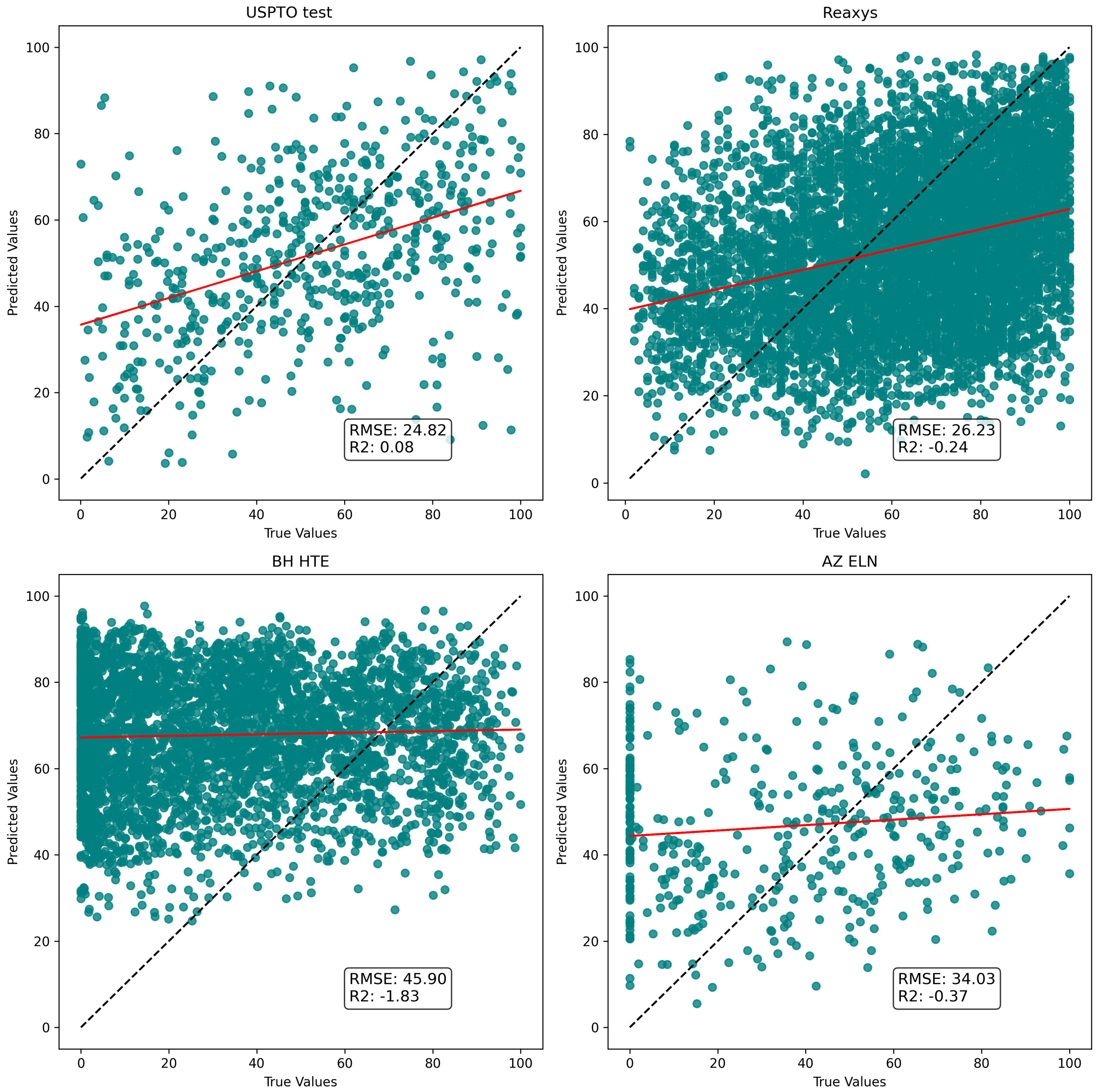
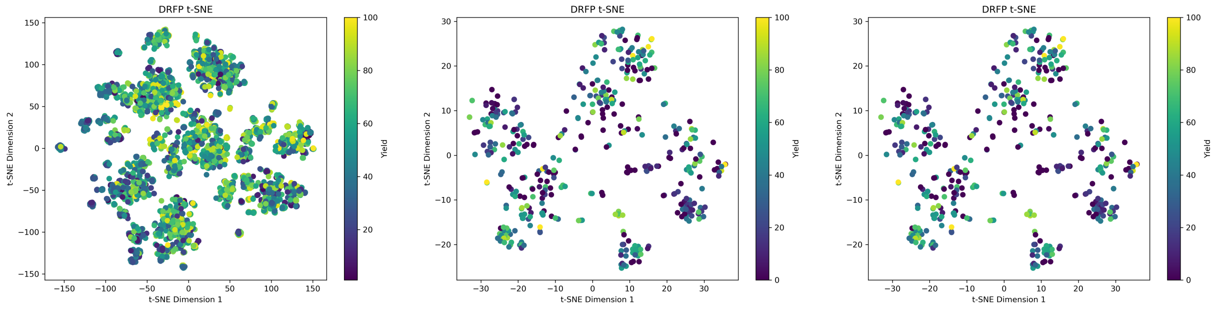


Figure A.6 Yield-BERT model trained on USPTO Buchwald-Hartwig selection and tested on other datasets. The red line represents linear fit.

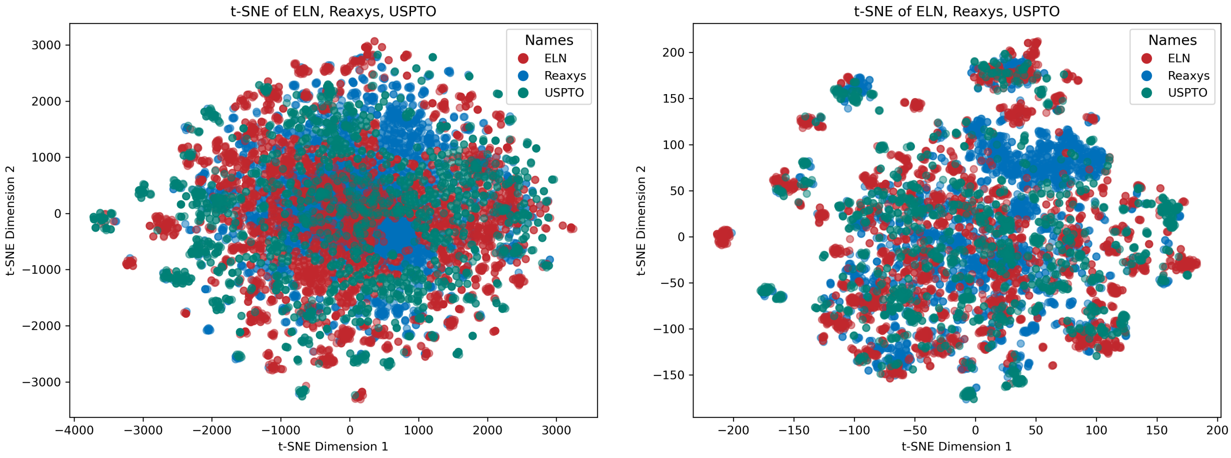


(a) Reaxys

(b) AZ ELN 750

(c) USPTO

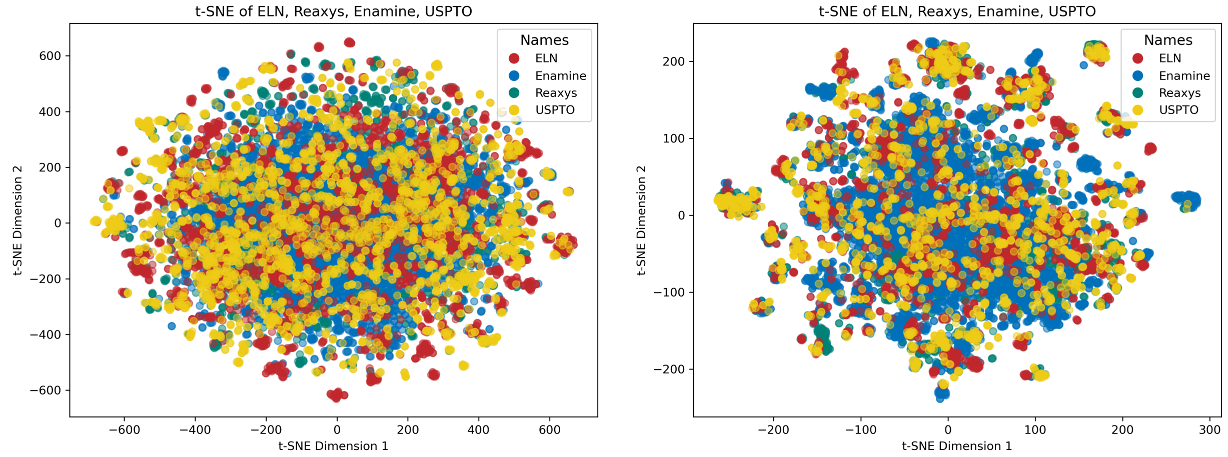
Figure A.7 t-SNE for other Buchwald-Hartwig datasets using DRFP fingerprint from different sources.



(a) ECFP-based t-SNE

(b) Kallisto-based t-SNE

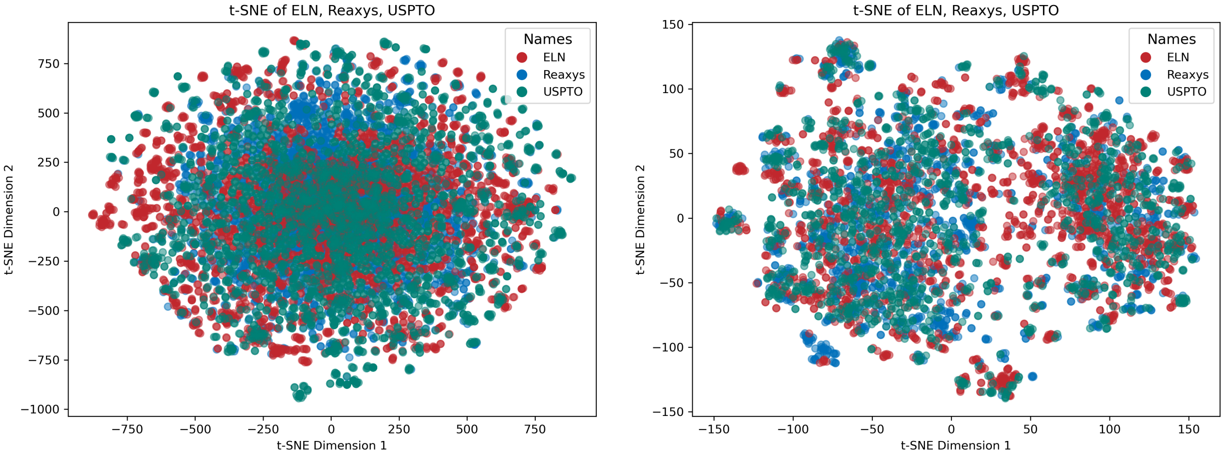
Figure A.8 The t-SNE plot depicts the distribution of reaction encodings based on Kallisto and ECFP representations for the Buchwald-Hartwig reaction.



(a) ECFP-based t-SNE

(b) Kallisto-based t-SNE

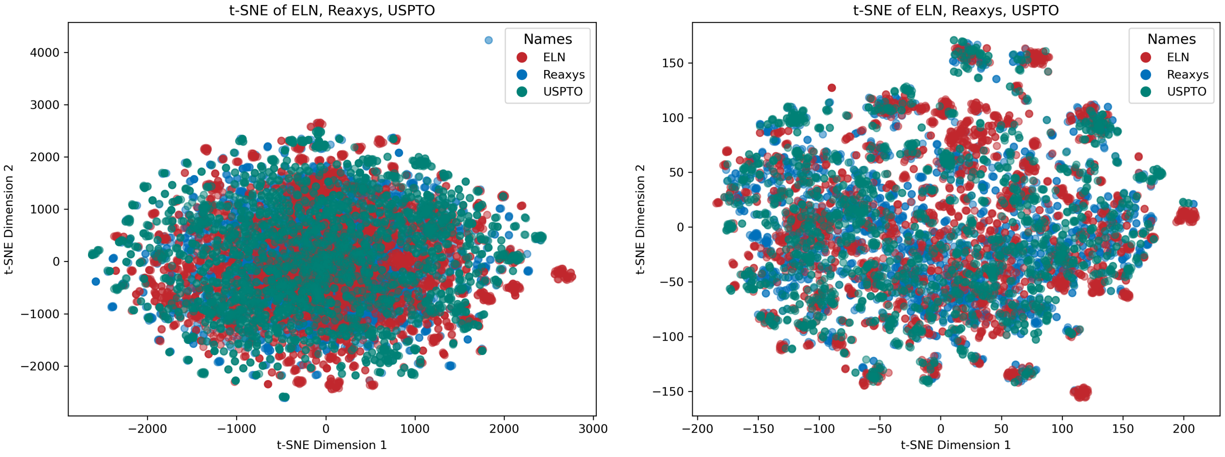
Figure A.9 The t-SNE plot depicts the distribution of reaction encodings based on Kallisto and ECFP representations for the $\mathrm{S_nAr}$ reaction.



(a) ECFP-based t-SNE

(b) Kallisto-based t-SNE

Figure A.10 The t-SNE plot depicts the distribution of reaction encodings based on Kallisto and ECFP representations for the Suzuki reaction.



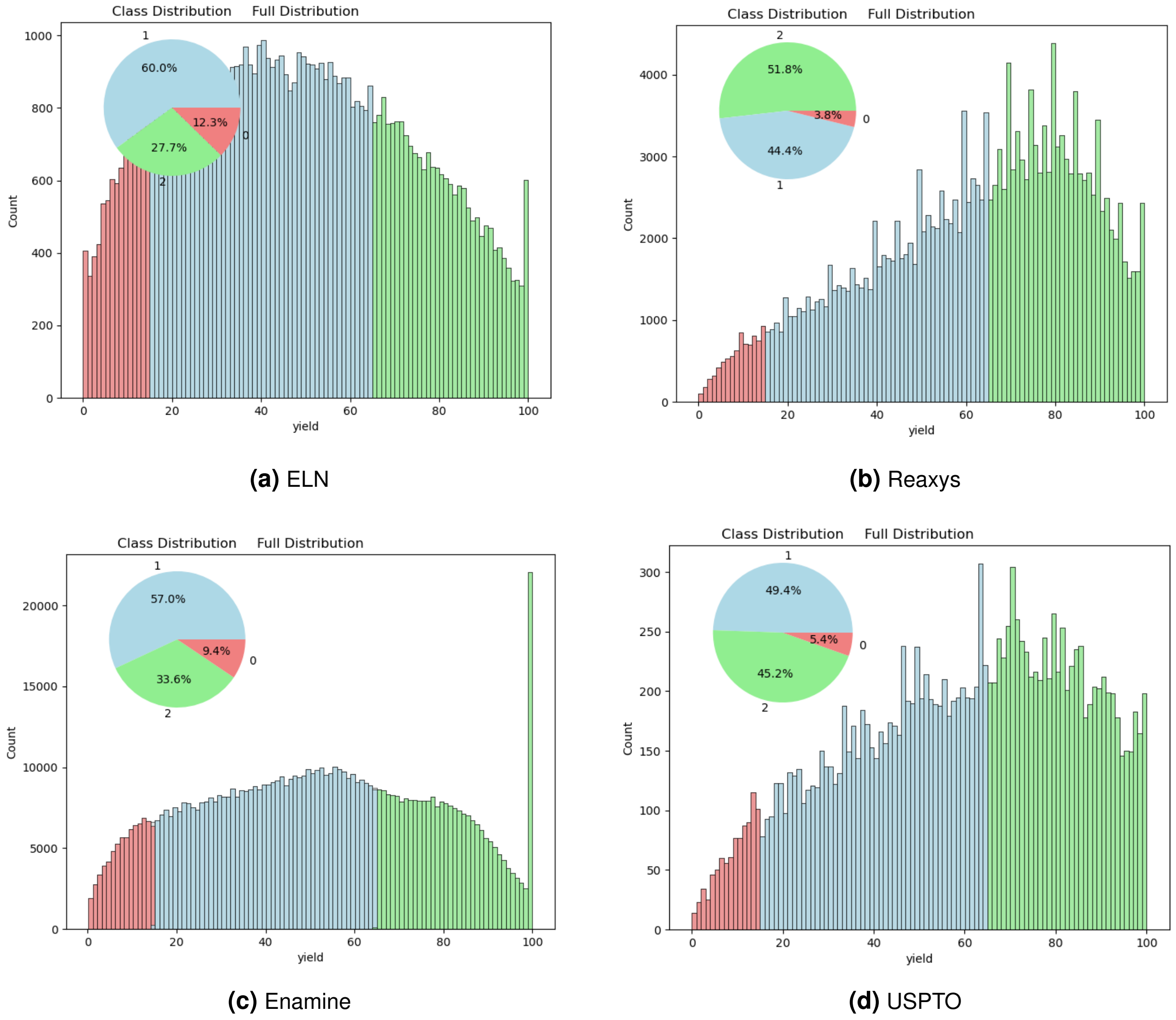
(a) ECFP-based t-SNE

(b) Kallisto-based t-SNE

Figure A.11 The t-SNE plot depicts the distribution of reaction encodings based on Kallisto and ECFP representations for the reductive amination reaction. One could see an outlier in ECFP clustering

Reaction	NextMove reaction class code	NextMove reaction class name
Amide coupling	2.1.2	Carboxylic acid + amine condensation
	2.1.3	Carboxylic acid + hydrazine condensation
	2.1.61	Carboxylic acid + amidine condensation
$\mathrm{S_nAr}$	1.3.6	Bromo N-arylation
	1.3.7	Chloro N-arylation
	1.3.8	Fluoro N-arylation
	1.3.9	Iodo N-arylation
	1.3.10	Triflyloxy N-arylation
	1.3.12	Mesyl N-arylation
	1.3.13	Mesyloxy N-arylation
	1.3.14	Tosyloxy N-arylation
Reductive amination	1.2.1	Aldehyde reductive amination
	1.2.5	Ketone reductive amination
	1.2.9	Alcohol + amine condensation
	1.2.10	Formaldehyde reductive amination
Buchwald-Hartwig coupling	1.3.1	Bromo Buchwald-Hartwig amination
	1.3.2	Chloro Buchwald-Hartwig amination
	1.3.3	Iodo Buchwald-Hartwig amination
	1.3.4	Triflyloxy Buchwald-Hartwig amination
Suzuki coupling	3.1.1	Bromo Suzuki coupling
	3.1.2	Chloro Suzuki coupling
	3.1.3	Iodo Suzuki coupling
	3.1.4	Triflyloxy Suzuki coupling
	3.1.10	Tosyloxy Suzuki coupling

Table A.2 Summary of classes selected using NextMove software



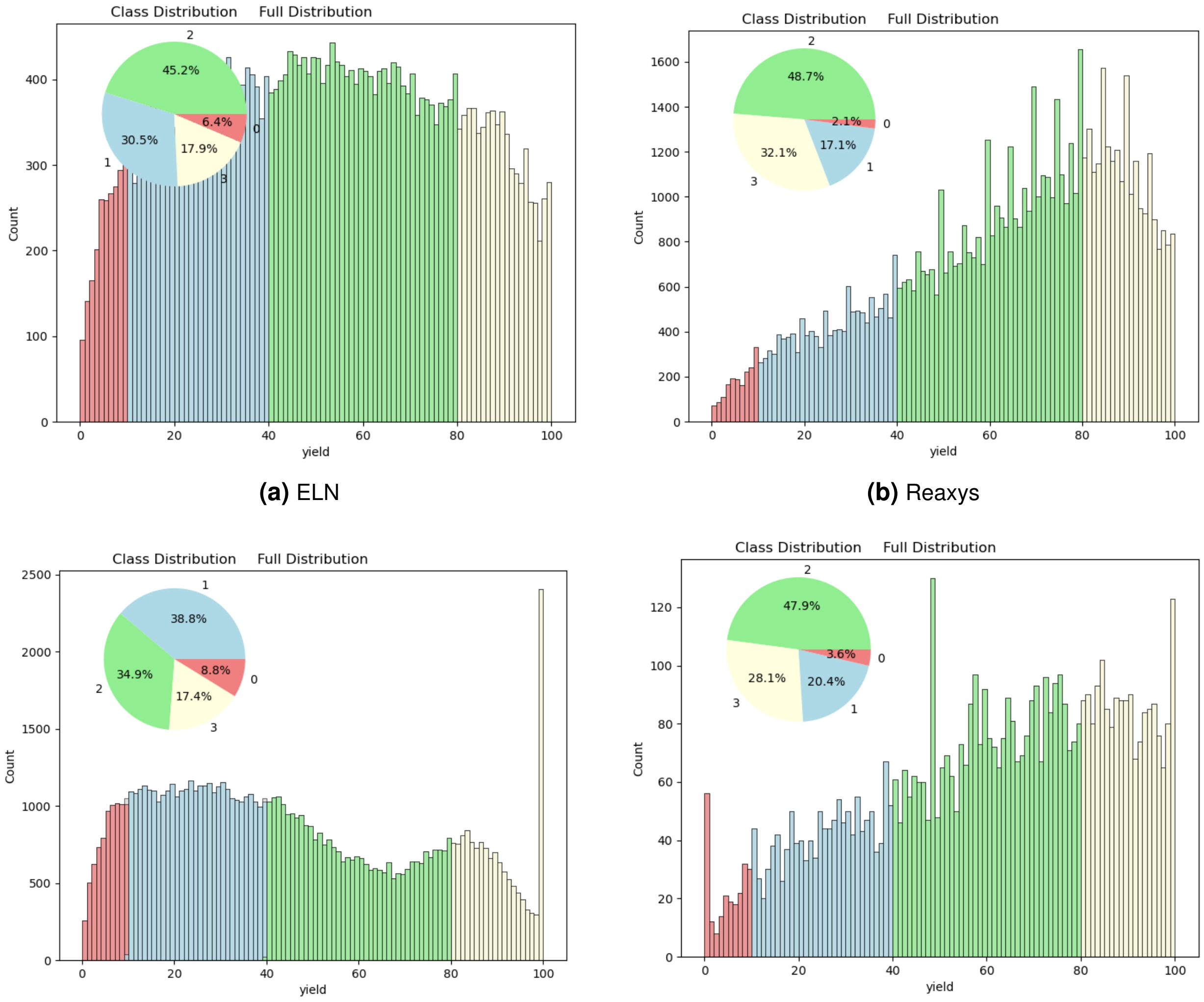
(a) ELN

(b) Reaxys

(c) Enamine

(d) USPTO

Figure A.12 Float and class distributions of the yield after reagents filtering using thresholds determined by Optuna trials. Amide coupling 3 classes



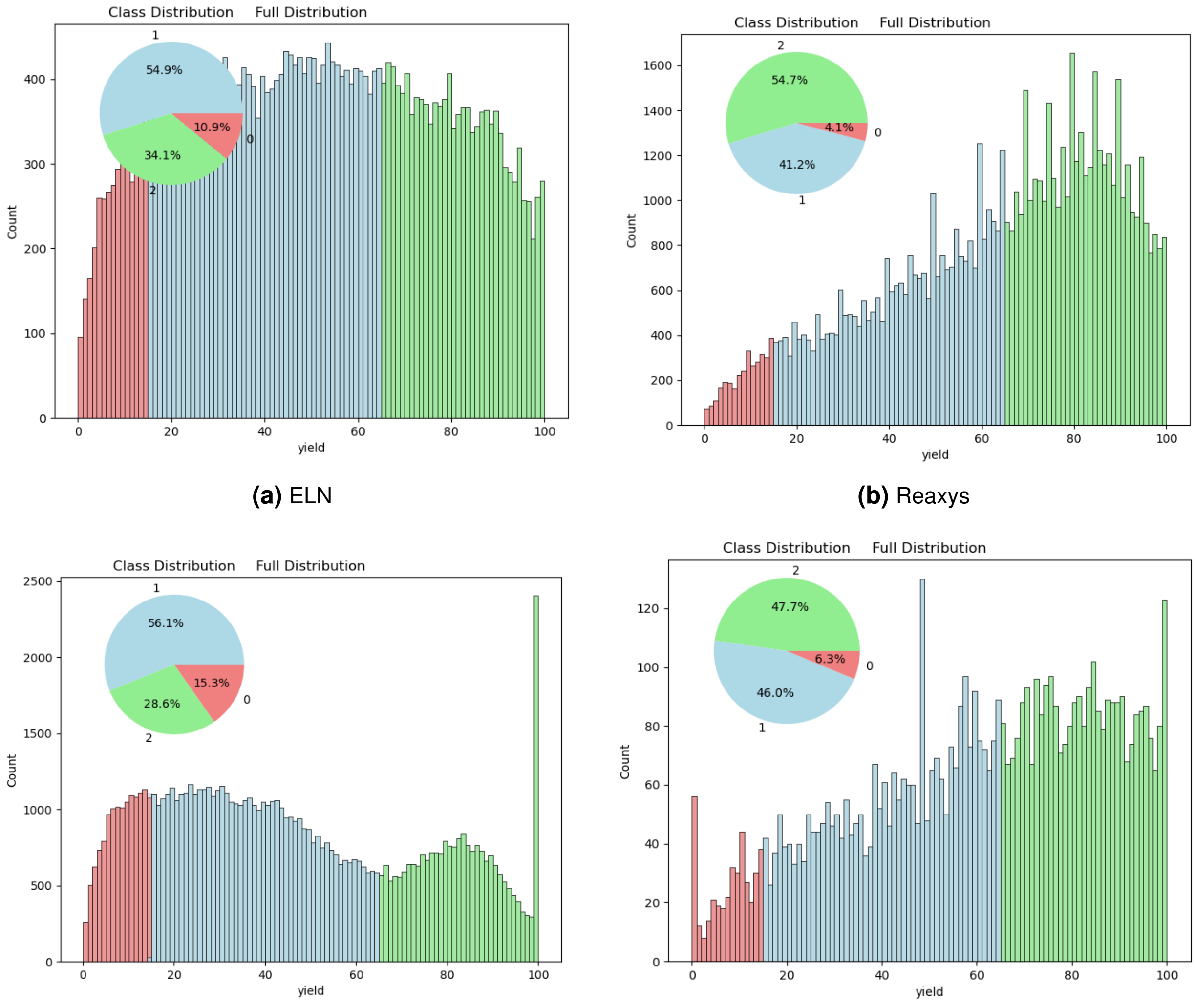
(a) ELN

(b) Reaxys

(c) Enamine

(d) USPTO

Figure A.13 Float and class distributions of the yield after reagents filtering using thresholds determined by Optuna trials. $\mathrm{S_nAr}$ 4 classes



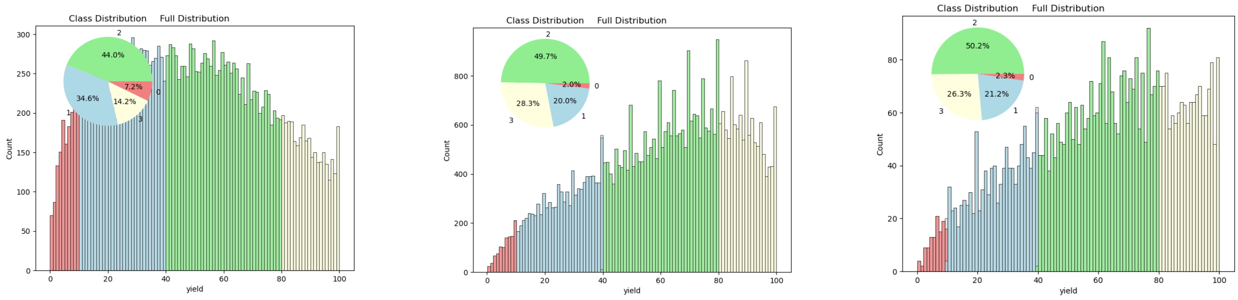
(a) ELN

(b) Reaxys

(c) Enamine

(d) USPTO

Figure A.14 Float and class distributions of the yield after reagents filtering using thresholds determined by Optuna trials. $\mathrm{S_nAr}$ 3 classes

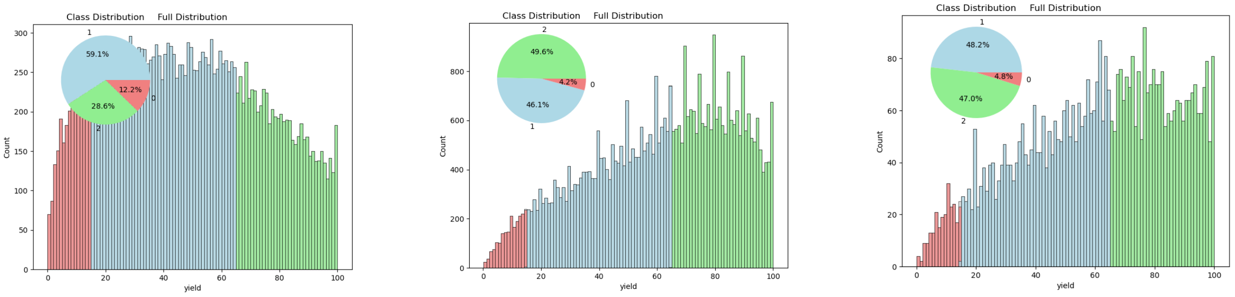


(a) ELN

(b) Reaxys

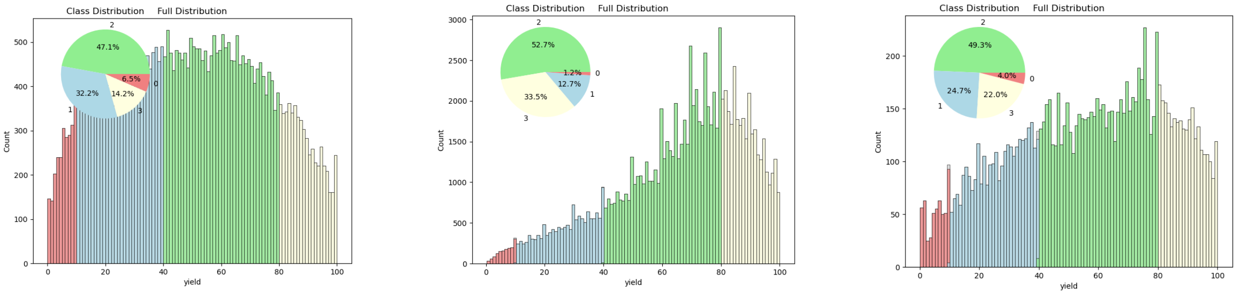
(c) USPTO

Figure A.15 Float and class distributions of the yield after reagents filtering using thresholds determined by Optuna trials. Reductive amination 4 classes



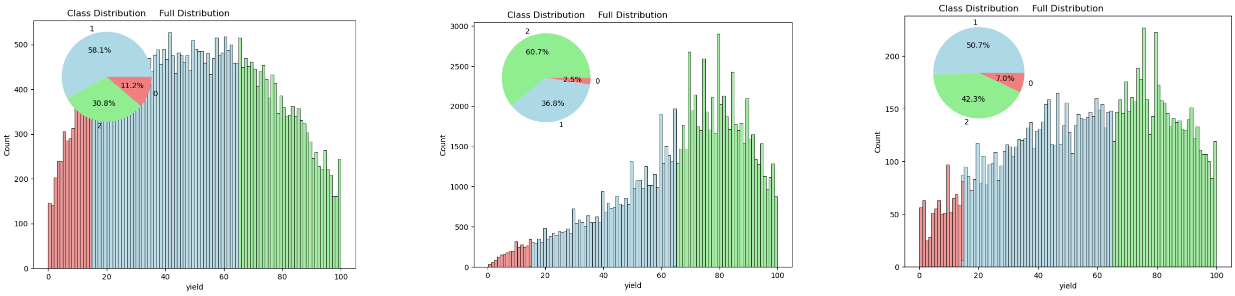
(a) ELN **(b)** Reaxys **(c)** USPTO

Figure A.16 Float and class distributions of the yield after reagents filtering using thresholds determined by Optuna trials. Reductive amination 3 classes



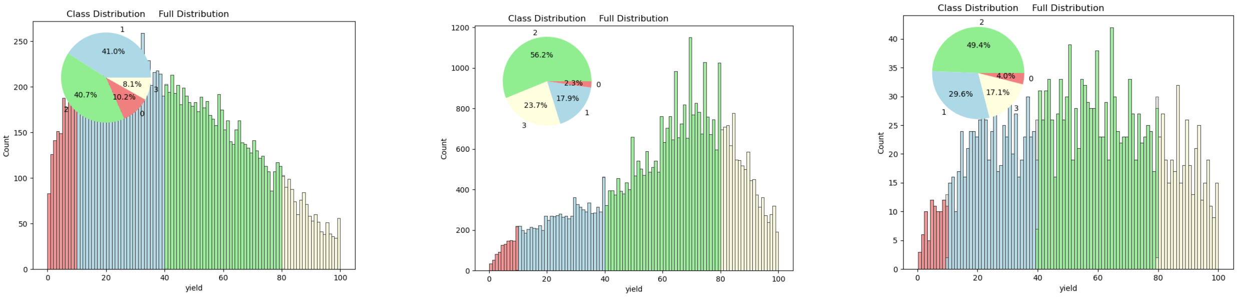
(a) ELN **(b)** Reaxys **(c)** USPTO

Figure A.17 Float and class distributions of the yield after reagents filtering using thresholds determined by Optuna trials. Suzuki coupling 4 classes



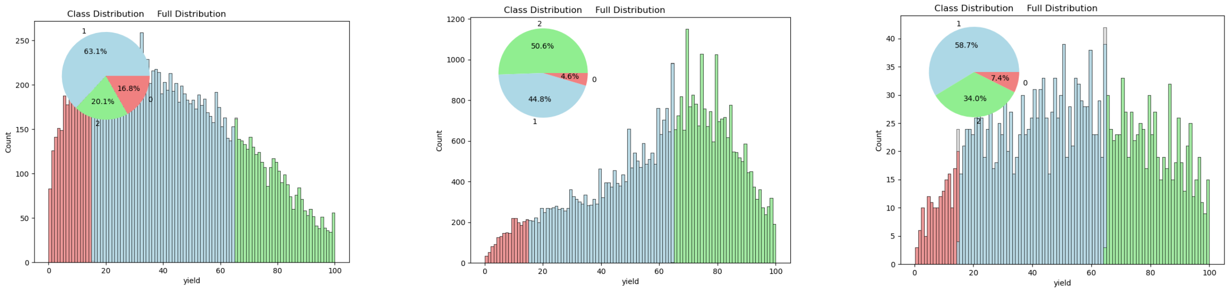
(a) ELN **(b)** Reaxys **(c)** USPTO

Figure A.18 Float and class distributions of the yield after reagents filtering using thresholds determined by Optuna trials. Suzuki coupling 3 classes



(a) ELN **(b)** Reaxys **(c)** USPTO

Figure A.19 Float and class distributions of the yield after reagents filtering using thresholds determined by Optuna trials. Buchwald-Hartwig amination 4 classes



(a) ELN

(b) Reaxys

(c) USPTO

Figure A.20 Float and class distributions of the yield after reagents filtering using thresholds determined by Optuna trials. Buchwald-Hartwig amination 3 classes

	Reaction	Original selection	Yield+fps	Reagents	Purification
ELN	Amide coupling	208333	90126	75348	52112
	$\mathrm{S_nAr}$	144106	57547	36048	25585
	Reductive amination	110147	44797	23028	16876
	Suzuki coupling	161731	63175	39783	32245
	Buchwald-Hartwig amination	69841	22124	15625	12800
USPTO	Amide coupling	104729	39847	23493	16792
	$\mathrm{S_nAr}$	74892	27045	6180	4194
	Reductive amination	64605	24482	6993	5054
	Suzuki coupling	68868	25210	12081	10320
	Buchwald-Hartwig amination	13870	5685	2201	1844
Reaxys	Amide coupling	568466	255935	200918	-
	$\mathrm{S_nAr}$	294290	149126	73905	-
	Reductive amination	705433	319071	45393	-
	Suzuki coupling	335450	204425	106776	-
	Buchwald-Hartwig amination	97384	64999	44272	-
Enamine	Amide coupling	1254391	1005331	758356	-
	$\mathrm{S_nAr}$	105175	86180	84190	-

Table A.3 Number of data points at each stage of data cleaning for the datasets for ECFP fingerprint

	Reaction	Original selection	Yield+fps	Reagents	Purification
ELN	Amide coupling	201593	88150	73823	51056
	$\mathrm{S_nAr}$	108655	42142	26102	18835
	Reductive amination	91104	37813	21646	15815
	Suzuki coupling	150660	59529	37431	30286
	Buchwald-Hartwig amination	65349	20670	14559	11977
USPTO	Amide coupling	104729	37566	22300	16009
	$\mathrm{S_nAr}$	74892	18143	3772	2697
	Reductive amination	64605	18453	6302	4659
	Suzuki coupling	68868	23268	11181	9513
	Buchwald-Hartwig amination	13870	4624	1690	1385
Reaxys	Amide coupling	543944	234358	186277	-
	$\mathrm{S_nAr}$	283873	107521	34532	-
	Reductive amination	606064	217245	39786	-
	Suzuki coupling	328015	187301	25666	-
	Buchwald-Hartwig amination	90178	55094	34188	-
Enamine	Amide coupling	1254389	995002	745657	-
	$\mathrm{S_nAr}$	105173	77209	75263	-

Table A.4 Number of data points at each stage of data cleaning for the datasets for Kallisto fingerprint

Model	R-Squared	RMSE	Time Taken
RandomForestRegressor	0.35	20.50	322.18
ExtraTreesRegressor	0.32	20.99	103.05
XGBRegressor	0.30	21.31	35.44
BaggingRegressor	0.30	21.39	32.66
LGBMRegressor	0.26	21.87	3.39
HistGradientBoostingRegressor	0.26	21.97	5.75
KNeighborsRegressor	0.22	22.44	4.03
MLPRegressor	0.20	22.77	41.08
GradientBoostingRegressor	0.17	23.28	69.51
SVR	0.16	23.42	385.18
NuSVR	0.14	23.61	403.28
LassoCV	0.10	24.20	3.53
LassoLarsCV	0.10	24.20	2.46
LassoLarsIC	0.10	24.20	1.12
ElasticNetCV	0.10	24.20	4.52
BayesianRidge	0.10	24.21	1.06
RidgeCV	0.10	24.21	1.34
Ridge	0.10	24.21	0.55
LinearRegression	0.10	24.21	0.81
TransformedTargetRegressor	0.10	24.21	0.81
PoissonRegressor	0.10	24.24	1.40
HuberRegressor	0.09	24.25	3.32
LinearSVR	0.09	24.33	10.91
GammaRegressor	0.06	24.67	0.56
AdaBoostRegressor	0.04	24.93	13.53
ElasticNet	0.04	24.97	0.61
Lasso	0.03	25.16	0.62
LassoLars	0.03	25.16	0.56
DummyRegressor	-0.00	25.49	0.40
DecisionTreeRegressor	-0.15	27.29	4.88
PassiveAggressiveRegressor	-0.89	35.02	0.91
KernelRidge	-3.41	53.50	377.87
GaussianProcessRegressor	-5.00	62.44	4122.58

Table A.5 LazyPredict executed on yield-cleaned Kallisto FP Regression models.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	**0.508**	0.273	0.431	0.233	0.447	0.26	0.403	0.203
	Reaxys	**0.306**	0.081	0.3	0.11	0.248	-0.008	0.272	0.058
	ELN	0.287	0.047	**0.287**	0.06	0.276	0.033	0.255	0.004
	Enamine	**0.271**	0.034	0.257	0.012	0.246	-0.008	0.249	0.001
Reaxys	Inner test	0.467	0.282	0.461	0.315	**0.467**	0.358	0.447	0.29
	USPTO	**0.541**	0.401	0.42	0.371	0.259	0.035	0.258	0.016
	ELN	**0.322**	0.101	0.284	0.073	0.264	0.032	0.316	0.128
	Enamine	0.28	0.044	0.251	0.001	0.258	0.018	**0.292**	0.073
ELN	Inner test	0.476	0.264	0.472	0.294	**0.546**	0.442	0.467	0.305
	Libs test	0.245	-0.01	0.242	-0.028	0.227	-0.038	**0.264**	0.056
	USPTO	**0.308**	0.093	0.291	0.065	0.256	0.014	0.279	0.05
	Reaxys	0.288	0.057	0.29	0.054	0.295	0.03	**0.305**	0.09
	Enamine	0.285	0.055	0.273	0.06	0.27	0.016	**0.287**	0.065

Table A.6 Results from $\mathrm{S_nAr}$ trained reagents filtered datasets, 4 classes.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.542	0.262	**0.576**	0.374	0.541	0.339	0.47	0.225
	Reaxys	**0.398**	0.124	0.391	0.143	0.342	0.023	0.355	0.072
	ELN	**0.388**	0.106	0.381	0.104	0.351	0.043	0.349	0.058
	Enamine	**0.369**	0.061	0.366	0.066	0.329	-0.011	0.363	0.076
Reaxys	Inner test	**0.557**	0.347	0.539	0.379	0.53	0.407	0.521	0.342
	USPTO	**0.597**	0.42	0.488	0.38	0.326	-0.034	0.353	0.091
	ELN	0.409	0.156	0.396	0.139	0.34	0.017	**0.416**	0.173
	Enamine	0.381	0.086	0.37	0.073	0.355	0.044	**0.393**	0.13
ELN	Inner test	0.584	0.34	0.589	0.385	**0.616**	0.518	0.57	0.376
	Libs test	0.332	0.011	0.333	0.000	0.318	-0.037	**0.359**	0.086
	USPTO	**0.397**	0.119	0.366	0.155	0.33	-0.01	0.373	0.106
	Reaxys	0.384	0.087	0.352	0.114	0.357	0.039	**0.39**	0.138
	Enamine	**0.379**	0.08	0.334	0.016	0.341	0.01	0.376	0.113

Table A.7 Results from $\mathrm{S_nAr}$ trained reagents filtered datasets, 3 classes.

	Test set	RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
		BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.444	0.208	**0.462**	0.256	0.371	0.191	0.461	0.211
	Reaxys	**0.335**	0.108	0.272	0.11	0.254	0.002	0.276	0.093
	ELN	**0.281**	0.042	0.259	0.035	0.264	0.015	0.255	0.006
Reaxys	Inner test	0.438	0.268	0.428	0.342	**0.474**	0.397	0.443	0.324
	USPTO	**0.482**	0.316	0.273	0.056	0.252	0.015	0.25	-0.001
	ELN	0.284	0.059	0.26	0.013	0.26	0.007	**0.295**	0.118
ELN	Inner test	0.468	0.247	0.433	0.248	**0.517**	0.397	0.426	0.288
	Libs test	0.262	0.05	0.252	0.009	0.24	0.015	**0.304**	0.134
	USPTO	**0.291**	0.059	0.264	0.034	0.25	0.002	0.261	0.023
	Reaxys	0.273	0.024	0.255	-0.004	0.278	0.049	**0.298**	0.061

Table A.8 Results from Suzuki coupling trained reagents filtered datasets, 4 classes.

	Test set	RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
		BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.534	0.265	0.511	0.279	0.54	0.336	**0.564**	0.285
	Reaxys	**0.415**	0.136	0.408	0.21	0.35	0.047	0.329	-0.001
	ELN	**0.374**	0.081	0.364	0.078	0.344	0.031	0.342	0.028
Reaxys	Inner test	0.527	0.322	0.522	0.435	**0.539**	0.426	0.533	0.382
	USPTO	**0.54**	0.35	0.368	0.115	0.336	0.015	0.349	0.053
	ELN	0.377	0.092	0.336	0.008	0.349	0.033	**0.396**	0.14
ELN	Inner test	0.538	0.268	0.516	0.295	**0.601**	0.473	0.521	0.301
	Libs test	0.358	0.058	0.333	0.016	**0.369**	0.079	0.366	0.074
	USPTO	**0.385**	0.097	0.345	0.068	0.333	-0.002	0.334	0.016
	Reaxys	0.36	0.063	0.34	0.061	0.314	-0.045	**0.389**	0.164

Table A.9 Results from Suzuki coupling trained reagents filtered datasets, 3 classes.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.385	0.176	**0.393**	0.201	0.365	0.236	0.355	0.184
	Reaxys	**0.309**	0.089	0.298	0.127	0.271	0.047	0.274	0.054
	ELN	**0.276**	0.029	0.27	0.033	0.252	0.002	0.25	0
Reaxys	Inner test	0.446	0.261	0.436	0.309	**0.502**	0.39	0.431	0.299
	USPTO	**0.449**	0.285	0.375	0.289	0.251	0.027	0.262	0.021
	ELN	0.289	0.048	0.273	0.049	0.267	0.036	**0.328**	0.158
ELN	Inner test	0.472	0.261	0.462	0.293	**0.514**	0.373	0.469	0.306
	Libs test	0.257	0.103	0.317	0.232	0.306	0.19	**0.324**	0.236
	USPTO	**0.286**	0.042	0.276	0.032	0.248	-0.029	0.252	-0.001
	Reaxys	0.279	0.036	0.279	0.037	0.26	0.019	**0.301**	0.1

Table A.10 Results from reductive amination trained reagents filtered datasets, 4 classes.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.462	0.189	0.451	0.259	0.477	0.292	**0.487**	0.292
	Reaxys	0.393	0.107	**0.401**	0.171	0.34	0.02	0.344	0.041
	ELN	0.364	0.052	**0.38**	0.098	0.342	0.017	0.337	0.038
Reaxys	Inner test	0.523	0.313	0.507	0.365	**0.538**	0.422	0.519	0.348
	USPTO	**0.531**	0.326	0.476	0.334	0.341	0.043	0.356	0.074
	ELN	0.385	0.094	0.39	0.115	0.373	0.082	**0.404**	0.173
ELN	Inner test	0.561	0.3	0.556	0.338	**0.591**	0.426	0.572	0.349
	Libs test	0.34	-0.052	0.329	-0.027	0.355	0.04	**0.38**	0.151
	USPTO	**0.374**	0.087	0.347	0.071	0.337	-0.01	0.338	0.054
	Reaxys	0.364	0.057	0.347	0.091	0.334	0.027	**0.403**	0.172

Table A.11 Results from reductive amination trained reagents filtered datasets, 3 classes.

	Test set	RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
		BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.396	0.167	**0.414**	0.195	0.377	0.268	0.364	0.193
	Reaxys	**0.326**	0.101	0.303	0.119	0.254	0.005	0.284	0.079
	ELN	**0.286**	0.077	0.279	0.067	0.242	-0.027	0.25	0
Reaxys	Inner test	**0.491**	0.326	0.487	0.368	0.484	0.387	0.485	0.363
	USPTO	**0.594**	0.441	0.286	0.169	0.228	-0.038	0.285	0.075
	ELN	0.299	0.083	0.252	0.019	0.250	0.005	**0.317**	0.118
ELN	Inner test	0.456	0.214	0.447	0.259	**0.488**	0.370	0.407	0.226
	Libs test	0.368	0.084	0.311	-0.052	0.333	0.000	**0.544**	0.301
	USPTO	**0.288**	0.057	0.271	0.044	0.252	0.021	0.266	0.054
	Reaxys	0.285	0.036	0.297	0.086	0.262	0.007	**0.325**	0.141

Table A.12 Results from Buchwald-Hartwig amination trained reagents filtered datasets, 4 classes.

	Test set	RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
		BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.488	0.236	**0.541**	0.296	0.443	0.240	0.425	0.219
	Reaxys	**0.406**	0.137	0.379	0.165	0.327	-0.019	0.336	0.018
	ELN	0.368	0.049	0.353	0.054	**0.388**	0.077	0.348	0.044
Reaxys	Inner test	0.580	0.387	0.569	0.445	**0.585**	0.452	0.556	0.413
	USPTO	**0.657**	0.459	0.450	0.323	0.333	0.000	0.364	0.081
	ELN	**0.398**	0.096	0.351	0.036	0.364	0.047	0.381	0.071
ELN	Inner test	0.552	0.255	0.551	0.306	**0.567**	0.366	0.55	0.305
	Libs test	0.436	-0.096	0.500	0.000	0.522	0.213	**0.709**	0.412
	USPTO	**0.367**	0.044	0.336	0.050	0.335	0.012	0.335	0.006
	Reaxys	0.392	0.081	0.333	0.008	0.335	-0.002	**0.41**	0.142

Table A.13 Results from Buchwald-Hartwig amination trained reagents filtered datasets, 3 classes.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.45	0.242	0.42	0.252	**0.498**	0.332	0.357	0.157
	Reaxys	**0.349**	0.125	0.323	0.145	0.252	0.006	0.253	0.021
	ELN	**0.306**	0.077	0.294	0.081	0.255	0.017	0.251	0.008
	Enamine	**0.279**	0.048	0.256	0.027	0.250	0.002	0.251	0.005
Reaxys	Inner test	**0.472**	0.284	0.46	0.331	0.462	0.362	0.433	0.279
	USPTO	**0.501**	0.339	0.421	0.351	0.261	0.037	0.277	0.055
	ELN	**0.332**	0.114	0.281	0.07	0.267	0.032	0.312	0.128
	Enamine	**0.282**	0.05	0.251	0.004	0.249	-0.003	0.277	0.065
ELN	Inner test	0.488	0.274	0.481	0.293	**0.663**	0.562	0.455	0.297
	Libs test	**0.339**	0.073	0.284	0.101	0.192	0.013	0.339	0.029
	USPTO	**0.313**	0.085	0.291	0.072	0.259	0.011	0.267	0.039
	Reaxys	0.3	0.061	0.284	0.039	0.261	-0.001	**0.302**	0.081
	Enamine	0.288	0.058	0.272	0.056	0.262	0.028	**0.288**	0.07

Table A.14 Results from $\mathrm{S_nAr}$ trained on only yield filtered datasets, 4 classes.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.525	0.282	0.511	0.324	**0.558**	0.364	0.486	0.23
	Reaxys	**0.429**	0.168	0.419	0.199	0.347	0.046	0.369	0.101
	ELN	0.4	0.133	**0.4**	0.156	0.346	0.036	0.368	0.081
	Enamine	**0.372**	0.066	0.351	0.045	0.338	0.014	0.35	0.043
Reaxys	Inner test	0.545	0.343	**0.545**	0.406	0.533	0.415	0.509	0.353
	USPTO	**0.565**	0.381	0.49	0.362	0.353	0.05	0.367	0.104
	ELN	**0.419**	0.158	0.396	0.136	0.36	0.059	0.416	0.188
	Enamine	0.381	0.086	0.366	0.064	0.32	-0.024	**0.386**	0.12
ELN	Inner test	0.58	0.341	0.568	0.359	**0.773**	0.651	0.565	0.374
	Libs test	0.387	0.081	0.332	-0.02	0.333	-0.013	**0.428**	0.125
	USPTO	**0.397**	0.122	0.361	0.139	0.354	0.06	0.36	0.074
	Reaxys	0.387	0.091	0.348	0.095	0.349	0.049	**0.386**	0.125
	Enamine	**0.384**	0.091	0.335	0.021	0.332	-0.001	0.375	0.113

Table A.15 Results from $\mathrm{S_nAr}$ trained only yield filtered datasets, 3 classes.

	Test set	RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
		BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	**0.428**	0.223	0.406	0.254	0.419	0.277	0.351	0.171
	Reaxys	**0.319**	0.102	0.277	0.097	0.282	0.061	0.276	0.081
	ELN	**0.301**	0.066	0.285	0.06	0.290	0.072	0.285	0.044
Reaxys	Inner test	**0.493**	0.34	0.465	0.36	0.437	0.379	0.418	0.313
	USPTO	**0.485**	0.314	0.292	0.149	0.259	0.024	0.269	0.058
	ELN	**0.319**	0.1	0.277	0.055	0.265	0.027	0.33	0.148
ELN	Inner test	0.517	0.303	0.506	0.31	**0.580**	0.423	0.458	0.307
	Libs test	0.251	0.012	0.279	0.089	0.242	0.098	**0.268**	0.071
	USPTO	**0.3**	0.074	0.285	0.053	0.264	0.017	0.28	0.068
	Reaxys	0.294	0.05	0.272	0.011	0.271	0.027	**0.315**	0.097

Table A.16 Results from reductive amination trained on only yield filtered datasets, 4 classes.

	Test set	RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
		BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	**0.518**	0.304	0.499	0.313	0.483	0.303	0.453	0.23
	Reaxys	**0.414**	0.171	0.387	0.148	0.350	0.045	0.368	0.095
	ELN	0.396	0.127	**0.399**	0.14	0.385	0.117	0.367	0.126
Reaxys	Inner test	**0.566**	0.385	0.534	0.419	0.507	0.429	0.497	0.382
	USPTO	**0.56**	0.36	0.403	0.203	0.345	0.032	0.374	0.11
	ELN	0.404	0.149	0.36	0.061	0.350	0.037	**0.422**	0.196
ELN	Inner test	0.587	0.34	0.569	0.338	**0.665**	0.498	0.569	0.372
	Libs test	0.372	0.027	0.357	0.065	**0.410**	0.030	0.341	0.059
	USPTO	**0.387**	0.117	0.36	0.104	0.365	0.096	0.355	0.09
	Reaxys	0.394	0.137	0.346	0.067	0.363	0.081	**0.395**	0.173

Table A.17 Results from reductive amination trained on only yield filtered datasets, 3 classes.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.427	0.208	0.428	0.252	**0.497**	0.370	0.387	0.175
	Reaxys	**0.374**	0.174	0.311	0.216	0.243	-0.014	0.281	0.104
	ELN	**0.28**1	0.04	0.258	0.03	0.256	0.001	0.255	0.009
Reaxys	Inner test	**0.46**	0.3	0.442	0.353	0.294	0.218	0.417	0.309
	USPTO	**0.57**	0.425	0.304	0.168	0.250	0.003	0.251	0.008
	ELN	0.294	0.069	0.256	0.019	0.251	0.006	**0.305**	0.129
ELN	Inner test	**0.454**	0.233	0.441	0.26	0.445	0.352	0.428	0.257
	Libs test	**0.315**	0.099	0.239	-0.032	0.248	-0.021	0.3	0.138
	USPTO	**0.291**	0.056	0.272	0.048	0.269	0.035	0.259	0.029
	Reaxys	0.287	0.038	0.264	0.007	0.264	0.005	**0.286**	0.07

Table A.18 Results from Suzuki coupling trained on only yield filtered datasets, 4 classes.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	0.512	0.253	0.515	0.324	**0.533**	0.358	0.515	0.277
	Reaxys	**0.452**	0.209	0.422	0.252	0.36	0.099	0.363	0.082
	ELN	**0.378**	0.09	0.37	0.096	0.337	0.021	0.35	0.053
Reaxys	Inner test	**0.535**	0.338	0.514	0.392	0.371	0.215	0.487	0.363
	USPTO	**0.63**	0.45	0.437	0.277	0.335	0.012	0.365	0.098
	ELN	**0.381**	0.098	0.352	0.043	0.333	-0.001	0.393	0.143
ELN	Inner test	0.54	0.278	0.529	0.313	0.506	0.402	**0.52**	0.323
	Libs test	**0.365**	0.089	0.333	0	0.351	0.009	0.361	0.064
	USPTO	**0.377**	0.086	0.343	0.073	0.336	0.019	0.333	0
	Reaxys	0.366	0.069	0.343	0.087	0.336	0.021	**0.403**	0.194

Table A.19 Results from Suzuki coupling trained on only yield filtered datasets, 3 classes.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	**0.442**	0.206	0.416	0.248	0.395	0.232	0.338	0.158
	Reaxys	**0.373**	0.159	0.341	0.194	0.251	0.018	0.289	0.093
	ELN	**0.277**	0.051	0.266	0.047	0.250	0.008	0.25	0.003
Reaxys	Inner test	0.503	0.343	0.492	0.392	**0.498**	0.457	0.461	0.342
	USPTO	**0.629**	0.471	0.333	0.232	0.295	0.058	0.266	0.035
	ELN	**0.309**	0.081	0.255	0.01	0.270	-0.001	0.305	0.148
ELN	Inner test	0.455	0.21	0.441	0.247	**0.462**	0.343	0.423	0.251
	Libs test	0.398	0.117	0.264	-0.192	0.333	0	**0.468**	0.204
	USPTO	**0.281**	0.047	0.275	0.06	0.266	0.045	0.255	0.04
	Reaxys	0.278	0.015	0.281	0.071	0.258	0.012	**0.293**	0.067

Table A.20 Results from Buchwald-Hartwig amination trained on only yield filtered datasets, 4 classes.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
USPTO	Inner test	**0.501**	0.209	0.466	0.238	0.439	0.184	0.459	0.19
	Reaxys	**0.445**	0.184	0.405	0.23	0.34	0.007	0.344	0.056
	ELN	**0.367**	0.046	0.348	0.047	0.334	0.001	0.348	0.041
Reaxys	Inner test	0.583	0.391	0.58	0.469	**0.587**	0.507	0.56	0.403
	USPTO	**0.679**	0.482	0.443	0.287	0.378	0.088	0.363	0.065
	ELN	0.389	0.085	0.349	0.032	0.388	0.069	**0.409**	0.107
ELN	Inner test	0.553	0.262	0.55	0.303	**0.553**	0.373	0.525	0.277
	Libs test	0.546	0.103	0.5	0	0.5	0	**0.685**	0.39
	USPTO	**0.368**	0.038	0.337	0.052	0.348	0.032	0.338	0.02
	Reaxys	0.375	0.029	0.333	-0.003	0.344	0.001	**0.382**	0.062

Table A.21 Results from Buchwald-Hartwig amination trained on only yield filtered datasets, 3 classes.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	RMSE	R2	RMSE	R2	RMSE	R2	RMSE	R2
USPTO	Inner test	23.64	0.144	**23.088**	0.184	23.216	0.137	35.3	-1.101
	Reaxys	24.359	0.035	**23.89**	0.078	27.656	-0.245	37.823	-1.326
	ELN	27.986	-0.143	**27.393**	-0.1	31.631	-0.459	30.365	-0.346
Reaxys	Inner test	21.267	0.265	20.333	0.341	**19.927**	0.39	22.176	0.198
	USPTO	**22.971**	0.184	23.803	0.132	38.858	-1.389	25.435	-0.001
	ELN	28.215	-0.162	27.511	-0.11	28.955	-0.222	**25.749**	0.032
ELN	Inner test	21.765	0.308	20.863	0.357	**20.725**	0.354	22.451	0.256
	Libs test	22.366	0.349	**21.576**	0.39	44.157	-3.082	20.724	0.101
	USPTO	27.224	-0.146	27.559	-0.164	65.023	-5.688	**26.48**	-0.085
	Reaxys	27.453	-0.226	27.448	-0.217	59.758	-4.812	**27.321**	-0.214

Table A.22 Results from reductive amination trained reagents filtered datasets, regression.

		RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
	Test set	RMSE	R2	RMSE	R2	RMSE	R2	RMSE	R2
USPTO	Inner test	22.821	0.218	21.982	0.291	**22.188**	0.264	26.35	-0.036
	Reaxys	**22.98**	-0.07	24.643	-0.121	29.779	-0.791	27.308	-0.511
	ELN	26.2	-0.038	26.232	-0.034	40.586	-1.469	**25.722**	-0
Reaxys	Inner test	19.175	0.257	18.533	0.364	**17.809**	0.35	19.136	0.268
	USPTO	**23.744**	0.155	26.307	-0.028	55.737	-3.659	25.922	-0.008
	ELN	28.146	-0.197	29.401	-0.299	38.205	-1.188	**25.271**	0.035
ELN	Inner test	21.836	0.284	21.433	0.322	**19.136**	0.441	22.352	0.25
	Libs test	22.078	0.287	**21.847**	0.308	47.748	-3.149	23.752	-0.003
	USPTO	26.337	-0.04	26.678	-0.057	61.38	-4.65	**25.817**	0.001
	Reaxys	28.638	-0.661	30.974	-0.77	43.872	-2.888	**23.838**	-0.151

Table A.23 Results from Suzuki coupling trained on reagents filtered datasets, regression.

	Test set	RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
		RMSE	R2	RMSE	R2	RMSE	R2	RMSE	R2
USPTO	Inner test	**22.116**	0.285	22.511	0.284	23.141	0.201	38.594	-1.235
	Reaxys	**23.964**	0.052	24.642	0.031	27.813	-0.278	42.152	-1.933
	ELN	27.464	-0.058	**27.17**	-0.04	33.785	-0.6	34.624	-0.681
	Enamine	31.177	-0.232	**30.111**	-0.152	37.99	-0.83	32.758	-0.36
Reaxys	Inner test	20.254	0.314	20.339	0.343	**19.493**	0.343	21.652	0.237
	USPTO	**21.953**	0.295	24.094	0.186	43.221	-1.733	26.205	-0.005
	ELN	27.053	-0.026	27.237	-0.045	32.348	-0.467	**25.717**	0.073
	Enamine	31.351	-0.246	30.469	-0.179	37.786	-0.81	**28.978**	-0.065
ELN	Inner test	21.303	0.353	20.579	0.407	**18.927**	0.467	22.417	0.313
	Libs test	21.368	0.377	**20.414**	0.427	28.566	-0.795	21.325	-0
	USPTO	26.4	-0.02	28.097	-0.107	35.996	-0.896	**26.17**	-0.002
	Reaxys	27.571	-0.255	28.415	-0.289	42.945	-2.048	**26.4**	-0.15
	Enamine	27.618	0.033	**27.595**	0.033	33.477	-0.421	28.107	-0.002

Table A.24 Results from $\mathrm{S_nAr}$ trained on reagents filtered datasets, regression.

	Test set	RF ECFP		RF Kallisto		Chemprop		Yield-BERT	
		RMSE	R2	RMSE	R2	RMSE	R2	RMSE	R2
USPTO	Inner test	**21.948**	0.187	23.41	0.199	22.538	0.247	44.687	-2.096
	Reaxys	23.317	0.006	23.828	-0.014	**22.815**	0.05	53.006	-4.139
	ELN	**25.773**	-0.086	25.811	-0.084	31.491	-0.621	37.87	-1.344
Reaxys	Inner test	18.736	0.363	17.994	0.42	**17.646**	0.415	19.514	0.297
	USPTO	**21.036**	0.306	24.192	0.126	50.412	-2.983	25.345	-0.007
	ELN	26.913	-0.184	27.523	-0.232	34.626	-0.96	**24.992**	-0.021
ELN	Inner test	20.749	0.297	19.968	0.328	**19.544**	0.365	22.056	0.189
	Libs test	21.055	0.286	**20.566**	0.318	30.101	-2.584	15.856	0.006
	USPTO	27.19	-0.159	27.495	-0.129	40.856	-1.616	**25.589**	-0.026
	Reaxys	29.582	-0.601	29.244	-0.527	26.836	-0.315	**28.402**	-0.476

Table A.25 Results from Buchwald-Hartwig amination trained on reagents filtered datasets, regression.

Table A.26 Results from reductive amination trained on fully filtered ELN and evaluated on the inner test set of ELN and fully cleaned USPTO.

		3 classes				4 classes			
		RF ECFP		RF Kallisto		RF ECFP		RF Kallisto	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
ELN	Inner test	0.529	0.246	0.533	0.3	0.452	0.224	0.46	0.28
	USPTO	0.361	0.077	0.336	0.028	0.268	0.033	0.272	0.043

Table A.27 Results from $\mathrm{S_nAr}$ trained on fully filtered ELN and evaluated on the inner test set of ELN and fully cleaned USPTO.

		3 classes				4 classes			
		RF ECFP		RF Kallisto		RF ECFP		RF Kallisto	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
ELN	Inner test	0.548	0.286	0.55	0.327	0.469	0.245	0.468	0.29
	USPTO	0.392	0.101	0.348	0.097	0.287	0.052	0.277	0.054

Table A.28 Results from Suzuki coupling trained on fully filtered ELN and evaluated on the inner test set of ELN and fully cleaned USPTO.

		3 classes				4 classes			
		RF ECFP		RF Kallisto		RF ECFP		RF Kallisto	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
ELN	Inner test	0.527	0.264	0.515	0.3	0.432	0.212	0.423	0.26
	USPTO	0.387	0.107	0.347	0.085	25.803	-0.035	25.85	-0.029

Table A.29 Results from Buchwald-Hartwig coupling trained on fully filtered ELN and evaluated on the inner test set of ELN and fully cleaned USPTO.

		3 classes				4 classes			
		RF ECFP		RF Kallisto		RF ECFP		RF Kallisto	
	Test set	BA	MCC	BA	MCC	BA	MCC	BA	MCC
ELN	Inner test	0.529	0.232	0.53	0.275	0.468	0.242	0.423	0.238
	USPTO	0.375	0.056	0.34	0.098	0.285	0.052	0.276	0.063

Table A.30 Results trained on fully filtered ELN and evaluated on the inner test set of ELN and fully cleaned USPTO.

		RF ECFP		RF Kallisto	
	Test set	RMSE	R2	RMSE	R2
S_nAr	Inner test	21.081	0.313	20.639	0.337
	USPTO	26.557	-0.06	27.673	-0.124
RA	ELN	21.603	0.257	20.43	0.322
	USPTO	26.997	-0.157	26.977	-0.153
BH	Inner test	20.678	0.287	19.617	0.345
	USPTO	26.07	-0.103	26.475	-0.081
AC	Inner test	21.345	0.272	20.397	0.33
	USPTO	26.272	-0.08	26.258	-0.078
Suzuki	Inner test	21.458	0.273	20.702	0.326
	USPTO	25.803	-0.035	25.85	-0.029