UNIVERSITY OF CALIFORNIA,
IRVINE

Machine Intelligence for Chemistry: From Deep Learning Architectures to Open Data

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Computer Science

by

Mohammadamin Tavakoli

Dissertation Committee:
Pierre Baldi, Chair
David Van Vranken
Sameer Singh

2023

DEDICATION

To my parents, who sacrificed so much to help me become a better person. Thank you for everything.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I wish to acknowledge financial support from the National Science Foundation grant NSF 195811, NRT 1633631, Army Research Office grant W911NF1810349, and hardware contributions from NVIDIA.

Sections of this dissertation are based on pre-print material connected to publications of which I am the first author in the Journal of Chemical Information and Modeling and AAAI Conference. I wish to acknowledge them for granting permission to incorporate the material in this dissertation.

I wish to thank all co-authors of those publications, whose contributions have made this dissertation possible by providing the necessary research foundation.

Finally, I extend my deepest appreciation to my advisor, Pierre Baldi, for providing this opportunity and for his invaluable guidance and support.

VITA

Mohammadamin Tavakoli

EDUCATION

Doctor of Philosophy in Computer Science **2023**
University of California, Irvine *Irvine, California*

Bachelor of Science in Physics **2016**
Sharif University of Technology *Tehran, Iran*

Bachelor of Science in Mechanical Engineering **2016**
Sharif University of Technology *Tehran, Iran*

RESEARCH EXPERIENCE

Graduate Research Assistant **2017–2023**
University of California, Irvine *Irvine, California*

Research Assistant **2015–2016**
Sharif University of Technology *Tehran, Iran*

TEACHING EXPERIENCE

Deep Learning and Neural Networks Teaching Assistant **2020**
CS273 - University of California, Irvine *Irvine, California*

Machine Learning Teaching Assistant **2019**
CS273 - University of California, Irvine *Irvine, California*

Computer Vision and Image Understanding Teaching Assistant **2018**
CS295 - University of California, Irvine *Irvine, California*

REFEREED JOURNAL and CONFERENCE PUBLICATIONS

RMechDB: A Public Database of Elementary Radical Reaction Steps **2023**
Journal of Chemical Information and Modeling

Deep learning models of the discrete component of the Galactic interstellar -ray emission **2023**
Physical Review D

Improved Modeling of the Discrete Component of the Galactic Interstellar Gamma-ray Emission and Implications for the Fermi-LAT Galactic Center Excess **2022**
Physcial Review D

Quantum mechanics and machine learning synergies: graph attention neural networks to predict chemical reactivity **2022**
Journal of Chemical Information and Modeling

Rxn Hypergraph: a Hypergraph Attention Model for Chemical Reaction Representation **2022**
AAAI 2022, Deep Learning on Graphs: Methods and Applications

Splash: Learnable activation functions for improving accuracy and adversarial robustness **2021**
Neural Networks

Methyl cation affinities of canonical organic functional groups **2012**
Journal of Organic Chemistry

Methyl anion affinities of the canonical organic functional groups **2012**
Journal of Organic Chemistry

Tourbillon: a Physically Plausible Neural Architecture **2012**

Deep Learning to Reconstruct Gas Skymaps for Dark Matter Detection **2020**
Neural Information Processing Systems 2020, Machine Learning and the Physical Sciences Workshop

Continuous Representation of Molecules Using Graph Variational Autoencoder **2019**
Journal name

Deep learning for chemical reaction prediction **2018**
Molecular System Design and Engineering

IN PREPARATION AND UNDER REIVEW PUBLICATIONS

PMechDB: A Public Database of Elementary Polar Reaction Steps **2023**
Journal of Chemical Information and Modeling

AI for Interpretable Chemistry: Predicting Radical Mechanistic Pathways via Contrastive Learning **2023**
Neural Information Processing Systems

A Deep Learning System for Interpretable Radical Reaction Prediction **2023**
Journal of Chemical Information and Modeling

Towards Biologically Plausible Learning by Stacking Circular Autoencoders **2023**
Neural Information Processing Systems

ABSTRACT OF THE DISSERTATION

Machine Intelligence for Chemistry: From Deep Learning Architectures to Open Data

By

Mohammadamin Tavakoli

Doctor of Philosophy in Computer Science

University of California, Irvine, 2023

Pierre Baldi, Chair

Achieving human-expert performance in predicting the outcomes of chemical reactions is a major open challenge in AI and chemistry. A solution to this challenge would have significant practical applications in areas ranging from drug design to atmospheric chemistry. However, in order to address this challenge, many issues need to be overcome including the lack of open data, the combinatorial and physical complexity of chemical reactions, and the need for interpretable solutions that illuminate the underlying reaction mechanisms. We will describe three projects aimed at addressing these challenges including the development and deployment of public databases of chemical reaction steps, and the development and training of deep graph neural network and transformer architectures to predict reaction outcomes in interpretable ways.

Chapter 1

Introduction

Attaining chemist-level performance in predicting the outcomes of chemical reactions remains an unresolved challenge in computational chemistry. A solution to this challenge would have applications in many areas of chemistry both in the laboratory, for instance in synthesis, retrosynthesis, and drug design; and in nature, for instance: in biology and atmospheric chemistry. Here we focus on the prediction of reactions at the level of mechanistic steps using machine learning approaches.

Currently, three main classes of approaches are being developed to address reaction prediction in general: (1) Quantum Mechanics (QM) simulations; (2) template-based predictions; and (3) template-free predictions. These approaches are not mutually exclusive and can be combined in different ways, as in [3].

Quantum Mechanics Simulations: Methods based on quantum mechanics and the simulation of chemical reactions at the atomic level provide the most accurate predictions of reaction outcomes. However, these fine-grained simulations are computationally expensive and may require human intervention [4, 5]. Quantum mechanics (QM) simulations are particularly sensitive and have operational limitations that restrict their applicability to a narrow range of chemical systems. Consequently, in current environments, QM simulations are not suitable for making high-throughput predictions of reaction outcomes.

Template-based methods: These types of methods often require a set of pre-defined chemistry rules that can be applied to a set of reactant molecules to predict chemical reaction outcomes. Thus these methods are also called rule-based methods. These rules, which are often referred to as "chemical templates" can be extracted manually by human experts [6, 7] or can be computationally inferred [8] from a data set of chemical reactions [9, 10, 11]. In both cases, the templates are limited to either the knowledge of the human extractor or the chemistry of the given data set. The rule-based system can produce rapid predictions and can be used in "brewing" experiments where multiple reactants are iteratively reacted with each other and their products [12, 13]. High-throughput template-based predictions can be applied to retrosynthetic experiments using programs such as Synthia [14, 15]. Machine learning methods have been used to help select templates [9, 16]. One important issue is that there is no obvious gradient in the discrete space of symbolic learning rules and thus efficient machine learning methods require finding a continuous embedding, leading to template-free methods. Another issue often seen with template-based methods is that they have limited generalization capabilities. The selection of general templates typically disregards local interactions between atoms in a small neighborhood, while the selection of specific templates may neglect the effects of reaction context, such as the presence of particular reagents.

Template-free methods: In contrast, template-free methods can achieve broad generalization to a wide range of chemical reactions while offering fast predictions. These methods, are purely based on machine learning predictive techniques and require large data sets for training and development [17, 18, 19]. This is a significant issue by itself since there is no comprehensive publicly available data set covering all, or even most, known chemical reactions [20]. Nevertheless, various partial data sets have become available enabling at least a partial development of these methods. Different methods can be developed for different representations of the reactions, which are rooted in the representation of the reactant and product molecules. There are at least four distinct ways of representing molecules: labeled graphs, text strings (e.g. SMILES strings leading to SMIRKS reaction representations), fingerprints, and sets of atoms with their 3D coordinates). Here, as in most of the literature we focus on machine learning methods based on graphs or text.

Graph-based methods represent each molecule as a graph with nodes associated with atoms, and edges associated with bonds[21, 22, 23, 24]. A chemical reaction is described by the graphs associated with the reactant molecules on one side, and the product molecules on the other. These labeled graphs can be processed using some form of recursive, or graphical, neural networks using an inner or outer (convolutional) approach [25, 26, 27, 28, 29] . In these methods, neural networks are used iteratively to pass information between graph neighborhoods and compute vectorial representations that are increasingly more abstract and encompassing. Such contextual representations can be exploited to generate graphs of product molecules using different approaches. For instance, [30] leverages these representations to identify graph edits that lead to the product molecules, whereas [31] employs a generative model to yield the 2D graph of the products.

Text-based methods typically utilize the simplified molecular-input line-entry system (SMILES) and its extension to reactions (SMIRKS) [32]. Various kinds of neural network architectures, ranging from recurrent/recursive to transformer architectures, can be used to process such variable-length text strings, similarly to what is done for natural language processing (NLP) problems [33, 34, 35]. Reaction prediction can be cast as a sentence-to-sentence or "translation problem" from the reactant sentence to the product sentence [36, 37]. Typically an encoder module is used to encode the discrete string associated with the reactant molecules into a continuous vector representation, and a decoder module is used to decode the continuous vector representation into the discrete string associated with the product molecules

[38, 35]. The encoding and decoding processes are implemented using deep recurrent neural network architectures that may include long short-term memory (LSTM) [39] units and Transformers [40]. The decoder module can also leverage techniques such as beam search and attention mechanisms to generate the most accurate SMILES string [35, 38].

There is no foundational advantage for any of these machine learning approaches over the other ones, although preliminary evidence [38] together with the current success enjoyed by large language models suggest that text-based methods may be particularly effective. This is in spite of the facts that: (1) SMIRKS do not contain all the information needed to accurately model molecules and reactions [41]; (2) the same reaction (resp. molecule) can be represented by multiple SMIRKS (resp. SMILES), including those associated with any permutation of the molecules within the reactants or within the products[42]. Finally, text-based methods, as well as other machine learning methods, require large training sets, which is a major challenge in reaction chemistry. Compensatory methods, such as regularization and data augmentation, are being used but are often not enough.

The majority of recently created models, independently of the way they represent reactions, are being designed and trained using the chemical transformation dataset derived from the US Patent Office (USPTO) [43] and a few other minor datasets that were introduced in several studies [28, 44, 45]. All these data sets have significant limitations, including: lack of chemistry coverage, lack of precise and complete atom mapping, lack of balance between reactants and products, and lack of elementary reaction step information. The USPTO dataset of chemical reactions, for example, portrays chemical reactions restrictively as overall transformations, the majority of which result in a single primary product (and are thus unbalanced). This dataset provides little information about the underlying mechanisms, critical intermediates, and side products. Additionally, it is challenging to extract information about radical reactions, which appear to be underrepresented in this data set.

Utilizing this limited source of data to train data-driven models results in models that may be chemically limited or biased, predict unbalanced reactions, and lack information about intermediate byproducts. Additionally, these models provide no interpretations of the underlying chemistry that leads to the predicted products. Although there has been some work (e.g. [35]) using machine learning interpretability methods to highlight the importance of certain portions of the reactants, this does not provide any information on the mechanistic interactions underlying the reactions. Lastly, due to the limited available training data, there is no widely used reaction predictor for radical reactions. Radical reactions play a significant role, for instance in synthetic pathway planning and biological and atmospheric chemistry, and they often proceed through a complex series of chemical steps and highly branched mechanistic pathways. Thus developing an accurate radical reaction predictor free of some of the shortcomings mentioned above is important.

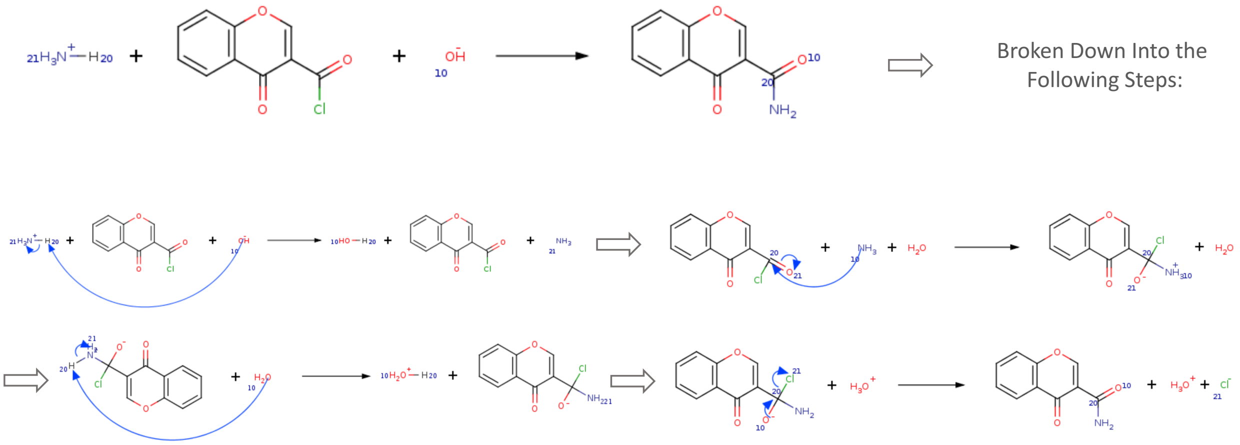


Figure 1.1: The reaction at the top is an overall, unbalanced, transformation from the USPTO data set. It can be broken down into four mechanistic steps with arrow-pushing mechanisms. This provides chemical interpretability for each step, as well as for the overall pathway while maintaining full balance at each step.

Chapter 2

RMechDB: A Public Database of Elementary Radical Reaction Steps

2.1 Abstract

We introduce RMechDB, an open-access platform for aggregating, curating, and distributing reliable data about elementary radical reaction steps for computational radical reaction modeling and prediction. RMechDB contains over 5,300 elementary radical reaction steps, each with a single transition state at or around room temperature. These elementary step reactions are manually-curated plausible arrow-pushing steps for organic radical reactions. The steps were taken from a variety of sources. Over 2,000 mechanistic steps were extracted from textbooks and or constructed from research publications. Another 3,000 were taken from gas-phase atmospheric reactions of isoprene and other organic molecules on the MCM (Master Chemical Mechanism) website. Reactions are encoded in SMIRKS format with accurate atom mapping and annotations for arrow-pushing mechanisms. At its core, RMechDB consists of a database schema with an online interactive search interface and a

request portal for downloading the raw form of elementary step reactions with their metadata. It also offers an interface for submitting new reactions to RMechDB and expanding the data set through community contributions. Although there are several applications for RMechDB, it is primarily designed as a central platform of radical elementary steps with a unified and structured representation. We believe that this open access to this data and platform enables the extension of data-driven models for chemical reaction predictions and other chemoinformatics predictive tasks.

2.2 Introduction

A free radical is a chemical compound (e.g. atom, molecule) with at least one half-occupied orbital. The presence of the half-occupied orbitals makes a radical compound highly reactive. Because of this high reactivity, free radicals have the potential to both serve as powerful chemical tools and be extremely harmful contaminants. Chemical reactions involving a free radical are radical reactions that are an essential part of synthetic, biochemical, atmospheric, and plasma chemistry [46, 47, 48]. For instance, the climate crisis has dramatically altered fire activity worldwide. Wildland fires are increasing in frequency, duration, intensity, and size. The chemistry of flames is dominated by radical reactions and the chemical composition of fire smoke changes during atmospheric transport. This so-called "aging" of smoke is poorly understood but known to be largely driven by free radical processes [49, 50, 46]. As another example from the pharmaceutical industry, the composition of drug formulations changes gradually upon storage. As a result, all drug companies are required to study those changes through forced degradation studies under several conditions, including photochemical and oxidative conditions, which mostly involve radical reactions [51, 52]. Thus, it is of great importance to study the chemistry of radical reactions and their outcomes.

During the past few years, data-driven methods such as deep learning have provided new powerful tools for addressing chemoinformatics problems [53, 23, 54, 42, 23, 24, 4, 55]. Due to important applications ranging from automated drug discovery to computer-aided synthetic chemistry, there has been an increasing interest in developing deep learning models to predict the outcome of chemical reactions [56, 30, 35, 57, 58]. While the deep learning models have been evolving in sophistication and complexity, a major stumbling block has remained the lack of comprehensive, standard, and public, reaction data [20]. The majority of recently developed models are being trained using the data set of chemical transformations from the US Patent office [59], as well as a few other smaller data sets [54, 44, 45]. These data sets are spread across different platforms without unified and structured representations and metadata. Additionally, they suffer from significant limitations in terms of overall size, chemistry coverage, and balance, and lack of meta-data, atom mapping, a reactant or product balance, and elementary reaction step information. For instance, the USPTO data set of chemical reactions restrictively represents chemical reactions in the form of overall transformations, most of which lead to one single major product. It contains little information about underlying mechanisms and about key intermediates and side products. Furthermore, radical reactions are hard to extract and appear to be underrepresented. On the other hand, radical reactions often proceed through a complex series of chemical steps and highly branched mechanistic pathways. Developing an accurate machine learning model for predictive tasks on radical reactions (e.g. predicting the outcome of radical reactions) requires a training data set of purely radical reactions with information about the mechanistic pathways and intermediate products. To overcome the above limitations, and provide a source of data for radical reactions with their unique natural characteristics, we developed RMechDB as a central platform for aggregating, curating, and distributing elementary step radical reactions. RMechDB is designed as an extendable database schema, capable of hosting huge sources of radical reactions in the form of elementary steps. RMechDB is publicly available in the form of an online web server with interactive interfaces where users can search, download, and upload elementary step radical reactions. The initial version of the

RMechDB data set consists of over 5300 manually curated radical reactions and is accessible through the DeepRXN website at `https://deeprxn.ics.uci.edu/rmechdb`.

2.3 Mechanistic Pathways vs Overall Transformations

The term reaction can be ambiguous and is most commonly used to describe either: 1) a chemical transformation with reactants, products, chemical conditions, and yields; or 2) a single step in an arrow-pushing mechanistic pathway. Therefore, in this work, instead of using the vague term “reaction”, we use the more specific terms of transformation and elementary step to refer to the definitions above respectively. Every mechanistic pathway can be decomposed into a series of discrete elementary steps, each with a single transition state [60, 61]. In several aspects, it is advantageous to show every step in a mechanistic pathway. First, when all the steps in a pathway are elementary, there is no chance of missing key intermediates that give rise to competing pathways during chemical transformation. This becomes extremely important with the presence of free radicals as radical transformations often proceed through a complex series of chemical steps and highly branched mechanistic pathways For example, when the transformation of ISOPAO to C524O2 is depicted as a one-step process, it misses the potential for the allyl radical intermediate to form an isomeric peroxy radical and downstream products (Figure 2.1). The second advantage to mechanistic pathways based on elementary reaction steps is that they can be described using curved half-arrows that correspond to the interaction of singly occupied molecular orbitals with a HOMO and/or LUMO [62]. The curly arrows, also known as electron flow specifications or arrow-pushing mechanisms, are depicting the interaction between molecular orbitals. This representation of elementary steps is highly informative and, when elementary steps are chained together, an interpretation of the corresponding transformation can readily be derived. This becomes even more important specifically for deep learning approaches to reaction product prediction for at least three reasons. First, the prediction of mechanistic pathways leads to predictions that are interpretable. Interpretability is an essential consideration in machine learning, especially for so-called "black-box" approaches such as deep learning [63, 64]. Second, when machine learning models operate at the level of elementary steps, the balance between reactants and products is always preserved together with the underlying atom mapping. Maintaining the balance through a chain of reactions can be extremely important in the study of retrosynthesis pathways. And third, by considering the pathways, all intermediary and final products can be accounted for, which is an important consideration in synthetic chemistry applications.

Given the crucial advantages of representing chemical reactions in the form of mechanistic pathways, it is highly beneficial to synthesize a data set of elementary radical steps. Such data sets can facilitate the training and development of deep learning models that are able to automate complex predictive tasks in radical chemistry.

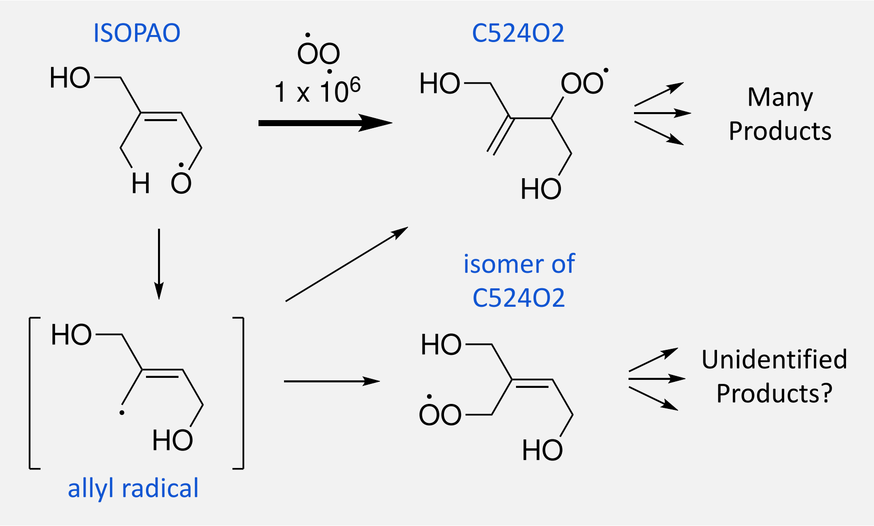


Figure 2.1: Missing steps and intermediates prevent identification of products. The formation of an allyl radical was not depicted for the transformation of ISOPAO to C524O2 in the MCM. It is not clear why the missing allyl radical intermediate would not also generate an isomer of C524O2 and account for more downstream products.

2.4 Approaches to Chemical Reaction Modeling and Predictions

An open-source, publicly available database of pedagogical elementary reaction steps will facilitate training and development of tools for automating chemoinformatics tasks such as the prediction of reaction mechanisms. There are two common approaches to the prediction of step-wise mechanisms of organic transformations using databases of elementary reaction steps. The quantitative approach uses a database of kinetic and thermodynamic parameters to accurately predict the products of the reactions and the pathways by which they form. This approach, as it is used in [65, 66] is not restricted to elementary reaction mechanisms, but it does require kinetic parameters. The approach is best applied to cases where the product structures are known but the abundances are not known. The qualitative approach such as [67, 54, 56] is to use a database of diverse plausible (fast at or below 100 °C) mechanistic steps, to match chemical structures (and mechanistic pathways) to mysterious, unknown, or not structurally characterized analytes in readily available spectra or chromatograms. This approach is best applied when the abundance is known, but the chemical structure is unknown. The chemical structure can provide powerful insight into biological effects, phase partitioning, and reactivity under changing reaction conditions. Public databases of mechanistic steps will empower the use of machine learning to create tools that assign chemical structures and mechanisms to products of environmental, synthetic, and environmental transformations of organic compounds.

2.5 Existing Data Sets of Elementary Reaction Steps

There are several large commercial databases of organic transformations such as REAXYS, SciFinder, and very few open-access databases such as the Open Reaction Database (ORD)

[68]. Those databases are composed of recipes that describe reactants, conditions, yields, and a list of products that rarely sums to 100%. The proprietary REAXYS database currently has over 57 million transformations. The SciFinder Scholar database has over 126 million transformations, which includes sequential reactions. Organic transformations were mined from US Patents from 1976-2016 and are publicly available. The growing ORD already gathers about 2 million chemical transformations from other available sources[68]. These databases of chemical transformations allow synthetic organic chemists or systems trained with machine learning, [69] to plan out synthetic routes composed of sequential laboratory experiments, but the data don't reveal the underlying mechanisms of any individual transformations. Databases of transformations are not new, and neither is the application of AI to the planning of synthetic routes. Why is there no database of elementary arrow-pushing reaction steps? Sadly, when curved arrows were first introduced in 1922,[70][71] the connection between curved arrows, frontier orbitals, and transition states was not recognized, so there was no incentive to apply them solely to elementary mechanistic steps. As a result, curved arrow mechanisms and half-arrow radical mechanisms have been used inconsistently, throughout the organic chemistry literature and are rendered in graphical forms that are not easily recoverable through data mining. Reaction Mechanism Generator (RMG) supports the only existing database of elementary mechanistic reaction steps. RMG predicts mechanistic pathways through a quantitative approach, using thermochemical and kinetic parameters to model species concentrations and rates for each step[65]. RMG is supported by a searchable database, consisting of 98 families of reaction types[65]. Almost half (40/98) of the reaction families in the current RMG database involve radicals. About a fourth of the reaction families supported by RMG do not correspond to elementary reaction steps at or around room temperature (e.g., unimolecular keto-enol tautomerization). Most of the mechanistic steps and kinetic data were developed to support high-temperature processes up to 2000 K and many of the steps would be implausibly slow at room temperature. For example, the kinetic parameters for homolysis of a CH_3 group

from isoprene would proceed with a half-life of over 10^{42} years. Many of the steps that proceed through a single transition state at high temperatures (e.g., over 1500 K) would involve more than one mechanistic step at room temperature.[65] For example, at room temperature, the addition of HO• to the double bond of alpha-pinene should not be concerned with ring opening. The requirement for accurate thermochemical and kinetic creates a major hurdle for applications involving complex organic structures. Additionally, RMG development has so far been focused on processes involving simple reactants with just a single organic functional group and up to one heteroatom: CH_4, CH_3CH_3, $CH_3CH_2CH_3$, exo-tetrahydro dicyclopentadiene, $C_{10}H_{16}$, CH_3OCH_3, $CH_3(CH_2)_3OH$, $CH_3(CH_2)_5CH_3$, $((CH_3)_2CH)_2CO$, $CH{=}CHCH{=}CHCH_2CH_3$, $HCC(CH_2)_4CCH$, $C_6H_5(CH_2)_5CH_3$, $(CH_3)_2CHCH_2OH$, $CH_3(CH_2)_4CH_3$, $H_2NCH_2CH_3$, and $((CH_3)_3C)_2S$, C_6H_5OH. A few other examples of data sources containing elementary steps are NIST Chemical Kinetics Database [72], Mechanism and Catalytic Site Atlas (M-CSA) [73], and Master Chemical Mechanism [66, 74, 75, 76, 77, 78, 79], all of which suffer from unorganized, unstructured form of elementary steps with extremely limited online support.

2.6 RMechDB: Underlying Data Set

2.6.1 A Data Set of PLAUSIBLE Radical Elementary Steps

Organic transformations in databases such as REAXYS, SciFinder, and ORD are easily validated because published products are rigorously characterized using convenient spectroscopic techniques such as mass spectrometry, NMR, and IR. In contrast, mechanistic steps with one transition state are not easily validated. Experimental proof of a mechanistic step usually requires electronic structure calculations and/or laborious experimental tools such as chemical kinetics, isotopic labeling, crossover experiments, etc. It is often quoted that one

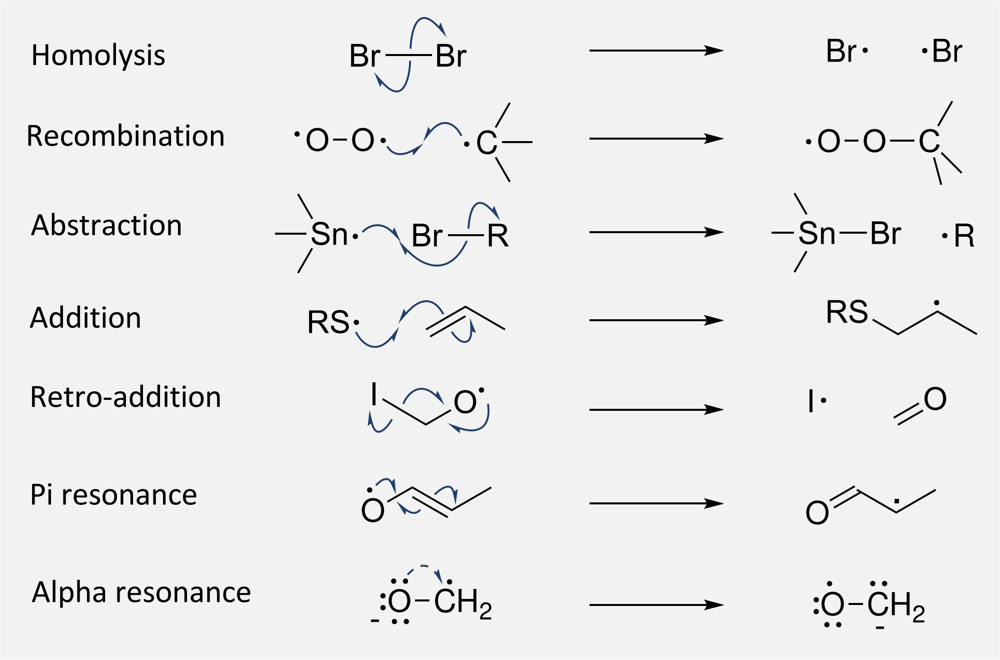


Figure 2.2: Seven different categories of mechanistic steps involving radicals.

can never prove a mechanism, but only dis-prove the plausible alternatives[80]. We set out to construct a data set of plausible elementary reaction steps, which are useful to chemists in constructing mechanistic pathways and predicting byproducts of organic reactions. Plausibility is subjective. For RMechDB, we define an elementary mechanistic step as plausible if a half-life of a day or less is expected at room temperature under the conditions cited. If more than one pathway has been postulated in the literature, it is expedient to include steps from both potential pathways in the data set until the discrepancy is resolved. That way, any pathway proposed using the data will reflect the ambiguity in the body of literature. In theory, the plausibility of any elementary reaction step can ultimately be validated using electronic structure calculations.

2.6.2 Composition of the RMechDB Data Set

The initial data set in RMechDB consists of over 5,300 pedagogically chosen elementary radical mechanistic steps based on published transformations. The majority of the pub-

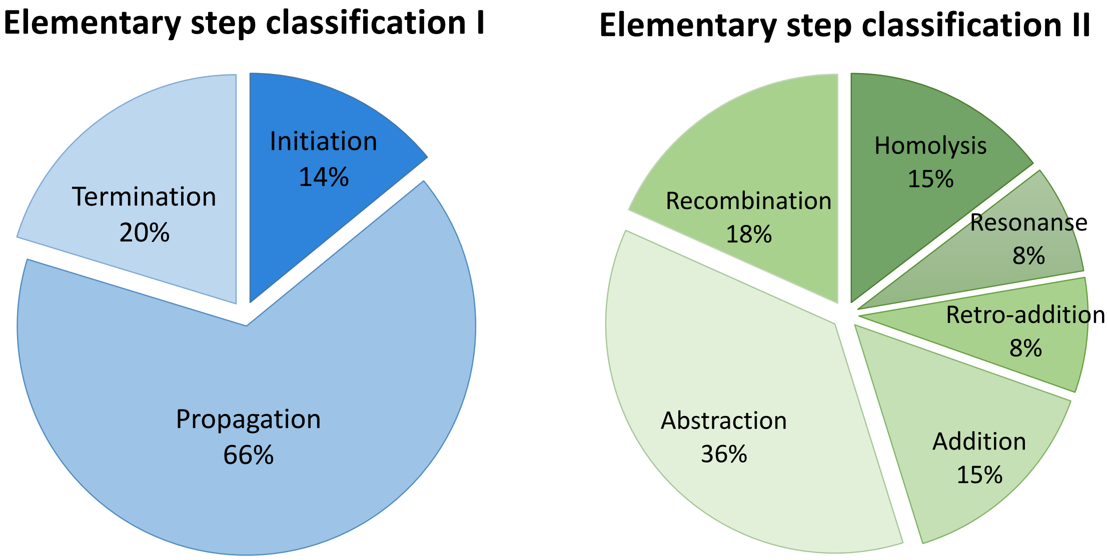


Figure 2.3: The distribution of the different classes of reaction in the current version of the RMechDB data set.

lished mechanistic steps had to be further decomposed into elementary reaction steps with individual transition states. Over 880 steps were taken from eight introductory[81, 82, 83, 84, 85, 86, 87, 88] organic chemistry textbooks, advanced organic chemistry books[89][90], and an atmospheric chemistry textbook[91]. Over 800 reactions were taken from the primary research literature including mechanisms for common synthetic transformations (atom transfer, tin chemistry, radical cyclizations), autoxidation, atmospheric reactions, and explosives. The literature mechanisms also included steps leading to 14 common industrial polymers: ethylene, propylene, butadiene, chloroprene, isoprene, acrylamide, acrylic acid, methyl acrylate, ethyl acrylate, butyl acrylate, methyl methacrylate, acrylonitrile, styrene, p-methylstyrene, vinyl chloride, vinyl fluoride, tetrafluoroethylene, chlorotrifluoroethylene, vinylidene fluoride, vinyl acetate, N-vinylpyrrolidinone. The conditions for polymerization, often including more than one type of initiator, were taken from the research literature and are not necessarily the proprietary initiators and conditions used for industrial synthesis. The data from textbooks and research literature are considered the core of the RMechDB database.

The core data set has been augmented with a large number of mechanistic steps related to the atmospheric oxidation of organic molecules. We refer to this dataset as specific steps. A large number (847) of specific steps were taken from a comprehensive review of atmospheric isoprene oxidation that traced the fate of each individual carbon atom detailing the highly branched pathways from reaction with HO•, O_2, NO, Cl• and other species[92]. For simplicity, we focus on the daytime atmospheric chemistry of isoprene at atmospherically relevant conditions (average atmospheric T=278 K), neglecting elementary steps involving NO_3, which is a dominant nighttime oxidant. Most of the elementary steps were inferred from composite transformations. About 3,000 mechanistic steps were coded from the first two stages of the major oxidation pathways in the Master Chemical Mechanism (MCM)[66]. The MCM contains mechanisms for atmospheric oxidation of 143 volatile organic compounds initiated by both HO• and NO_3, including reactions of isoprene. Steps more than ten times slower than the fastest process (with the same reactants) were also excluded. Steps second-order in reactive intermediates were excluded on the assumption that they would not slow under typical conditions. For both the Wennberg and MCM steps, transformations initiated by pericyclic [3+2] cycloaddition of O_3 with alkenes were excluded from this initial data set, but depicting the cycloaddition as a diradical process could be an expedient[93]. Photolysis steps were also excluded. Any steps left out of this initial data set can be introduced in the future.

The individual mechanistic steps are also labeled using two distinct classification schemes: (1) Three-class classification, where each elementary step falls into one of the three possible phases of a radical chain reaction: initiation, propagation, and termination; and (2) The more detailed seven-class classification, where an elementary step reaction falls into one of seven different categories: homolysis, recombination, abstraction, addition to pi bonds, retro-addition to pi bonds, pi (e.g., allylic) and alpha lone pair resonance (e.g., ketyls). All seven classes are depicted in Figure 2.2. In RMechDB, resonance is represented as a mechanistic step, even though there is no transition state. Homolysis and recombination are mechanistic reverses of each other, like addition and retro-addition. Alpha resonance is represented with a single curved half-arrow, but it is acknowledged that the half-arrow falsely implies the formation of a partial double bond. The steps in radical chain mechanisms are often classified as initiation, propagation, or termination steps, but many transformations involving radicals do not involve chain mechanisms. Homolysis is a typical chain initiation step. Atom abstraction, addition, retro-addition, and resonance are typical chain propagation steps. Recombination is a typical chain termination step. Within the RMechDB data set, we try to emulate the natural distribution of radical reactions based on the classifications described above. Figure 2.3 represents the distribution of different classes of radical reactions in the RMechDb data set.

2.6.3 Structure of the Data

The initial version of RMechDB contains over 5300 pedagogically chosen elementary radical step reactions based on published transformations. Steps are categorized into two major types: (1) Core elementary steps, extracted and curated from textbooks and the scientific literature, capturing generic radical mechanisms; and (2) Specific elementary steps, curated from multiple sources, capturing mechanisms associated with atmospheric chemistry. Given

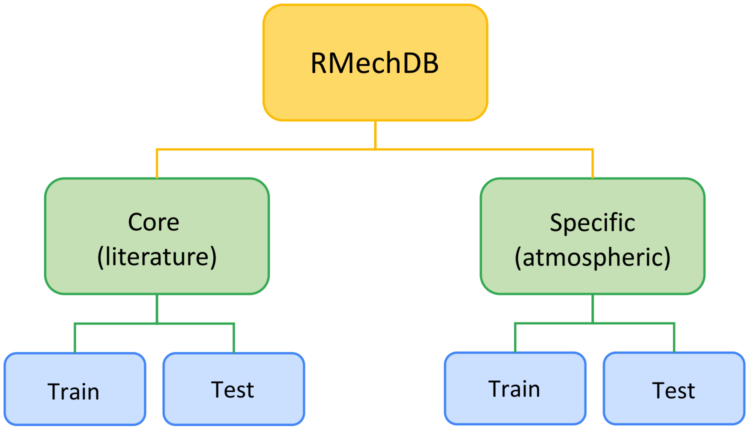


Figure 2.4: The general format of the RMechDB data set.

that one of the main goals for RMechDB is to provide a source of data for machine learning models, each type is carefully split into a canonical train and test data (Figure 2.4).

While machine-learning users can of course split the data in any way they want, having a canonical train/test data split facilitates standardized training and evaluation workflows, as well as the comparison of performance across different research groups. This canonical split is manually curated to ensure balance and coverage consistency between the train and test data. Specifically, we use two criteria: **balanced categorical distribution** and **consistent chemistry coverage**. To maintain the balance in categorical distribution, we ensure that the distribution of the seven categories described above (Figure 2.2) is approximately the same in the train and test data. To maintain consistent chemistry coverage, for any mechanistic steps in the train data, we ensure that there is at least one mechanistic step with similar reacting functional groups in the test data. As a result, using this presented train and test split leads to a more interpretable evaluation of the generalization capabilities of predictive models.

Each entry of RMechDB consists of elementary reaction steps in the SMIRKS format including atom mapping for atoms that are a part of the transformation. Each SMIRKS is associated with its electron flow specification representing the atom indices on the curved half-arrows (Figure 2.5). Additionally, each elementary step has been decorated with the following properties: (1) The initial condition of the reaction which falls into the room temperature (298 K), heat, or light conditions; (2) The reaction class I which is the type of the radical elementary step; (3) The reaction class II which is the type of the radical elementary step based on a more fine-grained categorization; and (4) The scholarly source of the elementary step. The addition of more important properties such as phase, solvent, wavelength, and enthalpy is left for future work.

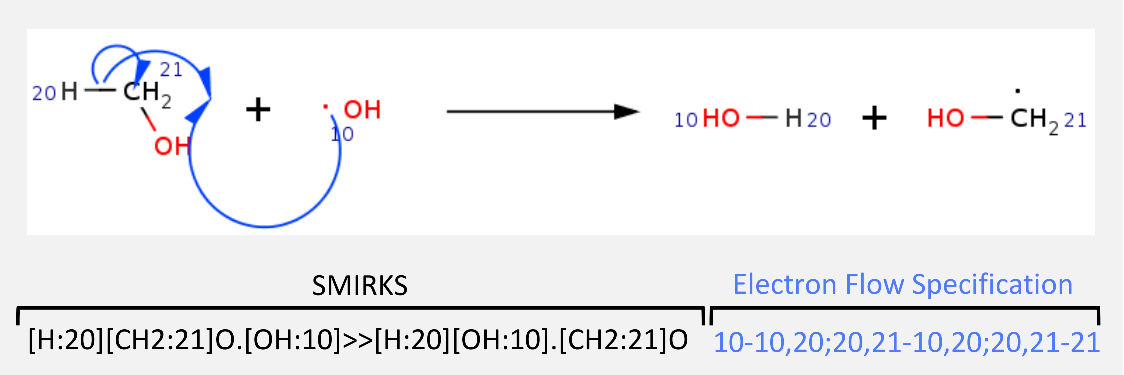

20 H
CH2 21
OH
OH 10
10 HO — H 20
HO — CH2 21
SMIRKS
Electron Flow Specification
[H:20][CH2:21]O.[OH:10]>>[H:20][OH:10].[CH2:21]O
10-10,20;20,21-10,20;20,21-21

Figure 2.5: RMechDB format for depicting reactions and arrow pushing mechanisms. The atom participating in the reaction are mapped on both sides of the reaction.

2.7 RMechDB: The Core Database

2.7.1 Standard Elementary Step Model

In addition to serving as a central source of reaction data for machine learning models, RMechDB is designed to be extendable by community contribution. To maintain that, it is crucial to use a standard and unified representation of elementary step reactions. This standard representation would enable consistent data sharing, model reproduction, and scalable expansion. We model the elementary step reaction using the reaction model introduced in [67, 61]. In this model– so-called "elementary step model", the transition state is modeled as the movement of one single electron from one half-occupied molecular orbital (MO) to another. We use the atom labels in the arrow code of the elementary step to track the movement of the electron. Lone pairs or π-bonds adjacent to π-bond MOs can be chained to allow longer-range resonance rearrangement. In this model, each MO is associated with its main atom. As a result, each radical elementary step has two reactive atoms and two reactive MOs. We use the elementary step model to construct and populate the database schema described in the next section.

2.7.2 Database Schema

The database is implemented using the PostgreSQL [94] database management system [95], to store, query, and retrieve reaction instances both efficiently and safely. We use OpenEye Scientific Software [96] toolkits OEChem [97], OEDepict [98], and GraphSim [99] for chemoinformatics processing and depiction. In addition, we use Chemaxon Marvin [100] for displaying and characterizing chemical structures, substructures, and steps with their corresponding arrow-pushing mechanisms.

The RMechDB database schema comprises three fundamental models: (1) `Reaction`, (2) `Molecule`, and (3) `Atom` as shown in Figure 2.6. The inter- and intra-integration of these three models allow for fast and efficient reaction search and retrieval. As the naming suggests, each elementary step is stored as an instance of the `Reaction` model which comes with several descriptive fields. These fields are designed to uniquely represent an elementary step reaction and all the available metadata associated with it. Here we list the main fields of the `Reaction` model.

1. **Reaction ID:** Each reaction is associated with a unique ID number.

2. **Canonicalized atom mapped SMILES of the reactants:** The SMILES string of the reactants molecules, with integer labels for atoms that are participating in the reaction. We use a labeling convention where the labels of the participating atoms on the nucleophile part start from 10 and increment by one per atom and the labels of the participating atoms on the electrophile part start from 20 and increment by one per atom.

3. **Canonicalized SMILES of the products:** The unique SMILES representation of the product molecules generated from the reactive reactants.

4. **Canonicalized arrow codes:** The standard codes for arrow pushing mechanisms

contain the integer labels of the participating atoms on the reactants side. The standard arrow codes begin from the integer label (starting at 10) on the nucleophilic group.

5. **Spectator molecules:** The unique SMILES representation of the molecules that are present in the reaction but not participating in the electron transfer.

6. **Reactive atom I:** The SMILES string of the molecule containing the first reactive atom (based on the RMechDB orbital model) whose label is 1.

7. **Reactive atom II:** The SMILES string of the molecule containing the second reactive atom whose label is 1.

8. **Step type:** Core or specific step (Figure 2.4).

9. **Initial heat or energy:** The initial condition of the step which can be independent of external energy – represented as blank, "heat", or "light".

10. **Step classification I:** The class of the step according to the 3-class classification into initiation, propagation, and termination.

11. **Step classification II:** The class of the step according to the 7-class classification into homolysis, recombination, addition, retro-addition, abstraction, alpha- resonance, and pi-resonance shown in Figure 2.2.

Given the fields above associated with the `Reaction` model, an instance of the `Reaction` model in RMechDB can be uniquely retrieved from the database using either the **Reaction ID** or the combined properties 2-5 as the key.

The `Molecule` model has three fields corresponding to the unique molecule ID, canonicalized SMILES string of the molecule, and the OEChem MolBase object [97]. An instance of the `Molecule` model has a many-to-many relation with the reactant molecules, product molecules, and spectator molecules fields of the `Reaction` model.

The `Atom` model has three fields corresponding to the unique ID, canonicalized atom mapped SMILES string of the parent molecule, and the OEChem AtomBase object [97]. An instance of the `Atom` model has a many-to-many relation with the reactive atom I and reactive atom II fields of the `Reaction` model.

The schema with the fields described above is designed not only to provide efficient storage and retrieval but also to enable the automated population of the fields for new steps that are contributed to RMechDB by the community as described in the section on Uploading New Data.

2.8 RMechDB: Web Server

The web server of the RMechDB includes three interfaces for: (1) Searching the data; (2) Downloading the data; and (3) Uploading new data.

2.8.1 Searching the Data

RMechDB provides an interactive search interface available at `https://deeprxn.ics.uci.edu/rmechdb/rsearch` where users can search through the database using a variety of methods. At the highest level, the interface allows for reaction search and compound search.

Reaction Search

1. **Exact search:** Using the exact search method, the user inputs the query in the form of the SMIRKS of an elementary step containing reactants and products (no arrow code needed). Then the system finds and displays all the elementary steps with the same reactants and products as in the query reaction but with additional molecules

involved as reagents or spectators.

2. **Search by reactants:** Using the search by reactant (or by reactants), the user inputs the query in the form of a set of molecules, separated by ".". Upon hitting the search button, the system finds and displays all the elementary steps with reactants containing the query molecules. This search is useful when the user does not know the exact reaction and how molecular orbitals might react.

3. **Search by products:** Similar to the search by reactants, using the search by-products (or by-products), the user inputs the query in the form of a set of molecules, separated by ".". Upon hitting the search button, the system finds and displays all the elementary steps with products containing the query molecules.

4. **Similarity search:** Using the similarity search method, the user again inputs the query in the form of the SMIRKS of an elementary step containing reactants and products (no arrow code needed). Then the user specifies a similarity metric and the number of similar reactions (N) to be retrieved under this query. Upon hitting the search button, N elementary steps sorted from the most similar to the least similar to the input query are displayed.

The current version of RMechDB is equipped with the following similarity metrics computed on various representations of the elementary steps:

1. The Tanimoto, dice, and cosine distance between the binary Extended Connectivity Fingerprints (ECFP) of the elementary steps.

2. The Euclidean distance between the embedding of the elementary steps derived using a pretrained transformer architecture, trained on the SMIRKS of the USPTO data set [35, 59].

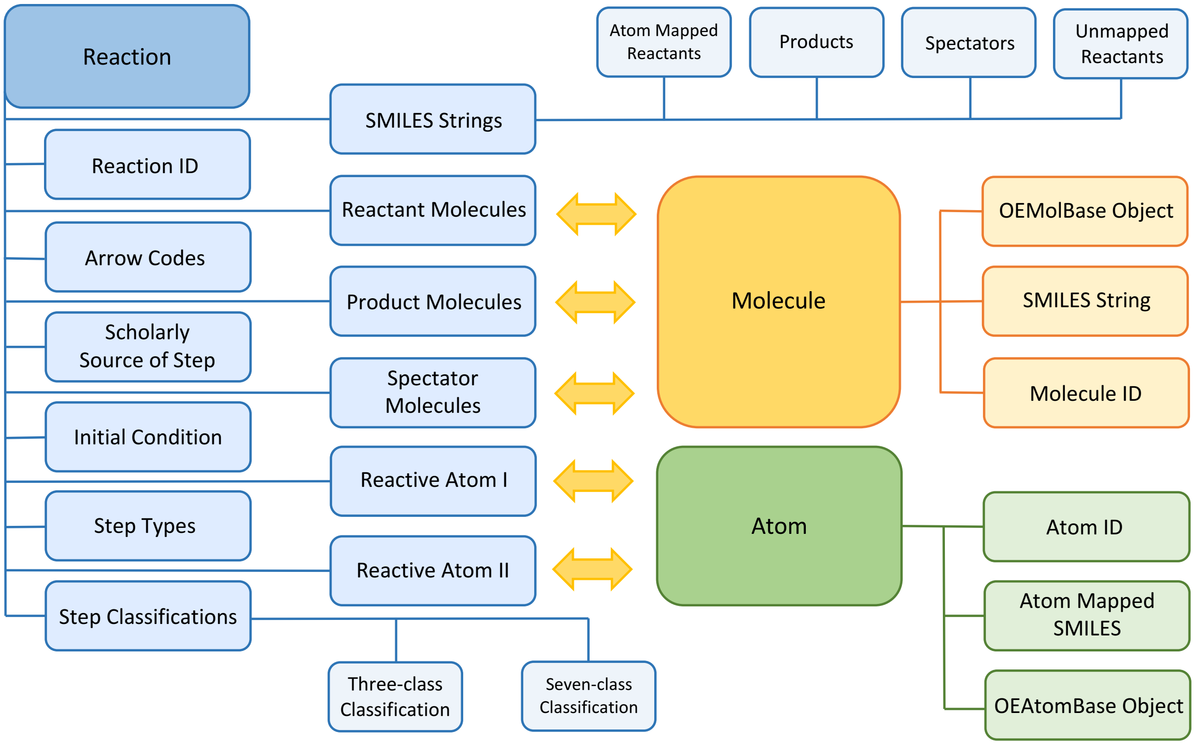


Figure 2.6: The three fundamental models of the RMechDB database and how they integrate. The yellow arrows show the `many-to-many` relations.

3. The Euclidean distance between the embedding of the elementary steps derived using the pretrained RxnHypergraph method [42].

Compound Search

In addition to search capabilities based on elementary steps, RMechDB provides search capabilities based on smaller chemical entities as follows:

1. **Molecule search:** In this search, the user inputs the SMILES string of the desired molecule. After testing the validity of the input SMILES, RMechDB displays those elementary steps in the database that contains the desired molecule in the reactant or product side of the elementary step.

2. **Reactive atom (molecular orbital) search:** In this search, the user inputs the atom-mapped SMILES string of the molecule where the reactive atom is labeled using an integer between 1 and 9, while the other atoms are not labeled. After testing the validity of the input SMILES with the labeled atom, RMechDB displays all the elementary steps in the database where the labeled atom is acting as one of the two main reactive atoms in the elementary step.

3. **Substructure search:** In this search, the user inputs the SMARTS of a chemically valid substructure. RMechDB displays all the elementary steps in the database with molecule(s) containing the input substructure. The molecule that contains the input substructure can be in the reactant or product side of the elementary step.

In addition, the results of each search can also be filtered using the following properties: (1) the type of the elementary steps (core or atmospheric); and (2) the category of the elementary step based on either of the two categorization schemes described in the Composition of the RMechDB Data Section.

The result of each search will be displayed as a table containing the depiction of the filtered reactions along with their reactive atom-mapped SMIRKS, arrow codes, masses of the products, and the initial conditions. The search query inserted by the users will also be displayed in a separate box.

2.8.2 Downloading the Data

The data set of the chemical reactions in RMechDB is available for download at `https://deeprxn.ics.uci.edu/rmechdb/download`. The data set is licensed under the *Creative Commons Attribution-NonCommercial-NoDerivs (CC-BY-NC-ND)* license, which limits its free public usage to non-commercial purposes. Under this license, the users are not allowed to modify and distribute the data set or to distribute the original data set without referencing the original source. After submitting basic information (name, email, and institution) and accepting the license terms, users receive an email containing a comma-separated value (CSV) file containing all the data and metadata.

2.8.3 Uploading New Data

While we continue to insert new data in RMechDB, we invite the community to contribute new radical elementary steps. Uploading new data can be done at: `https://deeprxn.ics.uci.edu/rmechdb/upload`.

Contributing users must fill out two fields: (1) the SMIRKS of the elementary step; and (2) the corresponding electron flow specification (codes for arrow pushing) as shown in Figure 2.5. There are also two optional fields where the user can provide information about the source of the elementary step (e.g. the title of a textbook, or a publication) and provide an optional note (e.g. the necessity of initial energy). After uploading the elementary step, it will be checked for validity, duplication, and plausibility (Figure 2.7).

Validity Check

A submitted elementary step is considered to be valid if it satisfies the following three criteria:

1. The SMILES string of all the molecules on both sides of the submitted elementary step must be correct and convertible to graphs representing valid molecules. We use the Openeye Scientific Software [96] toolkit OEChem [97] to convert the input SMILES/SMARTS strings into molecular graphs.

2. The annotations for the arrow-pushing mechanisms must be correct. This implies

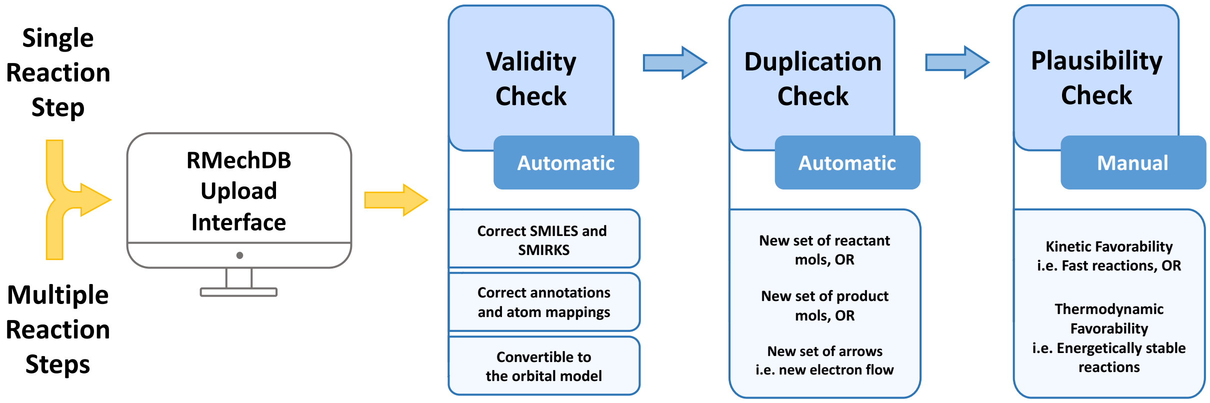

Single Reaction Step
Multiple Reaction Steps
RMechDB Upload Interface
Validity Check
Automatic
Correct SMILES and SMIRKS
Correct annotations and atom mappings
Convertible to the orbital model
Duplication Check
Automatic
New set of reactant mols, OR
New set of product mols, OR
New set of arrows i.e. new electron flow
Plausibility Check
Manual
Kinetic Favorability i.e. Fast reactions, OR
Thermodynamic Favorability i.e. Energetically stable reactions

Figure 2.7: Schematic depiction of how new data contributed to RMechDB and goes through different checking stages.

that the reacting atoms on the reactant side of the elementary step must be labeled with distinct integers. These integers form the basis for the arrow-pushing mechanisms associated with electron transfers. The arrow codes must be consistent with the integers used to label the reacting atoms. An example of a valid atom mapping and arrow codes is shown in Figure 2.5.

3. The entered SMIRKS and arrow codes are then used to extract the interacting orbitals. We used our elementary step model described in Section 2.7 to create the elementary step object. Using this object, we extract the interacting molecular orbitals and their corresponding atoms. If the input SMIRKS and arrow codes fail to create the elementary step object, the input is considered invalid. This failure usually implies a mismatch between the labeled atoms and the corresponding arrow codes.

Duplication Check

In this step, we check that the valid uploaded elementary step is not equivalent to any elementary step already included in the RMechDB data set. We consider two steps to be equivalent if they have the same:

1. Canonicalized SMILES string of the reacting molecules.
2. Canonicalized SMILES string of the product molecules.
3. Canonicalized SMILES string of the spectator molecules.
4. Conventional representation of the codes for the arrow pushing mechanism. The labels of the participating atoms on the nucleophilic component start from 10 with increments of one per atom, and the labels of the participating atoms on the electrophilic component start from 20 with increments of one per atom. It is important to mention that the user can use any integers to label the participating atoms. The conventional arrow codes will be automatically generated by RMechDB.

Once an elementary step is uploaded, RMechDB performs the validity and duplication tests automatically. In case of failure of either test, an informative error message is displayed with details about the corresponding errors.

Plausibility Check

Once the submitted elementary step passes both tests, it is further manually reviewed by the RMechDB curators for overall quality and plausibility, before being imported into the RMechDB.

2.9 Conclusion

The main obstacle to the large-scale application of AI methods to chemical reactions is the lack of data [20]. Some efforts have begun to try to address this fundamental bottleneck at the level of chemical transformations [68, 59]. Here we have presented a complementary effort aimed at building an open platform and database, RMechDB, for elementary steps in radical reactions. A parallel effort is underway to cover also polar reactions.

Databases of elementary steps introduce a new perspective and new opportunities for computer-aided reaction prediction and modeling. In particular, when properly deployed, they should facilitate addressing the central problems of explainability and causality found in many applications of AI in chemistry and other domains. The ability to decompose a transformation into a sequence of elementary steps is one way to understand how and why it occurs.

The RMechDB platform is designed to facilitate training deep learning and other AI models in data-driven workflows using its tabular data, with no need for additional pre-processing steps. While RMechDB is designed primarily to facilitate the training and evaluation of data-driven models for predicting all the potential outcomes of radical reactions, it can be used also for other tasks, such as reagent versus reactant classification, initial condition prediction, and reaction classification.

RMechDB is intended to be a live platform for contributing, aggregating, curating, and distributing data in the form of elementary radical reaction steps to accelerate research in chemoinformatics and reaction modeling. It provides a unified model that ought to facilitate data sharing, model building, dissemination, and publications. Future updates will be reported through the RMechDB website at `https://deeprxn.ics.uci.edu/rmechdb`. We encourage the community to explore and use the RMechDB data and functionalities and contribute to its expansion.

2.10 Data and Software Availability

RMechDB website is accessible at `https://deeprxn.ics.uci.edu/rmechdb`. The RMechDB data set can be downloaded through the download interface at `https://deeprxn.ics.uci.`

edu/rmechdb/download. Documentation on how to use the RMechDB interfaces is also provided at https://deeprxn.ics.uci.edu/rmechdb/howtouse.

Chapter 3

Radical Predictor: A Deep Learning System for Interpretable Radical Reaction Prediction

3.1 abstract

Deep learning-based reaction predictors have undergone significant architectural evolution. However, their reliance on reactions from the US Patent Office results in a lack of interpretable predictions and limited generalizability to other chemistry domains, such as radical and atmospheric chemistry. To address these challenges, we introduce a new reaction predictor system, RMechRP, that leverages contrastive learning in conjunction with mechanistic pathways, the most interpretable representation of chemical reactions. Specifically designed for radical reactions, RMechRP provides different levels of interpretation of chemical reactions. We develop and train multiple deep-learning models using RMechDB, a public database of radical reactions, to establish the first benchmark for predicting radical reac-

tions. Our results demonstrate the effectiveness of RMechRP in providing accurate and interpretable predictions of radical reactions, and its potential for various applications in atmospheric chemistry.

3.2 Introduction

Attaining chemist-level performance in predicting the outcomes of chemical reactions remains an unresolved challenge in computational chemistry. A solution to this challenge would have applications in many areas of chemistry both in the laboratory, for instance in synthesis, retrosynthesis, and drug design; and in nature, for instance: in biology and atmospheric chemistry. Here we focus on the prediction of an important class of reactions–radical reactions–at the level of mechanistic steps using machine learning approaches.

Currently, three main classes of approaches are being developed to address reaction prediction in general: (1) Quantum Mechanics (QM) simulations; (2) template-based predictions; and (3) template-free predictions. These approaches are not mutually exclusive and can be combined in different ways, as in [3].

Quantum Mechanics Simulations: Methods based on quantum mechanics and the simulation of chemical reactions at the atomic level provide the most accurate predictions of reaction outcomes. However, these fine-grained simulations are computationally expensive and may require human intervention [4, 5]. Quantum mechanics (QM) simulations are particularly sensitive and have operational limitations that restrict their applicability to a narrow range of chemical systems. Consequently, in current environments, QM simulations are not suitable for making high-throughput predictions of reaction outcomes.

Template-based methods: These types of methods often require a set of pre-defined chemistry rules that can be applied to a set of reactant molecules to predict chemical reaction outcomes. Thus these methods are also called rule-based methods. These rules, which are often referred to as ”chemical templates” can be extracted manually by human experts [6, 7] or can be computationally inferred [8] from a data set of chemical reactions [9, 10, 11]. In both cases, the templates are limited to either the knowledge of the human extractor or the chemistry of the given data set. Rule-based systems can produce rapid predictions and can be used in “brewing” experiments where multiple reactants are iteratively reacted with each other and their products [12, 13]. High-throughput template-based predictions can be applied to retrosynthetic experiments using programs such as Synthia [14, 15]. Machine learning methods have been used to help select templates [9, 16]. One important issue is that there is no obvious gradient in the discrete space of symbolic learning rules and thus efficient machine learning methods require finding a continuous embedding, leading to template-free methods. Another issue often seen with template-based methods is that they have limited generalization capabilities. The selection of general templates typically disregards local interactions between atoms in a small neighborhood, while the selection of specific templates may neglect the effects of reaction context, such as the presence of particular reagents.

Template-free methods: In contrast, template-free methods can achieve broad generalization to a wide range of chemical reactions while offering fast predictions. These methods, are purely based on machine learning predictive techniques and require large data sets for training and development [17, 18, 19]. This is a significant issue by itself since there is no comprehensive publicly available data set covering all, or even most, known chemical reactions [20]. Nevertheless, various partial data sets have become available enabling at least a partial development of these methods. Different methods can be developed for different representations of the reactions, which are rooted in the representation of the reactant and product molecules. There are at least four distinct ways of representing molecules: labeled graphs, text strings (e.g. SMILES strings leading to SMIRKS reaction representations), fingerprints, and sets of atoms with their 3D coordinates). Here, as in most of the literature we focus on machine learning methods based on graphs or text.

Graph-based methods represent each molecule as a graph with nodes associated with atoms, and edges associated with bonds[21, 22, 23, 24]. A chemical reaction is described by the graphs associated with the reactant molecules on one side, and the product molecules on the other. These labeled graphs can be processed using some form of recursive, or graphical, neural networks using an inner or outer (convolutional) approach [25, 26, 27, 28, 29] . In these methods, neural networks are used iteratively to pass information between graph neighborhoods and compute vectorial representations that are increasingly more abstract and encompassing. Such contextual representations can be exploited to generate graphs of product molecules using different approaches. For instance, [30] leverages these representations to identify graph edits that lead to the product molecules, whereas [31] employs a generative model to yield the 2D graph of the products.

Text-based methods typically utilize the simplified molecular-input line-entry system (SMILES) and its extension to reactions (SMIRKS) [32]. Various kinds of neural network architectures, ranging from recurrent/recursive to transformer architectures, can be used to process such variable-length text strings, similarly to what is done for natural language processing (NLP) problems [33, 34, 35]. Reaction prediction can be cast as a sentence-to-sentence or "translation problem" from the reactant sentence to the product sentence [36, 37]. Typically an encoder module is used to encode the discrete string associated with the reactant molecules into a continuous vector representation, and a decoder module is used to decode the continuous vector representation into the discrete string associated with the product molecules [38, 35]. The encoding and decoding processes are implemented using deep recurrent neural network architectures that may include long short-term memory (LSTM) [39] units and Transformers [40]. The decoder module can also leverage techniques such as beam search and attention mechanisms to generate the most accurate SMILES string [35, 38].

There is no foundational advantage for any of these machine learning approaches over the other ones, although preliminary evidence [38] together with the current success enjoyed by large language models suggest that text-based methods may be particularly effective. This is in spite of the facts that: (1) SMIRKS do not contain all the information needed to accurately model molecules and reactions [41]; (2) the same reaction (resp. molecule) can be represented by multiple SMIRKS (resp. SMILES), including those associated with any permutation of the molecules within the reactants or within the products[42]. Finally, text-based methods, as well as other machine learning methods, require large training sets, which is a major challenge in reaction chemistry. Compensatory methods, such as regularization and data augmentation, are being used but are often not enough.

The majority of recently created models, independently of the way they represent reactions, are being designed and trained using the chemical transformation dataset derived from the US Patent Office (USPTO) [43] and a few other minor datasets that were introduced in several studies [28, 44, 45]. All these data sets have significant limitations, including: lack of chemistry coverage, lack of precise and complete atom mapping, lack of balance between reactants and products, and lack of elementary reaction step information. The USPTO dataset of chemical reactions, for example, portrays chemical reactions restrictively as overall transformations, the majority of which result in a single primary product (and are thus unbalanced). This dataset provides little information about the underlying mechanisms, critical intermediates, and side products. Additionally, it is challenging to extract information about radical reactions, which appear to be underrepresented in this data set.

Utilizing this limited source of data to train data-driven models results in models that may be chemically limited or biased, predict unbalanced reactions, and lack information about intermediate byproducts. Additionally, these models provide no interpretations of the underlying chemistry that leads to the predicted products. Although There has been some work (e.g. [35]) using machine learning interpretability methods to highlight the importance of certain portions of the reactants, this does not provide any information on the mechanistic interactions underlying the reactions. Lastly, due to the limited available training data, there is no widely used reaction predictor for radical reactions. Radical reactions play a significant role, for instance in synthetic pathway planning and biological and atmospheric chemistry, and they often proceed through a complex series of chemical steps and highly branched mechanistic pathways. Thus developing an accurate radical reaction predictor free of some of the shortcomings mentioned above is important.

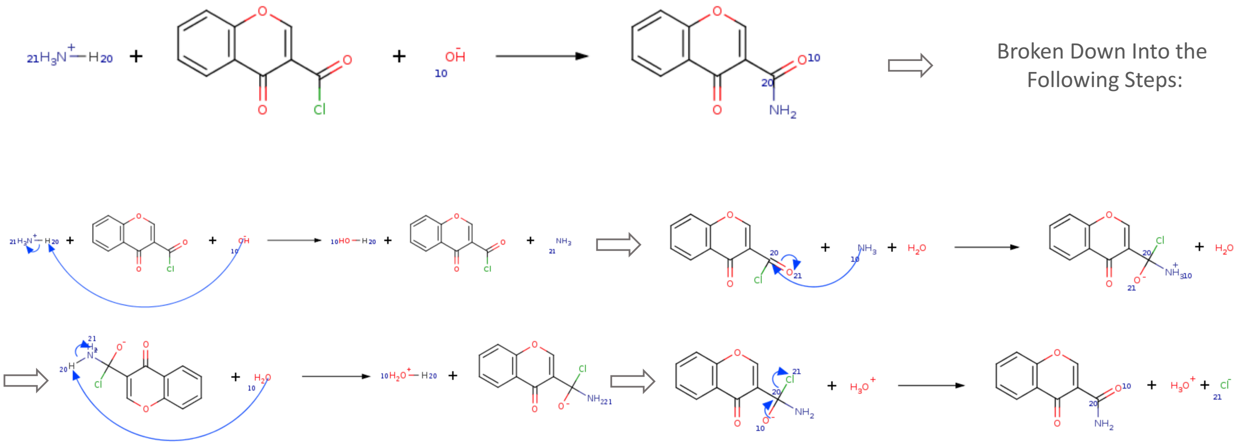


Figure 3.1: The reaction at the top is an overall, unbalanced, transformation from the USPTO data set. It can be broken down into four mechanistic steps with arrow-pushing mechanisms. This provides chemical interpretability for each step, as well as for the overall pathway, while maintaining full balance at each step.

3.3 Interpretability of Mechanistic Reaction Steps

Chemists typically employ an alternative approach to represent and conceptualize chemical reactions. Specifically, they utilize an intuitive representation that involves the consideration of single-state transitions and arrow-pushing mechanisms, which we refer to as elementary reaction mechanisms or elementary steps. Overall chemical transformations can be deconstructed into a chain of elementary reaction mechanisms, each characterized by a singular transition state [56, 101]. Figure 3.1 provides an illustrative example of this process. Curved arrows (or fish-hook arrows for radical reactions) are employed to depict the reaction mechanisms [62], and correspond to the interaction of singly occupied molecular orbitals with both the highest occupied molecular orbital (HOMO) and the lowest unoccupied molecular orbital (LUMO) [71]. Hence, the development of a reaction predictor that operates at the reaction mechanism level can confer three critical benefits (shown in Figure 3.1 that none of the current reaction predictors can offer.

Chemical interpretability: The first key benefit is enhanced chemical/orbital interpretability. The use of curly arrows, or arrow-pushing mechanisms, allows for an accurate understanding of the fundamental chemistry underlying each reaction step. This approach facilitates the understanding of the interactions between molecular orbitals, which ultimately drive each reaction step.

Pathway interpretability: The second key benefit is enhanced pathway/transformation interpretability. A predictor trained to predict elementary steps, can be iterated to expand a tree of such steps rooted at the initial reactants. This allows for the interpretation of any overall all transformations leading to several final products, some of which are unknown. By expanding such tree of pathways there is no chance of missing key intermediates that give rise to competing pathways during change. This level of interpretability enables several applications, most importantly for drug discovery and atmospheric chemistry where highly reactive radicals are present.

Balance and atom mapping: Finally, the third benefit is the preservation of the balance between reactants and products, in conjunction with the underlying atom mapping. The balance is maintained at all times, throughout the chain of reaction steps, which can be highly valuable, for instance when studying retrosynthesis pathways [102].

Table 3.1: The size of the various subsets of elementary radical reaction steps contained in the RMechDBdatabase [1]. These are used to train and test the predictors.

	Train	Test
Core	1512	150
Specific	3397	367
Combined	4909	517

3.4 Data

To develop a mechanistic-level radical reaction predictor, we utilize the standard train and test sets from the recently released RMechDB dataset [1]. This dataset comprises approximately 5500 radical elementary step reactions sourced from chemistry textbooks (core reactions) and scientific articles on atmospheric chemistry (atmospheric reactions). Each reaction in the RMechDB dataset is labeled with different categorizations enabling comprehensive evaluation of the predictive models trained on RMechDB. See Table 3.1 for a summary of the RMechDB dataset used in subsequent training and testing experiments.

3.5 Methods

Here we first describe the OrbChain, a model we developed to represent and process radical mechanistic reactions. Then we describe three different machine-learning approaches for predicting the outcome of radical mechanistic reactions with or without their associated arrow-pushing mechanisms. The first approach follows the methodology described in [56, 101, 103] where predictions are carried out using deep learning in two steps: a first one to identify reactive sites, and a second one to rank the plausibility of all reactive-site pairs. In the second approach, we directly predict the most reactive pair of molecular orbital. Within this direct approach, we test two different representations for the atoms: our own atom descriptor and RxnHypergraph [42] which applies a transformer model to the molecular graph representations. Lastly, we use a purely text-based approach leveraging transformers and large language models to predict reactions viewed as a "translation" from reactant to products [35]. Within this text-based approach, we test two different sequence-to-sequence transformer models: Molecular Transformer and Chemformer [38, 35]. These models are pre-trained on other data and fine-tuned on the RMechDB data with data augmentation. Both models output the string associated with the products, without the arrow-pushing information.

3.5.1 OrbChain: Model of Mechanistic Reactions

A radical mechanistic reaction is a reaction with a single transition state that involves at least one half-occupied orbital. More precisely, In the arrow-pushing mechanism representation, a radical mechanistic reaction consists of a set of reactant molecules $R = \{R_i\}_{i=1}^{n_r}$, a set of product molecules $P = \{P_i\}_{i=1}^{n_p}$, and a set of fish-hook arrows A showing the cleavage or movement of a single electron. Each of the reactant or product molecules (R_i or P_i) is represented by a connected molecular graph $G_i = (N_i, V_i)$ where the vertices N_i represent the atoms and V_i represent the bonds. The variable $\{a_j^i\}$ represents atom j in molecule i and $\{b_j^i\}$ represents bond j in molecule i. Inspired by [101], Orbchain models a radical mechanism with a single transition state as the interaction between two reactive Molecular Orbitals (MOs) m_1^* and m_2^* (we refer to a MO as m and a reactive MO as m^*). Each MO is associated with four distinct parameters $m = (a, e, n, c)$, where a represents the atom corresponding to the MO (i.e., the central atom of the MO), e denotes the number of electrons involved in the MO (0, 1, or 2), n indicates the atom adjacent to the atom a in the case of a bond orbital (such as a π or σ bond), and c signifies the possible chain of filled or unfilled MOs (such as a π system). The reactive orbitals are identified by following the sequence of electron transfers in the arrow-pushing diagram (Figure 3.2).

For a given mechanistic reaction, Orbchain uses the atom-mapped reactants R, the atom-mapped products P, and the arrow codes A ([1]), to uniquely determine the pair of reactive orbitals, $(m_1^*,\ m_2^*)$, in R. Alternatively, given that atom-mapped reactants R and a pair of orbitals $(m_1,\ m_2)$, Orbchain can uniquely determine the atom-mapped products P' and the arrow codes A'. Thus the two main functionalities of Orbchain can be summarized schematically by:

3.1.

$$\text{OrbChain}: \begin{cases} (1)\ R, P \xrightarrow{A} (m_1^*,\ m_2^*) \\ (2)\ R \xrightarrow{(m_1, m_2)} P', A' \end{cases} \tag{3.1}$$

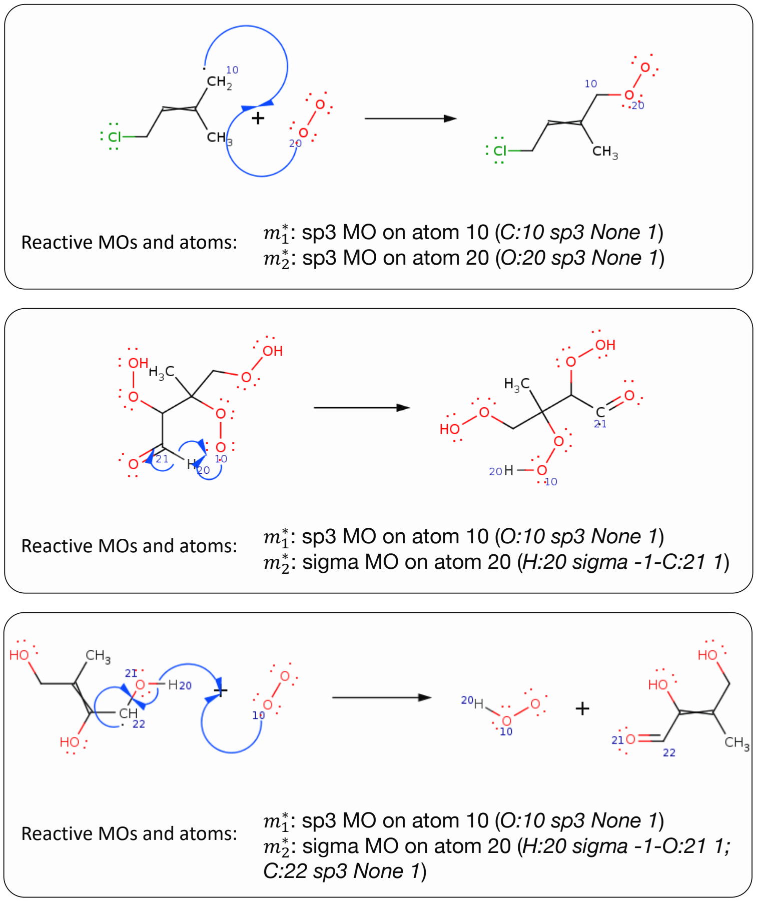

10
CH_2
Cl
CH_3
+
O
O
20
10
O
O
20
Cl
CH_3
Reactive MOs and atoms:
m_1^*: sp3 MO on atom 10 (*C:10 sp3 None 1*)
m_2^*: sp3 MO on atom 20 (*O:20 sp3 None 1*)
OH
H_3C
OH
O
O
O
O
21
H
20
10
OH
O
H_3C
O
HO
O
C
21
O
20H
O
10
Reactive MOs and atoms:
m_1^*: sp3 MO on atom 10 (*O:10 sp3 None 1*)
m_2^*: sigma MO on atom 20 (*H:20 sigma -1-C:21 1*)
HO
CH_3
21
O
H20
CH
22
HO
+
O
O
10
20H
O
10
O
+
HO
HO
21O
22
CH_3
Reactive MOs and atoms:
m_1^*: sp3 MO on atom 10 (*O:10 sp3 None 1*)
m_2^*: sigma MO on atom 20 (*H:20 sigma -1-O:21 1; C:22 sp3 None 1*)

Figure 3.2: Three radical mechanistic reactions from RMechDB. For each reaction, the reactive orbitals and reactive atoms are extracted using Orbchain.

3.5.2 Two Step Prediction

In the two-step prediction approach [56], given a set of reactants, we first identify all possible orbitals, then pair them in all possible ways, and finally find the most likely products. While chemically sound, this approach runs into computational complexity issues as the number of orbital pairs is quadratic in the number of orbitals. Thus we first apply a filtering step to reduce the number of candidate orbitals (i.e., likely to be reactive orbitals) that need to be paired. Second, we rank all possible pairs of reactive orbitals. Both steps are carried out using a deep neural network trained on the RMechDB data.

Reactive Sites Identification

As described above, a MO is defined using four parameters: (a, e, n, c), where a is the atom associated with the MO. Note that a MO is associated with a unique atom a, whereas an atom a can be associated with multiple MOs. Thus, in order to train a predictive model to filter the molecular orbitals, it is convenient to first train a predictive model to identify potential reactive atoms, and then consider all the orbitals associated with these reactive atoms. Thus the labeling of the reactive atoms in the training data corresponds to:

$$g(a_j^i) = \begin{cases} 0 & m^* \notin \{m \mid m = (a_j^i, e, t, c)\} \\ 1 & m^* \in \{m \mid m = (a_j^i, e, t, c)\} \end{cases} \tag{3.2}$$

The filtering neural network takes as input a vector describing an atom and its environment and produces an output via a logistic function with binary targets (reactive vs non-reactive). The input vector is constructed using a method similar to [56]. The components of this vector correspond to both atomic features and graph-topological features. Examples of atomic features include valence and electronegativity. Examples of graph-topological features include the counts or presence/non-presence of specific labeled paths and trees starting at the atom being considered. The complete list of these features is given in the Appendix. For all the atoms within the RMechDB datasets, we extract these feature vectors to train one feed-forward fully connected neural network that can classify each atom as reactive or non-reactive. The architecture and hyper-parameters of the trained model are given in the Appendix. As an alternative, which does not require the extraction of the graph-topological features, we also train a graph convolution neural network (GNN) [26, 27, 9] augmented with attention mechanisms [29], which uses only the atomic features as its inputs. The architecture and hyper-parameters parameters of the GNN are given in the Appendix. The two classification architectures are trained using the standard cross-entropy loss function.

The canonical training set in the RMechDB repository contains 4909 training reactions. To train each classifier, we iterate through 4909 training reactions from the RMechDB dataset. For each sample (R, P, A), we first find all the MOs within R. Then using P and A, we find the pair of reactive orbitals $(m_1^{(*)}, m_2^{(*)})$. All other orbitals are considered non-reactive MOs. Then using the function g, the corresponding atom of each MO a_j^i is augmented with a label $y_j^i = g(a_j^i)$ according to Equation 3.2. The iteration results in a dataset of atoms and their associated binary labels y. Using all the reactions in the RMechDB training set, we extract 73630 pairs of (a, y) with roughly 9800 reactive atoms (atoms with label 1). We re-weight the reactive and non-reactive classes to compensate for the class imbalance.

The results of reactive site identification are presented in Table 3.2. While these methods achieve reasonable accuracies, they overlook an essential aspect: the context of the reaction involving spectator and reagent molecules. As the reactivity of different sites and functional groups can be influenced by the reaction context, we propose a context-aware approach for reactive site prediction to address this limitation.

Plausibility Ranking

After identifying the reactive atoms within a set of reactant molecules, we consider all possible pairings between reactive atoms. More precisely, for each reactive atom we consider the complement of orbitals associated with it, Then we consider all the possible pairings of orbitals from the complement of the first reactive atom with orbitals from the complement of the second reactive atom. In a typical case, several of these orbital pairings can be discarded because they are not chemically possible (i.e., Orbchain cannot produce the products P' in Equation 3.1. For each viable pairing of orbitals, Orbchain produces the products and the arrow codes.

Subsequently, we rank these generated reactions to enhance the accuracy of reaction prediction, specifically aiming for top-N accuracy. To achieve this, we employ a deep Siamese architecture [104, 105] neural network [106, 103, 56]. This neural network architecture is well-suited for ranking entities. The shared module of the Siamese architecture is a neural network that takes as its input a representation of a mechanistic reaction, including reactants and products, and produces a real-valued numerical output. The Siamese architecture uses this module twice (weight sharing) to compare two mechanistic reactions. The parameters of the shared neural network module are optimized by minimizing the following loss function (per example):

$$\mathcal{L} = 1 - \sigma(f(\text{Rxn}_{plausible}) - f(\text{Rxn}_{implausible})) \tag{3.3}$$

where f is the function computed by the shared network, σ is the logistic function, $\text{Rxn}_{plausible}$ represents the reaction between the two reactive orbitals m_1^*, m_2^*, and $\text{Rxn}_{implausible}$ represents the reaction between two orbitals m_1, m_2 other than the reactive ones. By minimizing the loss function above, the function f tends to assign higher scores to the plausible reaction. Thus, the output of the function f can be interpreted as the plausibility score.

The performance of this model can be significantly affected by two key factors: the representation of the input reactions ($\text{Rxn}_{plausible}$ and $\text{Rxn}_{implausible}$), and the architecture of the neural network computing the function f. To study this effect, we deploy four reaction representations with their suitable neural architecture. More precisely, we use: (1) the predefined features vectors introduced in [56]; (2) reactionFP [107] with different molecular fingerprints including Morgan fingerprints [14], AtomPair (AP) [108] fingerprints; and Topological Torsions (TT) [109] fingerprints; (3) Differential Reaction Fingerprints (DRFPs) introduced in [110]; and (4) rxnfp [35] based on a transformer model pre-trained on the US PTO data [43]. To train the Siamese neural network, we use the reactions of RMechDB as plausible reactions (i.e., $\text{Rxn}_{plausible}$). For each plausible reaction (R, P, A) with reactive orbitals (m_1^*, m_2^*), we use the second functionality of OrbChain to produce 50 implausible reactions (R, P', A') with reactive orbitals (m_1^*, m_2), (m_2^*, m_2), (m_1, m_2^*), and (m_1, m_2). Here m_1 and m_2 are randomly chosen non-reactive MOs. The hyperparameters of each neural network are given in the Appendix. The comparative results of the plausibility ranking are presented in Table 3.3.

3.5.3 Contrastive Learning

As stated above, the context of a reaction can affect the dynamic of orbital interactions by changing the reactivity of different functional groups. An informative atom representation for reactive site identification must take this context into account. In this section, we solve this problem by proposing new methods that compute and learn the atom representations by considering the entire context of the reaction. The key idea is that instead of predicting the reactive atoms separately, we can predict the most reactive pairs of MOs directly, in one step. In other words, we must approximate the following probability distribution:

$$\mathcal{P}((m_i, m_j) = (m_1^*, m_2^*)|R) \tag{3.4}$$

Atom Pairs and Atom Descriptor

To establish a baseline for approximating the probability mentioned above, we utilize contrastive learning methods. In this approach, the positive data consists of the most productive reactions (R, P, A)(i.e., each sample of the RMechDB dataset), while the negative data includes all other possible reactions from the same set of reactants (R, P', A'). To train the contrastive model, reactions, whether positive or negative are represented as a pair of atoms where each atom is the representative of its reactive MO. The targets y_{ij} for an atom pair (a_i, a_j) is obtained as shown in Equation 3.2. We calculate the marginalized probability (Eqn. 3.4) by considering all possible atom pairs in R. This approach has two advantages: (1) It enables one-step reaction prediction by identifying the most reactive pair of MOs, determining the product and arrows according to Orbchain (statement (2)). (2) It reduces false negatives by not discouraging less reactive, yet still plausible, MO pairs. These pairs are ranked highly, with the most reactive MO pair receiving the highest ranking.

$$y_{(a_i,a_j)} = \begin{cases} 1 & m_1^* = (a_i, e, n, c) \;\; \& \;\; m_2^* = (a_j, e', n', c') \\ 0 & \text{Otherwise;} \end{cases} \tag{3.5}$$

Figure 3.3 (left side) shows the schematics of the contrastive model. Table 3.4 presents the results of this method, while the Appendix provides details on the objective function, parameters of the contrastive model (Figure 3.3), and the atom descriptor.

Rxn-Hypergraph

Considering all atom pairs in a contrastive learning fashion would take the reaction context into account. However, we can compute a more informative representation of atom pairs by utilizing the Rxn-Hypergraph. Rxn-Hypergraph introduced in [42] is a graph attentional neural network that operates on the hypergraph of the entire reaction. Instead of using an atom descriptor, we can train a Rxn-Hypergraph model, to automatically learn a contextual representation of all atoms of the reactant sides.

To adapt the Rxn-Hypergraph for the reaction predictor task (OrbChain Statement (2)), we make a modification by duplicating the reactants on both sides of the graph. This modification allows us to provide the reactants on one side while maintaining the full context on the other side. Following the training procedure of [42], we compute atom representations and generate pairs of atoms. These pairs are then fed into the contrastive network depicted in Figure 3.3 to obtain a ranked list of atom pairs. Each atom pair corresponds to an interaction between two orbitals, which enables us to generate products and arrows using OrbChain. The results of the reaction predictor using atom pair prediction with Rxn-Hypergraph are presented in Table 3.4.

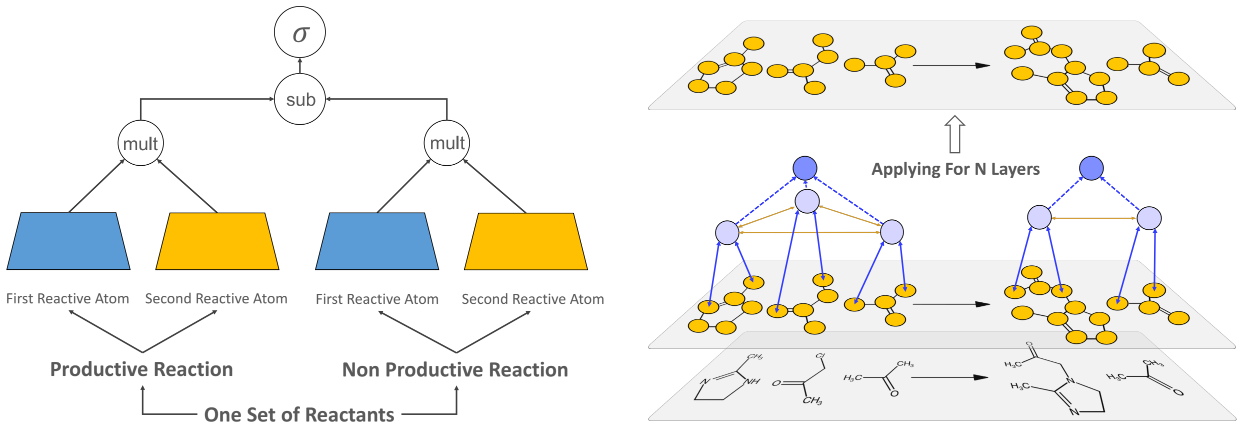


Figure 3.3: Left: The architecture of the contrastive learning approach. Right: The schematic depiction of the Rxn-hypergraph.

3.5.4 Text Representation and Sequence to Sequence Models

Considering the SMILES string as the text representation of molecules [32, 111], a chemical reaction can be seen as the transformation of a sequence of characters (reactants) to another (products). This makes the sequence-to-sequence models, such as the Transformer [36, 40] and models based on the recurrent neural network architecture, a suitable predictive model for chemical reaction predictions [38, 33, 34, 35], retrosynthesis prediction [112] and molecular optimization [113].

Existing text-based models for chemical reaction prediction have limitations, including non-interpretable and non-balanced predictions, as well as the need for extensive data augmentation due to the text representation of molecules breaking the inherent permutation invariance in reactions. However, we aim to leverage the success of these models and apply them to the prediction of radical mechanistic reactions. In particular, we adopt the pioneering text-based reaction predictor, Molecular Transformer [35], which utilizes a bidirectional encoder and autoregressive decoder with a fully connected network for generating probability distributions over possible tokens. The pretrained Molecular Transformers were trained using different variations of the USPTO dataset [43]. During training the encoder computes a contextual vector representation of the reactants by performing self-attention on the masked and randomly augmented (non-canonicalized) SMILES string of the reactant molecules. The decoder then uses the encoder output and the right-shifted SMILES string of the products to autoregressively generate the product tokens. Since the radical reactions in RMechDB are not labeled with reactants and reagents, we used the model which was pretrained using the USPTO_MIT_mixed dataset [43, 57, 114].

3.5.5 Fine-tuning Using RMechDB

Molecular Transformer enables the fine-tuning of pretrained models for downstream tasks like radical reaction prediction. In our approach, we utilized pretrained models and conducted reactant-to-product sequence translation. During the fine-tuning process, our only augmentation technique involved rearranging the reactant molecules within the SMILES string. Specifically, for each reaction containing N reactant molecules, we employed N

SMILES strings with reactants randomly reordered. We removed all atom mappings from the RMechDB training data and fine-tuned the model using the entire training set of the RMechDB.

Table 3.4 shows the performance of the text-based prediction for both pretrained and fine-tuned versions of the Molecular Transformer. Within the Appendix, we illustrate the phases of radical reaction prediction using Molecular Transformer. We also include detailed information on training and fine-tuning parameters, as well as tokenization statistics.

3.6 Results and Discussion

3.6.1 Performance on RMechDB

We evaluate the performance of the two-step prediction method, which includes reactive site identification and plausibility ranking. Table 3.2 displays the results of reactive site identification on the combined test datasets of RMechDB using the Top$N$ evaluation metric. More detailed Top$N$ accuracy for each RMechDB test set can be found in the Appendix.

We can see that GNN models can outperform the method based on the atom descriptor (predefined feature extraction). This behavior is expected as the atom descriptor is limited to a certain radius around the atom (in this case the radius is set to three). However, the number of GNN layers can be optimized to construct the most informative atom representations. The advantage of GNNs is more evident for the atmospheric test data where there are usually more molecules present in the context of the reaction.

The second model of the two-step prediction is the Siamese architecture that ranks the reactions based on their chemical plausibility. Four different architectures were used for the reaction fingerprint from [56], reaction*fp* from [107], DRFP [110], and the *rxnfp* [115]. The results of the topN accuracy and MRR ranking score [116] for the combined test sets of RMechDB are shown in Table 3.3. The table shows that DRFP outperforms the reaction*fp* and the feature extraction methods. We believe the reason is mainly because of the different nature and underlying chemistry of the radical mechanistic reaction and the USPTO reaction, which was used to pre-train the *rxnfp*.

Table 3.2: The performance of different methods for reactive site identification. Each number represents the percentage of reactions for which both reactive atoms are identified within the topN predictions.

Method	Top2	Top3	Top5	Top10
Atom Fingerprint	75.1	81.5	89.3	96.7
GNN	76.9	83.6	92.1	97.9

To predict the outcome of a mechanistic radical reaction using the two-step method, we must perform both reactive site identification and reaction ranking. Given the performance of each individual predictor, the best combination would be the GNN and DRFP. Therefore in Table 3.4, we use this combination to compare the performance of the other two methods with the two-step prediction.

Table 3.3: The performance of different methods for plausibility ranking. Each number represents the percentage of reactions for which the correct mechanism is predicted in the topN plausible reactions.

Method		Top1	Top3	Top5	Top10
Feature Extraction		73.1	79.2	88.3	96.3
reaction*fp*	AP	74.6	82.3	90.2	97.8
	Morgan2	74.3	81.9	90.0	97.3
	TT	74.3	82.4	90.0	97.8
DRFP		78.6	90.2	95.1	100.0
rxnfp (pretrained)		75.9	86.2	94.3	97.9

As it is shown in Table 3.4, the contrastive learning approach yields the most accurate prediction across all the metrics. In terms of inference time, the contrastive learning approach is faster than the two-step prediction as it only consists of one neural network. This becomes crucial in the case of pathway search where we exponentially expand the tree of the mechanistic pathways by predictions. However, the advantage of using the two-step method becomes more evident when the size of the reactant molecules increases. In Figure 3.4, we show the performance of all three methods based on the number of heavy atoms on the reactant sides. As expected, the prediction of both contrastive methods and text-based models becomes more faulty when big molecules react. This can be explained using the fact that the reactive sites identification part of the two-step method is mainly dependent on a local neighborhood of atoms and is not significantly affected by the size of the molecules.

The text-based models are outperformed by the other two graph-based methods. Although they offer faster inference times, they yield less accurate predictions. This poor performance is mainly because the Molecular Transformer model is trained on the USPTO_MIT_mixed dataset. Therefore, it learned to predict the only major product of overall transformations that are mostly polar. On the contrary, RMechDB data are balanced, mechanistic, and involve radical species. Align with this low performance, according to Table 3.4, fine-tuning on the relatively small RMechDB dataset results in a slight decrease in the performance. This can be attributed to the dissimilarities between RMechDB and the USPTO dataset. RMechDB comprises intermediate products of transformations and includes radical reactions with compounds not found in the USPTO dataset. During the tokenization process of the RMechDB reactions, new tokens emerge that have not been extracted from the USPTO dataset.

We also provided a separate performance of the predictors with respect to different radical reaction categories provided in RMechDB.

3.6.2 Pathway Search

After successfully predicting the outcomes of mechanistic radical reactions, we can chain these predictions to construct mechanistic pathways. Starting from a set of reactant molecules,

Table 3.4: Top N accuracy of all the three reaction prediction models on Core and Atmospheric test sets of RMechDB. For the text-based models, a prediction is considered to be correct, when at least one of the non-spectator product molecules are predicted correctly. For the text-based models, we use (p) and (f) for pretrained, and fine-tuned models respectively.

Model	Variant	Core				Atmospheric				Time
		Top N				Top N				
		1	2	5	10	1	2	5	10	
Two-step	Best Combination	62.4	71.9	93.2	97.2	60.4	70.9	91.6	96.3	1.38
Contrastive	Atom Descriptor	62.9	73.8	94.2	96.5	61.0	73.6	93.0	94.4	0.08
	RxnHypergraph	64.3	74.1	95.1	97.4	62.1	74.8	94.1	95.9	1.45
Text-based	Pretrained	58.2	64.3	84.2	91.0	58.0	67.3	82.6	91.0	1.30
	Fine-tuned	57.7	64.0	83.9	90.4	57.1	66.8	82.2	90.3	1.30

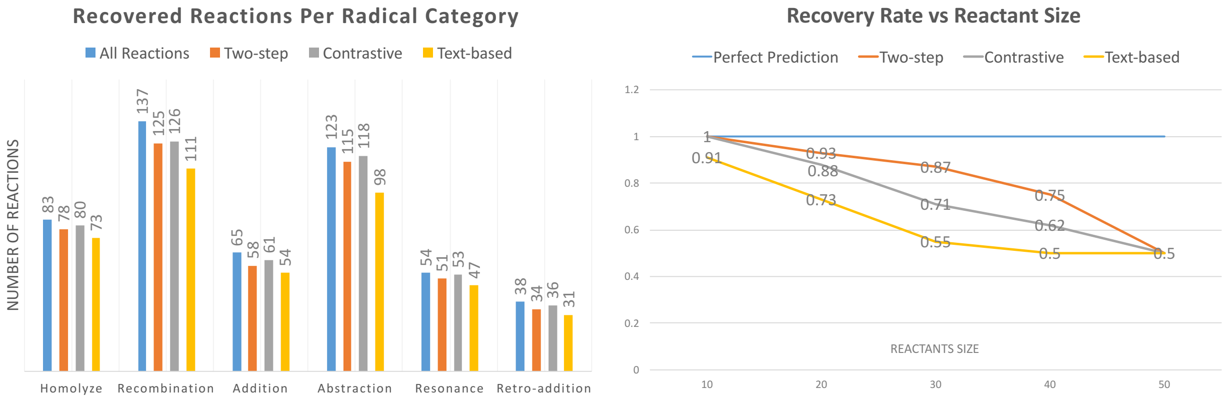


Figure 3.4: Left: The number of recovered reactions in top5 for different methods and different classes of RMechDB reactions. Right: The recovery rate of different methods with respect to the size of the reactants.

we perform a series of predictions by using each of the topN predicted products as the reactants for the next prediction. This would form a tree structure with the starting reactants as the root. This tree can be expanded to a desired depth, representing numerous mechanistic pathways leading to different products. These pathways are crucial for identifying synthetic routes, exploring intermediate products, and aiding in mass spectrometry analysis with broad applications to various fields, including drug design and drug degradation.

We create a dataset consisting of 100 radical reactants paired with target molecules that are expected to be formed from the reaction. Each pair is accompanied by a specified depth, indicating the length of the mechanistic chain reactions required to reach the target. We generate this dataset by simulating atmospheric conditions, taking inspiration from the atmospheric reactions observed in the RMechDB dataset and atmospheric literature [117]. The reactants in our dataset primarily consist of Isoprene, a prevalent atmospheric compound, along with other atmospheric molecules like radical Oxygen and Hydroxyl radical. To find these targets, we expand and search the tree of the mechanistic pathways by employing a breadth-first search algorithm. To expand the tree for each of the 100 reactants, we use the top 10 predictions at each step (i.e., the breadth of 10 to expand the tree), and we expand the tree up to the given depth. Given the average inference time of the predictive models presented in Table 3.4, we used the fastest method with interpretable predictions which is the contrastive model with atom descriptor. Upon running this pathway search for these 100 pathways, we observed a significant recovery rate of 60% meaning that for the 60% of the reactants, the given target was found within the expanded tree. These pathways along with their targets and detailed information on the pathway search are presented in the Appendix.

3.6.3 RMechRP Software

We develop and release the online RMechRP (**R**adical **Mech**anistic **R**eaction **P**redictor) accessible at `https://deeprxn.ics.uci.edu/rrp/`. This online predictor offers two interfaces: (1) Single-step prediction; and (2) Pathway search. The single-step predictor accessible at `https://deeprxn.ics.uci.edu/rrp/singlestep`, allows the user to input a set of reactant molecules. Then upon inserting a few parameters, such as the number of reactive atoms, the system will use the best model described above to find all the possible radical mechanistic steps. Then all the top predicted reactions will be displayed with side information, including arrow codes, SMIRKS, reactive orbitals, and plausibility scores. The pathway search interface accessible at `https://deeprxn.ics.uci.edu/rrp/pathway`, is designed for searching for a specific target molecule(s) through the mechanistic pathway of the input reactants. The user must input a set of reactants and a set of target molecules. After setting parameters such as the depth and breadth of the mechanistic pathways, the system will use the fastest predictive model described above to expand the mechanistic pathway as a tree rooted at the reactants. Once the target molecule is found within the expanded tree, the system will display the synthetic, mechanistic pathways from the starting reactants to the target molecule(s). More details on both interfaces and how to use them are presented in the Appendix.

3.7 Conclusion

We have successfully developed a radical reaction prediction system that offers a unique approach to reaction prediction by focusing specifically on radical reactions and operating at the mechanistic level. We trained and developed three deep learning models for radical reaction prediction, demonstrating that the contrastive learning method yields the most accurate results. By leveraging the RMechDB datasets, our radical predictor represents a significant advancement in interpretable reaction prediction. Furthermore, it provides various benefits, such as pathway interpretability and maintaining balance throughout the chain of mechanistic reactions, making it valuable for identifying synthetic pathways. Our predictor, RMechRP, as the only radical reaction predictor system, is available to the public through online interfaces available at `https://deeprxn.ics.uci.edu/rrp`.

3.8 Appendix

In this appendix, we provide a comprehensive description of the experimental details and environments in which the experiments were conducted. Additionally, we present detailed information and data pertaining to the pathway search. Furthermore, we offer an explanation of the various interfaces of the RMechRP software, which serves as the pioneering online radical reaction predictor. Each section in this appendix corresponds to the section with the same title in the main article. Finally, all the experiments are conducted using a single NVidia Titan X GPU.

3.8.1 Two Step Prediction

This method consists of two distinct steps, within each, we trained several neural networks. Here we explained the parameters used during the training of these networks.

Reactive Site Identification

For the Atom Fingerprint model, we constructed a fingerprint of length 800 for each atom. This fingerprint includes 700 graph topological features explained in [56] and 85 atomic features including a one-hot vector for atom type, and chemical features of the atoms such as valance and electronegativity. The graph topological features are extracted using a neighborhood of size three. The extracted fingerprints are fed into a fully connected model with an output layer for binary classification. For the GNN model, we used the atomic feature for the initial representations of atoms. The model consists of four GNN layers with an output layer for binary classification.

Combining both training sets presented in RMechDB [1], we extracted over 51000 atoms to train each of the models above. Both models are evaluated using a combination of two test sets in RMechDB and the topN accuracy of models is reported in Table 2 of the main article. Table 3.5 represents the parameters used for training the models.

Table 3.5: The parameters used for training the models for reactive site identification.

Model	Batch Size	Num Layers	Layers Dim	Act	Reg	Num Att Heads
Atom Fingerprint	32	3	512-256-1	GELU	L_2(5e-5)	-
GNN	32	4	64-64-64-1	ReLU	Dropout (0.3)	2

Plausibility Ranking

For the plausibility ranking experiments, we used the following four methods for representing chemical reactions:

Feature Extraction: We use the same features explained in [56] which results in extracting a vector of length 3200 for each reaction.

reaction*fp*: We use the RDKit [118] implementation of reaction*fp* [107]. For all three fingerprint types (Atom Pair, Morgan2, and Topological Torsions), we use a fingerprint of size 2048, with a bit ratio of 0.2. We considered non-agent molecules with a weight of 0.4 and agent molecules with a weight of 1.0.

DRFP: We use the DRFP fingerprint [110] with a size of 2048 with a min and max radius of zero and four, while including the hydrogen atoms and rings.

Feature Extraction: We use the default tokenizer and pretrained model for the *rxnfp* [115] which results in fingerprints of length 256.

For training, we use a combination of both training sets in RMechDB. For each sample of the training data (productive reaction), we generate (at most) 40 negative samples (unproductive reactions) by randomly sampling molecular orbitals other than the reactive MOs (m_1^*, m_2^*). This results in a data set of over 185000 pairs of productive and unproductive reactions. To train the plausibility rankers for each method, we use the parameters explained in Table 3.6.

Table 3.6: The parameters used for training the models for the plausibility ranking.

Model	Batch Size	Num Layers	Layers Dim	Act	Reg
Feature Extraction	32	3	512-256-1	GELU	Dropout (0.5)
reaction*fp*	32	3	400-200-1	GELU	Dropout (0.5)
DRFP	32	3	400-200-1	GELU	Dropout (0.5)
rxnfp	64	2	128-1	GELU	Dropout (0.5)

3.8.2 Contrastive Learning

Atom Pairs and Atom Descriptor

For the contrastive learning method using the atom pairs and atom descriptor, we use the same atomic feature and graph topological features above to represent one single atom. Specifically, for the graph topological features, we use the neighborhood of size one. These features plus the atomic features result in a vector of length 140 for atom representation. Using these vectors, we train a contrastive model depicted in Figure 2 (left) of the main article. The objective function to train this contrastive model is as follows:

$$\mathcal{L} = 1 - \sigma([f(a_1^*) \times g(a_2^*)] - [f(a_1') \times g(a_2')]) \tag{3.6}$$

Where a_1^* and a_2^* are the atoms of the reactive MOs m_1^* and m_2^*, while a_i' are randomly chosen atoms. Both f and g functions are characterized by a fully connected neural network. The first reactive atoms in both productive and unproductive reactions, are fed through the same network f, and similarly, the second reactive atoms are fed through the same network g. The outputs of both f and g are single real-valued numbers, which, when multiplied together, yield a score for the respective reaction. These scores are then utilized to construct the objective function, aiming to maximize the score of the productive reaction compared to the unproductive reactions using the same reactant set. Figure 3.5 represents a schematic depiction of this contrastive model.

We use a combination of both training sets in RMechDB to train f and g. For each productive reaction, we form unproductive reactions by considering at most 15 samples of $(a'_1,\ a^*_2)$, $(a^*_1,\ a'_2)$, and $(a'_1,\ a'_2)$. This negative sampling results in a dataset of over 200000 pairs of productive and unproductive atom pairs. We use this training dataset to minimize the objective function 3.6.

Both f and g have similar architectures that consist of three fully connected layers with a GELU activation function and a dropout with a rate of 0.5 applied to all layers. The dimensions of the layers are 128, 64, 1.

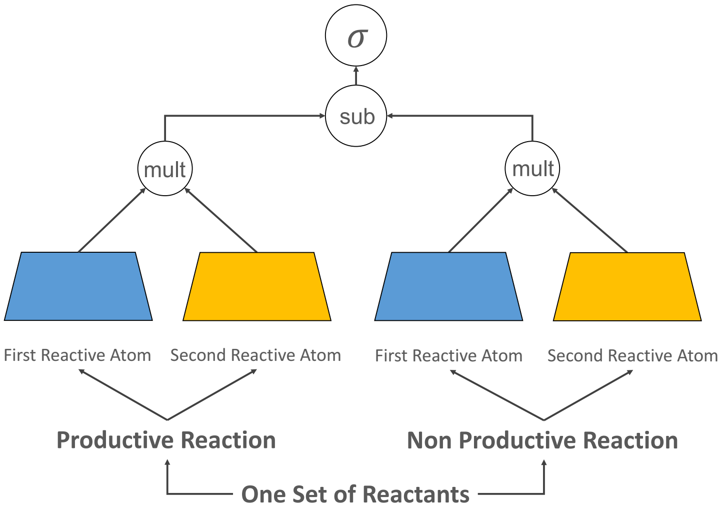


Figure 3.5: The architecture of the contrastive learning approach.

Rxn-Hypergraph

We use the Rxn-Hypergraph to replace form atom descriptors that are extracted automatically for minimizing the objective function 3.6. After processing the Rxn-hypergraph for N layers, the generated atom descriptors are used in the same setting above for the same minimization objective. Here in Table 3.7 we describe the parameters we use for training the Rxn-Hypergraph.

3.8.3 Text Representation and Sequence to Sequence Models

In order to develop a text-based radical reaction predictor, we utilize the pretrained molecular transformer which was trained using the USPTO_MIT_mixed dataset. We also used the tokenizer developed by Molecular Transformer. This tokenizer yields 523 distinct tokens for the USPTO_MIT_mixed dataset. There are nine tokens from the RMechDB dataset that do not match the 573 tokens of the USPTO. Therefore, we used the *unknown token* to represent these nine tokens.

For fine-tuning the pretrained model, we used the combination of both RMechDB training sets. We fine-tune the model using a simple data augmentation described in Section 4.5 for 10 epochs. Finally, for the evaluation of the text-based models, we considered all the generated *unknown token* as correct tokens.

3.8.4 Pathway Search

In the Pathway Search section, we conducted an experiment involving the execution of the pathway search for 100 specific reactants. Each of these reactants was associated with a desired target molecule, which was expected to be found within the mechanistic pathway tree. Additionally, a set of distinct parameters was assigned to each reactant to guide the pathway search process.

To provide detailed information and facilitate reproducibility, we have included supplementary materials accompanying the paper. Among these materials, you will find a file named *pathways.csv*. This file contains the reactants, corresponding targets, the provided context

Table 3.7: The parameters used for training the Rxn-Hypergraph for the contrastive model.

Batch Size	Num Layers	Layers Dim	Act	Reg	Num Att Heads	Learning Rate
32	5	all 64	GELU	L_2(5e-5)	2	0.001

(if any), and the anticipated depth at which the target molecule is expected to appear within the mechanistic pathway tree.

Furthermore, we have included another file titled *pathway_results.txt* in the supplementary materials. This file comprises the identified pathways leading to the specified target molecules. It presents the discovered pathways that were found during the experiment.

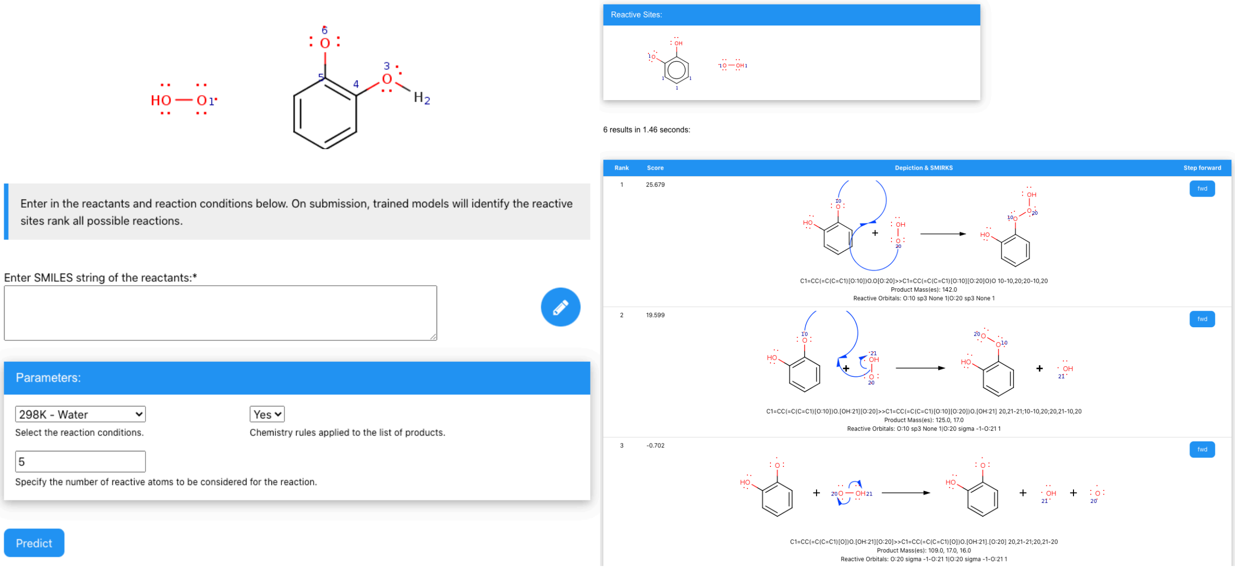
It is worth noting that the 100 pathways and their results will be published alongside the paper, subject to acceptance. These materials serve to provide comprehensive insights into the pathway search process and its outcomes, enabling readers to reproduce and further explore the obtained results.

3.8.5 RMechRP Software

In addition to the methods and results presented in the main article, we have developed an online web server that enables users to utilize the trained models for predicting the outcomes of mechanistic radical reactions with the highest levels of interpretability of the outcome. RMechRP (**R**adical **Mech**anistic **R**eaction **P**redictor) accessible via `http://deeprxn.ics.uci.edu/rrp`. RMechRP offers two interfaces: Single-step prediction and Pathway search.

Single-Step Prediction predicts the outcome of a mechanistic reaction with a single transition state. Users have the option to either input the reactants in written form or draw them using a drawing tool provided on the web server. Additionally, users can specify the reaction conditions, with the current option being standard temperature and pressure. The number of reactive molecular orbitals (MOs) to be considered can also be specified by the user.

To ensure flexibility, users can choose to filter out reactions that violate specific chemical rules, such as Bredt's rule [119]. Once the input and conditions are set, the user can click the Predict button. The system will then run the two-step prediction model, as described above, to generate and rank the potential products. These predicted products will be displayed, accompanied by additional information such as arrow codes, reactive MOs, and the mass of the products. The single-step predictor is accessible via `http://deeprxn.ics.uci.edu/rrp/singlestep`. Figure 3.6 shows the single-step interface and the displayed predictions for a simple reaction.



Enter in the reactants and reaction conditions below. On submission, trained models will identify the reactive sites rank all possible reactions.

Enter SMILES string of the reactants:*

Parameters:

298K - Water
Select the reaction conditions.

Yes
Chemistry rules applied to the list of products.

5
Specify the number of reactive atoms to be considered for the reaction.

Predict

Reactive Sites:

6 results in 1.46 seconds:

Rank	Score	Depiction & SMIRKS	Step forward
1	25.679	C1=CC(=C(C=C1)[O:10])O.O[O:20]>>C1=CC(=C(C=C1)[O:10][O:20]O)O 10-10,20;20-10,20 Product Mass(es): 142.0 Reactive Orbitals: O:10 sp3 None 1|O:20 sp3 None 1	fwd
2	19.599	C1=CC(=C(C=C1)[O:10])O.[OH:21][O:20]>>C1=CC(=C(C=C1)[O:10][O:20])O.[OH:21] 20,21-21;10-10,20;20,21-10,20 Product Mass(es): 125.0, 17.0 Reactive Orbitals: O:10 sp3 None 1|O:20 sigma -1-O:21 1	fwd
3	-0.702	C1=CC(=C(C=C1)[O])O.[OH:21][O:20]>>C1=CC(=C(C=C1)[O])O.[OH:21].[O:20] 20,21-21;20,21-20 Product Mass(es): 109.0, 17.0, 16.0 Reactive Orbitals: O:20 sigma -1-O:21 1|O:20 sigma -1-O:21 1	fwd

Figure 3.6: The single-step interface with the predictions of a simple reaction. Left: the input panel. Right: the table displaying the ranked predictions.

Pathway Search forms the tree of the mechanistic pathways up to a given depth and breadth. Users have the option to either input the reactants in written form or draw them using a drawing tool provided on the web server. Users must also input a set of targets (either mass or chemical structure) to look for within the expanded tree of the mechanistic pathways. users have the ability to provide a context for the reactions. The context consists of a set of molecules along with their corresponding frequencies of appearance within the mechanistic pathway tree. When a molecule from the context is consumed in a reaction, the system can automatically reintroduce that molecule back into the pathway tree. The frequency of appearance indicates how many times a molecule can be added to the mechanistic pathway tree.

In addition to the context, there are several additional parameters that can be specified by the user. These parameters include:

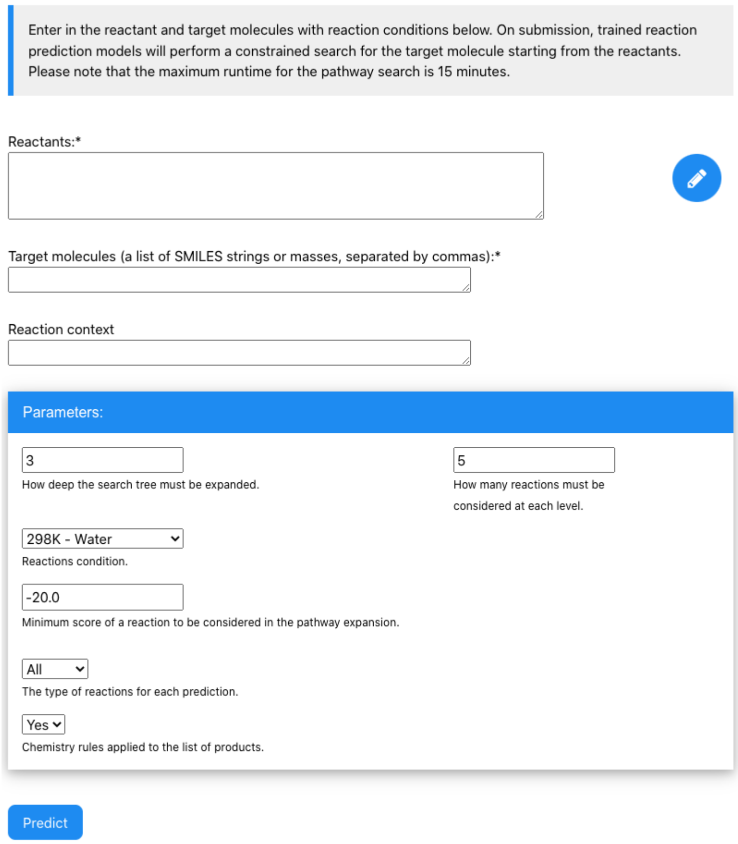
Depth of Pathway Search: Users can define the depth of the pathway search, which determines how many reaction steps will be explored in the mechanistic pathway tree.

Breadth (Branching Factor) of Pathway Search: This parameter controls the branching factor of the pathway search, influencing the number of alternative reaction pathways that will be considered.

Application of Chemistry Rules: Users have the option to apply certain chemistry rules during the pathway search. These rules can be used to filter out reactions that violate specific chemical principles or constraints.

Score Threshold: Users can set a threshold value to consider only reactions with scores higher than the specified threshold. This helps narrow down the focus to more favorable or promising reactions.

These additional parameters allow users to customize their pathway search and refine the results based on their specific requirements and preferences. By leveraging these features, users can gain deeper insights into the mechanistic pathways and explore a wider range of possible reaction outcomes. The pathway search interface is accessible via `http://deeprxn.ics.uci.edu/rrp/pathway`. Figure 3.7 shows the pathway interface and the required parameters.



Enter in the reactant and target molecules with reaction conditions below. On submission, trained reaction prediction models will perform a constrained search for the target molecule starting from the reactants. Please note that the maximum runtime for the pathway search is 15 minutes.

Reactants:*

Target molecules (a list of SMILES strings or masses, separated by commas):*

Reaction context

Parameters:

3
How deep the search tree must be expanded.

5
How many reactions must be considered at each level.

298K - Water
Reactions condition.

-20.0
Minimum score of a reaction to be considered in the pathway expansion.

All
The type of reactions for each prediction.

Yes
Chemistry rules applied to the list of products.

Figure 3.7: The pathway search interface.

Chapter 4

Rxn Hypergraph: a Hypergraph Attention Model for Chemical Reaction Representation

4.1 abstract

It is fundamental for science and technology to be able to predict chemical reactions and their properties. To achieve such skills, it is important to develop good representations of chemical reactions or good deep-learning architectures that can learn such representations automatically from the data. There is currently no universal and widely adopted method for robustly representing chemical reactions. Most existing methods suffer from one or more drawbacks, such as: (1) lacking universality; (2) lacking robustness; (3) lacking interpretability; or (4) requiring excessive manual pre-processing. Here we exploit graph-based representations of molecular structures to develop and test a hypergraph attention neural network approach to solve at once the reaction representation and property-prediction

problems, alleviating the aforementioned drawbacks. We evaluate this hypergraph representation in three experiments using three independent data sets of chemical reactions. In all experiments, the hypergraph-based approach matches or outperforms other representations and their corresponding models of chemical reactions while yielding interpretable multi-level representations.

Over the past few years, artificial intelligence has refashioned organic chemistry. Numerous problems such as chemical reaction prediction, synthesis route planning, drug design, etc., have benefited from the advancement of deep learning methods [30, 56, 35, 120, 121, 28]. For instance, accurate reaction yield predictions would help chemists to choose synthesis routes across high-yielding chemical reactions. As another example, estimating reaction rates *via* deep learning methods could circumvent the time and expense required to experimentally measure the reaction rate of reactions. Not to mention that because the true solution-phase reaction rates are bounded by the rate of molecular diffusion, it is impossible to measure the accurate rate experimentally. Lastly, predicting the outcome of chemical reactions using deep learning methods would automate and accelerate the demanding processes of drug design and discovery. All these problems require accurate predictions of chemical reactions' properties using deep learning models, consequently, necessitating optimizing every aspect of such deep learning models. One important facet that can significantly affect training dynamics and model performance is the representation of the input data.

In an attempt to find a suitable representation of chemical reactions for input into deep learning models, several methods have been proposed but each of them suffers from certain shortcomings. These shortcomings may be summarized with the following properties:

1. **Lack of universality** Several representations are derived based on predefined pattern matching algorithms [107, 14]. Since there is no learning process involved in deriving these representations, they cannot be automatically adjusted for different predictive

tasks.

2. **Lack of robustness** Atoms and molecules have no inherent ordering within a chemical reaction. Therefore, a robust representation must be invariant to permutations of atoms and molecules within a given reaction. Methods built upon the text representation of chemical reactions are an example of models with the lack of robustness [115, 53]. Using such methods, one can obtain different outcomes by only permuting atoms and molecules in the text representation of one single reaction.

3. **Non-interpretability** It is vital for chemists to understand the reasoning behind a reaction-level prediction. For example, capturing the correlation between the presence and absence of certain functional groups or the interaction between specific electrophiles and nucleophiles would provide useful insight for chemists [55, 4]. Thus it is important that a representation can provide means to interpret the final predictions.

4. **Need for expensive computations** Lastly, some other representations require hand-crafted implementations and processes in order to be used as the input of a predictive model. These hand-crafted processes usually include running pattern and subgraph matching algorithms [101] or performing massive data augmentations [122] which are extremely time-consuming.

In what follows, we review these methods and evaluate them based on the four mentioned shortcomings. Then we propose an augmented graph representation of a chemical reaction called *rxn-hypergraph* which is designed to fix these shortcomings and improve the proposed methods. Finally, through a set of experiments on the classification and plausibility-based ranking of chemical reactions, we empirically show the viability of our *rxn-hypergraph* representation.

4.2 Related Work

One commonly used method for numerically representing chemical reactions was introduced in [107]. This representation is called *reactionFP* and it is derived from the fingerprints of the molecules involved in the chemical reaction. The *reactionFP* can be described as follows:

$$w_1(\sum_{P_i} FP(P_i) - \sum_{R_i} FP(R_i)) + w_2(\sum_{A_i} FP(A_i)) \tag{4.1}$$

Where A, R, and P represent the molecular agents, reactants, and products respectively. w_1 and w_2 are two, potentially learnable, parameters that adjust the contribution between the agent molecules and the reactive molecules within the final representation. $FP(.)$ is a function that outputs a traditional fingerprint of a given molecule such as ECFP4 [14]. Since extracting the traditional molecular fingerprints is based on the presence or absence of predefined patterns and involves no learning process, this representation lacks universality and cannot be adjusted for a variety of reaction-level predictive tasks.

However, *reactionFP* was an early robust representation since Equation 4.1 is trivially invariant to the permutation of the molecules involved in the reaction. Additionally, many traditional molecular fingerprints such as ECFP4 [14] or AtomPair [108] are also invariant to atomic order within molecules, presenting a robust representation. Nevertheless, since the molecular fingerprints are obtained using non-invertible hashing mechanisms [123], this method cannot be easily interpreted to discover high-level patterns after learning. Additionally, according to Equation 4.1, obtaining *reactionFP* requires extra hand-crafted computations including: (1) an accurate atom mapping between reactants and products to identify the role of molecules in a chemical reaction; and (2) extracting the vectorized form of traditional fingerprints.

Graph-based methods have also been used to extract more informative reaction representations. To predict the outcome of chemical reactions, [56] used a neural network to rank a set of potential mechanistic reactions based on their thermodynamic plausibility. They form two separate count bit vector representations, one for the reactant and one for the product molecules by recursively counting the predefined paths and trees of different sizes rooted at each atom. Then, through a mutual information feature selection stage, they extract the most informative set of these count bits for a given downstream task. Finally, the chemical reaction may be represented as the difference between the count vector of reactants and the count vector of products. Since this method is based on a set of predefined patterns, it lacks universality in the sense that it cannot capture necessary information for different tasks. Although it is not discussed in [56], the mutual information feature selection step can potentially provide an additional interpretable view of the final prediction. However, this requires further expensive computations, not to mention that already massive computation is required for obtaining the count bit vectors.

Finally, the most successful reaction representation makes use of SMIRKS [111] of chemical reactions [35, 124, 115]. SMIRKS is a well-defined domain-specific language with special characters and grammatical rules for describing chemical transformations in the SMILES strings [32]. Authors in [115] deployed commonly used methods from natural language processing (NLP) to encode the SMIRKS of chemical reactions into a continuous vector. They train bi-directional encoder representation from transformers (BERT) [36] models to obtain task-specific reaction representations. They also trained large sequence-to-sequence transformer models for masked language model prediction (MLM) on the reaction SMIRKS. These models may also be used to extract pre-trained representations of any given chemical reactions [125]. These transformer-based methods are highly accurate across multiple reaction level predictive tasks [53, 120]. Such models provide a universal reaction representation that can be used for numerous reaction-level predictive tasks. Additionally, transformer architectures provide a character-level interpretable framework over the SMILES strings. However, one major downside of these NLP models is that their input, SMILES, and SMIRKS strings, are not permutation invariant with respect to the order of atoms and molecules. Depending on the canonical SMILES parsing algorithm, atom labeling, and a few other details, a single reaction may be correctly represented with many different SMIRKS. Some permutation invariance may be recovered through massive data augmentation, where all possible representations of the input reactions are generated and randomly selected during training [122]. It has been shown the performance of transformer models is highly dependent on the data augmentation, where they show surprisingly poor performance without the data augmentation [122, 35].

4.3 Methods

In this section, we describe the reasons for constructing the chemical reaction hypergraph (*rxn-hypergraph*) and why this hypergraph would yield an abstract and powerful representation of a chemical reaction. Then we explain the process of constructing the (*rxn-hypergraph*) for a given reaction, and finally, we discuss how to train neural networks using this hypergraph representation of chemical reactions.

4.3.1 Why Rxn-hypergraph?

Graph neural networks and their variants have become the preeminent tool for learning patterns and relations from graph-structured data [126, 127, 128, 23, 129, 31]. The main operation of graph neural networks is to recursively update the representation of nodes using a message-passing scheme only between the nodes and their neighbors. Then a readout function can be applied to the set of nodes' representations to provide an abstract representation of the entire graph.

There are several essential reaction-level properties where predicting them would be highly beneficial to the entire field of chemoinformatics. However, applying a graph neural network to the graph structure of chemical reactions in its most raw form would not lead to a rich representation that captures different properties of the reaction. Particularly, reactions are a more general form of a graph that consists of multiple disconnected graph components (molecules). The absence of a message-passing route between these components would result in node representations that are independent of the nodes and connections within the other graph components that are involved in the reaction. Consequently, applying any form of a read-out function to these independent node representations would result in a non-informative representation of the entire reaction.

On the contrary, self-attentional models (e.g. transformers) have been impressively successful in reaction-level property predictions [53]. The input to these models is the SMIRKS representation of a chemical reaction in which atoms are represented by alphabetical characters (tokens). Although this form of input representation is not invariant to the permutation, it provides a suitable structure for the transformer architectures. The key reason behind the success of such models can be found in applying multiple layers of self-attention mechanisms to every pair of input tokens. By this means, the representation of each token will be updated by attending to all other tokens including the atoms within other molecules.

Inspired by this crucial factor in representing a reaction, we form a hypergraph structure of a reaction by constructing efficient message-passing routes between every pair of atoms. These routes would improve the representation of the reaction by: (1) enabling message-passing schemes between every possible pair of atoms so the atom representations are updated with respect to all other atoms within the reaction, and (2) providing a learnable read-out function (i.e. pooling mechanism) which can attend to different parts in different levels of the reaction which are informative for a specific predictive task.

4.3.2 Constructing Rxn-hypergraph

A chemical reaction with N reactants and M product molecules is described by two distinct sets of disconnected graph components R and P. $R = \{G_i^r\}_{i=1}^N$ represents the set of reactant molecules, and $P = \{G_i^p\}_{i=1}^M$ represents the set of product molecules. Molecule G_i with n atoms (regardless of beginning a reactant or product molecule) is a graph $G_i = (V_i, E_i, A, S)$, where $V_i = \{a_j^i\}_{j=1}^n$ is the set of nodes (atoms) and $E_i = \{(a_u^i, a_v^i)\}$ is the set of edges (bonds). The set of possible labels for the vertices in A corresponds to atom types (e.g. C, O), and the set of possible labels for the edges in S correspond to edge types (single, double, triple, and aromatic). The idea behind forming the *rxn-hypergraph* of a chemical reaction is to efficiently construct new message-passing routes between these disconnected graph components (molecules) and form one connected hypergraph to represent the entire reaction.

To form this hypergraph, we begin by unifying all the G_is into one graph $G = (V, E, A, S)$ where $V = \bigcup V_i^r + \bigcup V_i^p$ and $E = \bigcup E_i^r + \bigcup E_i^p$, while A and S remains the same. For each of the disconnected graph components G_i (molecules), we add a hypernode to the graph as a *mol-hypernode* m_i. Then we add two types of new edges to the G: (1) a set of bidirectional edges connecting every atom to the *mol-hypernode* of their parent molecule, *mol-atom*= $\{(m_i, a_j^i), (a_j^i, m_i)\}$, and (2) a set of bidirectional edges connecting every pair of *mol-hypernodes* on either side of the reaction, *mol-mol*= $\{(m_i, m_j), (m_j, m_i)\}$. The edges of type *mol-mol* would form two fully connected subgraphs between the *mol-hypernodes* on each side of the reaction. We further augment G by adding two more hypernodes as *rxn-hypernodes* x^r and x^p, one for the reactant and one for the product side of the reaction. Then we add a new type of edge to the graph: a set of unidirectional edges from each *mol-hypernode* to the *rxn-hypernode* of the same side of the reaction, *mol-rxn*= $\{(m_i^r, x^r), (m_j^p, x^p)\}$. This augmented version of graph G is what we refer to as the *rxn-hypergraph*. Figure 4.1 shows a schematic drawing of the *rxn-hypergraph*. It can also be represented it as follows:

$$
\begin{aligned}
&\textit{rxn-hypergraph} = (V^*, E^*, A^*, S^*)\\
&\text{where:}\\
&V^* = V \cup \{m_i^r\}_{i=1}^{N} \cup \{m_i^p\}_{i=1}^{M} \cup \{x^r, x^p\},\\
&E^* = E \cup \{(m_i, a_j^i), (a_j^i, m_i)\} \cup \{(m_i^r, m_j^r), (m_j^r, m_i^r)\}\\
&\cup \{(m_i^p, m_j^p), (m_j^p, m_i^p)\} \cup \{(m_i^r, x^r), (m_j^p, x^p)\},\\
&A^* = A \cup \{\textit{mol-hypernode}, \textit{rxn-hypernode}\},\\
&S^* = S \cup \{\textit{atom-mol}, \textit{mol-atom}, \textit{mol-mol}, \textit{mol-rxn}\}.
\end{aligned}
\tag{4.2}
$$

4.3.3 Relational Graph Attention

Now that the rxn-hypergraph is formed, there is an intermolecular path of length three ($a_u^i \rightarrow m_i \rightarrow a_v^j \rightarrow m_j$) between every pair of atoms. Thus, we may construct robust contextual node representations by applying more than three layers of graph convolution neural networks to the *rxn-hypergraph* since this would ensure the receptive field of every atom would include all other atoms. Also, the presence of the unidirectional paths between *mol-hypernodes* and a *rxn-hypernode*, provides a form pooling mechanism where the *rxn-hypernode* can fully represent one side of the reaction. Assuming that both *rxn-hypernodes* carry an abstract representation of the reactants and product molecules, merging them into one vector would represent the entire chemical reaction.

To obtain this representation, we model the entire *rxn-hypergraph* as a relational graph where the relations are represented by S^* in the previous section. We apply and compare two forms of graph neural networks: relational graph convolution (RGCN) [130] and relational graph attention (RGAT) [131, 132]. We summarize the layer-wise operations for each of these graph neural networks.

First, a layer-wise operation of the relational graph convolution can be described as follows:

$$h_i^{l+1} = W_o h_i^l + \frac{1}{|\mathcal{N}_s(i)|}(\sum_{s \in S^*} \sum_{j \in \mathcal{N}_s(i)} W_n h_j^l) \tag{4.3}$$

Where h_i^l is the vector representation of atom a_i and layer l. W_o and W_n are two learnable weight matrices, and $\mathcal{N}_s(i)$ represent the set of nodes adjacent to a_i through edge type s.

The layer-wise operation of our version of relational graph attention and how the attention scores are computed are as follows. The term α_{ij}^s is the amount of attention that node a_i would pay to its neighbor node a_j (through edge type s).

$$h_i^{l+1} = \alpha_{ii} W h_i^l + (\sum_{s \in S^*} \sum_{j \in \mathcal{N}(i)} \alpha_{ij} W h_j^l) \tag{4.4}$$

$$\alpha_{(ij)}^s = \frac{\exp(A(Wh_i || Wh_j))}{\sum_{k \in \mathcal{N}_s(i)} \exp(A(Wh_i || Wh_k))} \tag{4.5}$$

To train a reaction-level predictive model, we apply $L \geq 3$ layers of relational graph attention/convolution to the *rxn-hypergraph*. The final representation of the reaction would be $X = f(x^{r^L}, x^{p^L})$ where f is a selected summarizing function to combine the two sides of the reaction. We use subtraction ($f(x, y) = x - y$) and concatenation ($f(x, y) = x||y$) as the summarizing function. Taking X as the final latent representation of our reaction, we apply a classification or regression head (typically a feed-forward MLP) to perform reaction-level classification or regression. In the Experiments section, we evaluate the viability of our proposed method by performing several reaction-level prediction tasks.

4.3.4 Interpretability

Applying any form of graph neural networks on the *rxn-hypergraph* would provide an interpretable framework using the standard attribution methods such as Integrated gradient [133] and Class Activation Map (CAM) [134]. However, the structure of the *rxn-hypergraph* is capable of providing a more in-depth interpretation. The attention weights obtained from applying GAT can be used as a measure of importance for propagating information relevant to the predictive task. Given that, we define three types of interpretations for the predictions of a GAT model trained using *rxn-hypeergraph*. (1) The *atom-rxn* interpretation: the multiplication of the attention scores (α) of the edges in the path from an atom to the corresponding *rxn-hypernode* (e.g. $a_u^i \rightarrow m_i^r \rightarrow x^r$). These scores can be considered as the contribution of each atom to the final representation of the reaction. (2) The *node-node* interpretation: the average of the attention scores of the edges over all the GAT layers used in architecture. These scores can measure the importance of bonds between adjacent atoms in the final representation of the reaction. Also, the relative importance of the reacting molecules in the final representation of the reaction can be measured by *node-node* scores of the edges between the *mol-hyernodes* and their corresponding *rxn-hypernode* (e.g. $m_i^r \rightarrow x^r$). This score can indicate the difference between reactants and reagents. (3) The intermolecular *atom-atom* interpretation: the multiplication of the attention scores of the edges in the path between pair of atoms of different molecules (e.g. $a_u^i \rightarrow m_i^r \rightarrow m_j^r \rightarrow a_v^j$). These scores can be considered as a measure of the pair-wise correlation between atoms.

4.4 Experiments

To evaluate the viability of the hypergraph representation in various different circumstances, we perform three experiments on different reaction-level predictive tasks using multiple datasets of chemical reactions.

4.4.1 Data

For the first experiment, we use the dataset of chemical reactions from the US patents office (USPTO)[59] to train a model to classify these reactions into the top 50 highly populated classes of chemical reactions. These reactions were classified according to reaction classes presented in [135, 136] and the RSC's RXNO ontology [137] using the NameRxn tool. Similar to the classification experiment presented in [107], we randomly sample 1000 reactions for each of the 50 reaction classes. Then we split each class into a subclass of 200 random reactions for training and 800 for testing. This results in 10,000 training reactions and 40,000 test reactions which are uniformly distributed over 50 classes of chemical reactions.

For the second experiment, we use an in-house curated dataset of reaction mechanisms (reaction with a single transition state). This dataset consists of three classes of mechanistic reactions: (1) over 11,000 polar reactions. The reactions wherein a pair of electrons would transfer from an electron donor orbital to an electron acceptor orbital; (2) 2800 of radical reactions. The reactions that involve a radical species; and (3) 2600 pericyclic reactions. The reactions wherein the transition state has a cyclic geometry. We split this dataset into an 80 percent training dataset and 10 percent validation and 10 percent holdout for final testing.

Lastly, we redo the reaction-level experiment described in [56] as the reaction ranker stage. In this stage, the authors in [56] used a set of over 11,000 productive polar mechanistic reactions to train a ranker system that ranks a reaction mechanism based on its thermodynamic plausibility.

4.4.2 Reaction Representations

For each experiment, we compare the performance of different models that are trained using the corresponding reaction representation. These representations which are introduced in the Related Work section are: (1) *reactionFP* representation based on AP, Morgan2, and TT fingerprints; (2) the representations from the transformer models on reaction SMIRKS, *rxnfp* (both pre-trained and trained from scratch); and (3) representations from the relational graph convolution/attention using *rxn-hypergraph*.

4.4.3 Training and Hyperparameter Optimization

We train the graph attention/convolution layers using either cross-entropy or mean squared error, depending on the task. We use the ADAM optimizer [138] and anneal the learning rate with an exponential schedule across the training duration. Training is performed across 4 GPUs for a period of 500 epochs for each of the experiments explained below.

Additionally, we used SHERPA [2] to optimize the hyperparameters associated with each predictive task, guided by Bayesian Optimization for each parameter. Specifically, we optimized the number of graph layers L, size of the latent representation of nodes D, learning rate, learning rate decay, and $L2$ weight regularization term. The final parameters for each experiment are presented in Table 4.4.

Table 4.1: Comparing testing accuracy of different representations on a test set of 40,000 chemical reactions from the USPTO dataset of chemical reactions.

Representation	Network	Accuracy
reactionFP	AP	0.854
	Morgan2	0.850
	TT	0.852
Transformers	pre-trained *rxnfp*	0.862
	rxnfp	0.925
rxn-hypergraph	RGCN	0.909
	RGAT	**0.928**

4.4.4 Classification of USPTO reactions

The results of this classification experiment are reported in Table 4.1. Both transformer representation (*rxnfp*) and RGAT on *rxn-hypergraph* achieve the highest accuracies. It is important to mention that this classification scheme is highly dependent on the presence of certain molecules and compounds on the reactant side of the reaction. Such textual dependencies are not representing the underlying chemistry of the reaction which implies that a highly accurate model that uses the text representations might not learn the actual chemistry of the reactions and take advantage of the presence of absence of these textual signatures.

4.4.5 Classification of Mechanistic Reactions

Here we classify the reactions into three classes polar, radical, and pericyclic reactions. Since the classification scheme is only based on the pair of reacting orbitals (i.e. single transition states), the underlying chemistry might not be complicated for models to learn. Nevertheless, the graph attentional models on *rxn-hypergraph* are outperforming other models and representations.

For this particular experiment, we also show the interpretations of the final predictions using the three metrics described in the Interpretability section. Figures 4.2, 4.3, and 4.4 are illustrating the interpretation results. In each figure, the reaction with labeled atoms is depicted at the top. The *atom-rxn* scores are shown in the middle while the *node-node* and intermolecular *atom-atom* scores are shown at the bottom right and bottom left of the figures.

4.4.6 Plausibility Ranking of Polar Mechanistic Reactions

This experiment was first introduced in [56], where they rank a set of possible reactions from the interactions between one set of reactant molecules. We precisely follow the procedure of ranking described in [56], we use Siamese network [104, 139] to train ranker models for *reactionFP* and transformer models. In each branch of the Siamese network, we used the hyperparameters used in the previous section.

Ranking Network Architecture

We learn pairwise rankings between different mechanisms using the DirectRanker architecture introduced in [140]. We first compute the learned latent representations of a pair of reactions. Afterward, we use the DirectRanker method of subtracting the two latent representations before feeding them through a bias-free fully-connected layer and a sign-preserving anti-symmetric nonlinearity which produces a scalar value for each pair of events. We train this network using mean squared error to predict which of the two input reactions is more plausible than the other, targets of 1 or -1.

After training this pairwise ranker, we still need a method for determining a final ranked order on a list of plausible mechanisms. To accomplish this, we use the ranked-pair voting

Table 4.2: Comparing testing accuracy of different representations on a test set of 5,455 chemical reactions from an in-house dataset of mechanistic reactions.

Representation	Network	Accuracy
reactionFP	AP	0.915
	Morgan2	0.915
	TT	0.913
Transformers	pre-trained *rxnfp*	0.955
	rxnfp	0.974
rxn-hypergraph	RGCN	0.988
	RGAT	**0.990**

Table 4.3: The top1, 2, 5, and 10 prediction accuracy of plausibility ranking models with different representations of a chemical reaction. All the metrics are computed for a test set of 200 real-world mechanistic reactions.

Representation		top1	top2	top5	top10
reactionFP	AP	58.01	67.05	77.60	84.33
	Morgan2	59.03	69.14	78.24	85.01
	TT	58.41	68.22	77.31	84.14
Transformers	pre-trained *rxnfp*	81.31	84.19	89.60	92.33
	rxnfp	**89.14**	93.22	96.09	98.55
rxn-hypergraph	RGCN	82.67	92.57	97.03	99.01
	RGAT	84.23	**96.06**	**98.57**	**99.28**

method [141]. This algorithm allows us to convert a pair-wise ranking matrix between all viable reaction mechanisms to an ordered list by preference. Ranked pairs construct a directed acyclic graph based on the sorted pair-wise score between different elements. The acyclic property is maintained by ignoring any pairs which would introduce a cycle. After all of the pairs are exhausted, we produce the final ordering by following a topological ordering on the nodes of the graph. The resulting ordering is guaranteed to obey certain criterion which is useful for this task such as independence from irrelevant alternatives, which means that extra implausible reactions that the network is unsure of will not spoil the top rankings.

Out of three experiments, ranking reaction based on thermodynamic plausibility requires a deeper understanding of the underlying chemistry. In this task, the correlation between the

Task	**USPTO Classification**	**Mechanism Classification**	**Polar Plausibility**
Num GAT Layers	10	10	5
Latent Rep Dim	128	128	64
lr Rate	4.11×10^{-4}	4.11×10^{-4}	1.14×10^{-2}
lr Decay	0.999995	0.999995	0.99986
$L2$ Reg	9.75×10^{-5}	9.75×10^{-5}	7.28×10^{-5}

Table 4.4: Table of hyperparameters selected through Bayesian Optimization using SHERPA [2].

textual signatures and plausibility of a reaction is minimal. This potentially explains the results presented in Table 4.3 where the RGAT model on *rxn-hypergraph* outperforms other models, especially those based on the text representations with a considerable margin.

4.5 Discussion and Conclusion

We proposed *rxn-hypergraph* representation of a chemical reaction which is suitable for training graph neural networks for the reaction-level predictive task. The key idea behind forming the *rxn-hypergraph* is to construct efficient message passing routes that provide a platform for (1) updating atom representation based on the atom and molecules if the other reacting molecules, and (2) a global pooling mechanism. *Rxn-hypergraph* is designed to be a universal and permutation-invariant representation that adapts to any downstream predictive task. There are no manual and hand-crafted pre-processing stages involved in computing the *rxn-hypergraph* and it provides different levels of interpretability. There are two potential aspects of this work that are left to future work: (1) several other demanding and practical reaction-level predictive tasks such as yield prediction, and reaction rate constant prediction that can be benefited by *rxn-hypergraph*, and (2) more complicated and expressive attention mechanisms such as multiplicative attentions and transformers can be applied to *rxn-hypergraph* which might result in more powerful reaction representations.

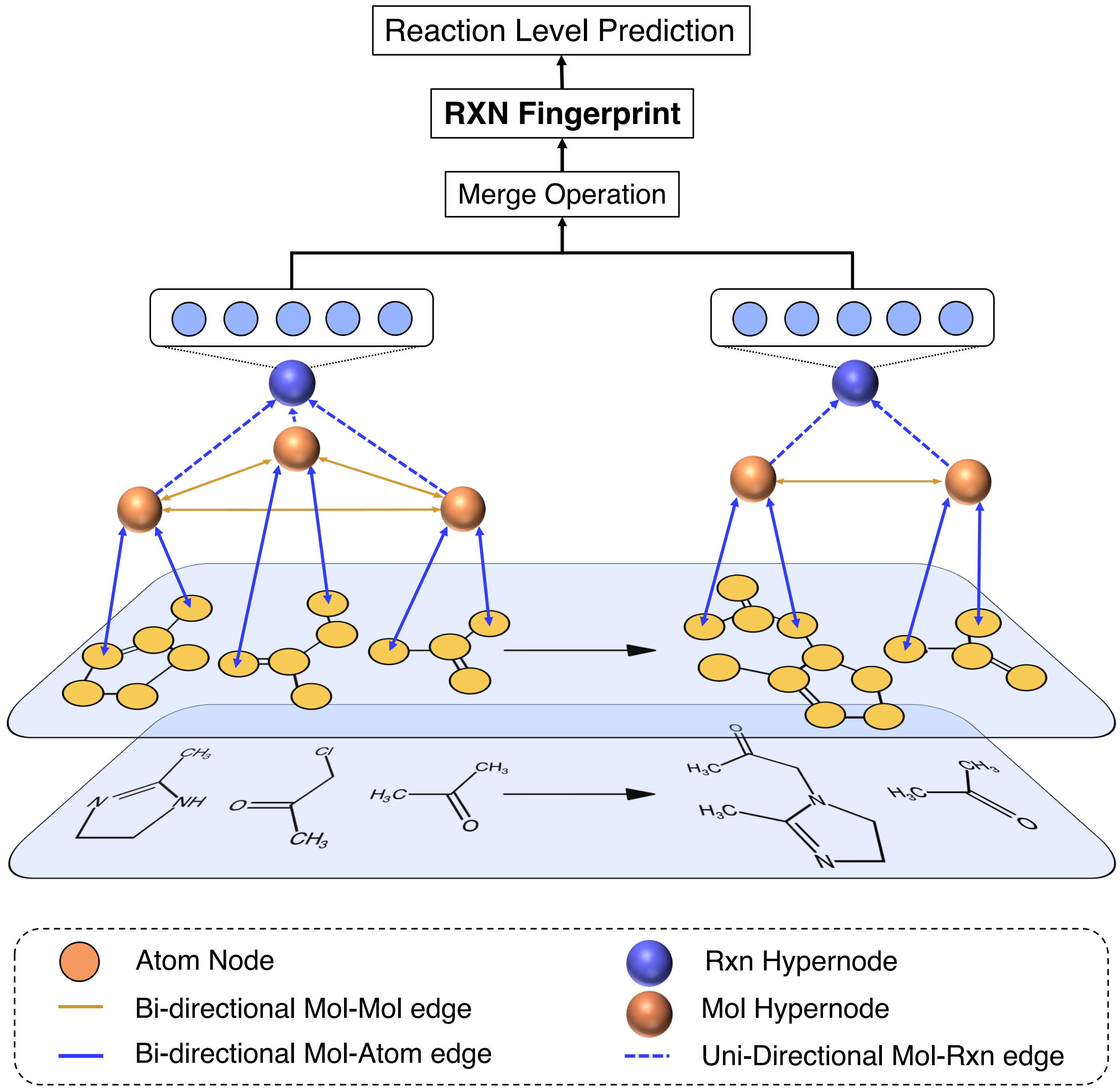

Reaction Level Prediction
RXN Fingerprint
Merge Operation
CH3
N
NH
Cl
O
CH3
H3C
CH3
O
O
H3C
N
H3C
N
H3C
CH3
O
Atom Node
Bi-directional Mol-Mol edge
Bi-directional Mol-Atom edge
Rxn Hypernode
Mol Hypernode
Uni-Directional Mol-Rxn edge

Figure 4.1: A depiction of the *rxn-hypergraph* corresponding to the reaction at the bottom.

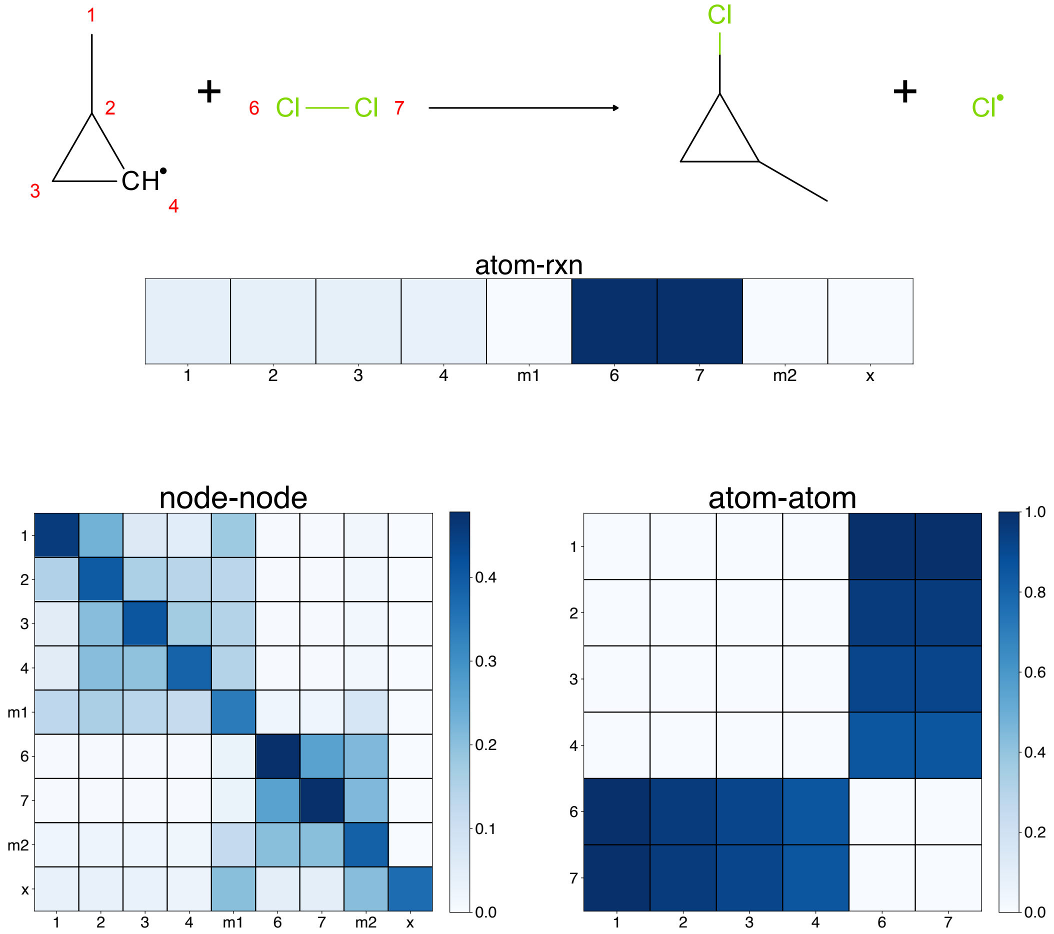


Figure 4.2: Interpretability plots for a radical reaction for the task of classifying the mechanistic reactions.

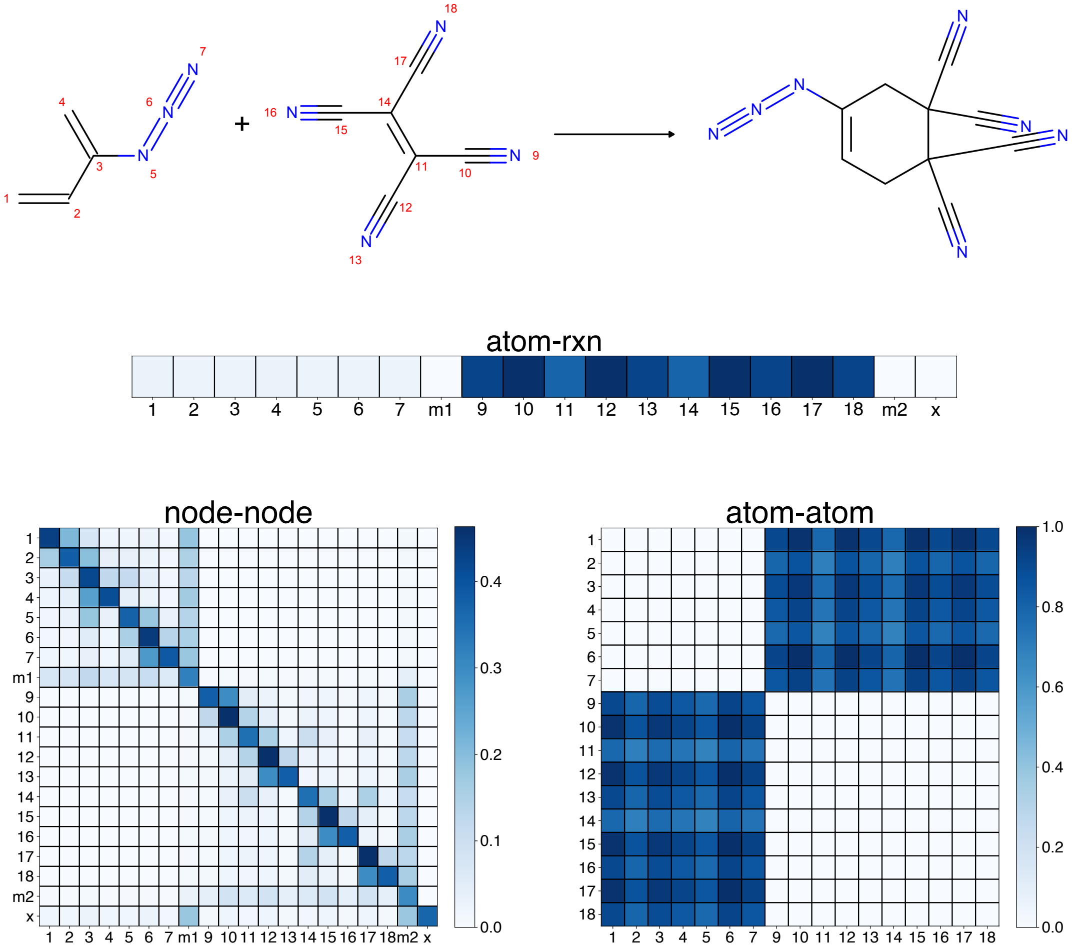


Figure 4.3: Interpretability plots for a pericyclic reaction for the task of classifying the mechanistic reactions.

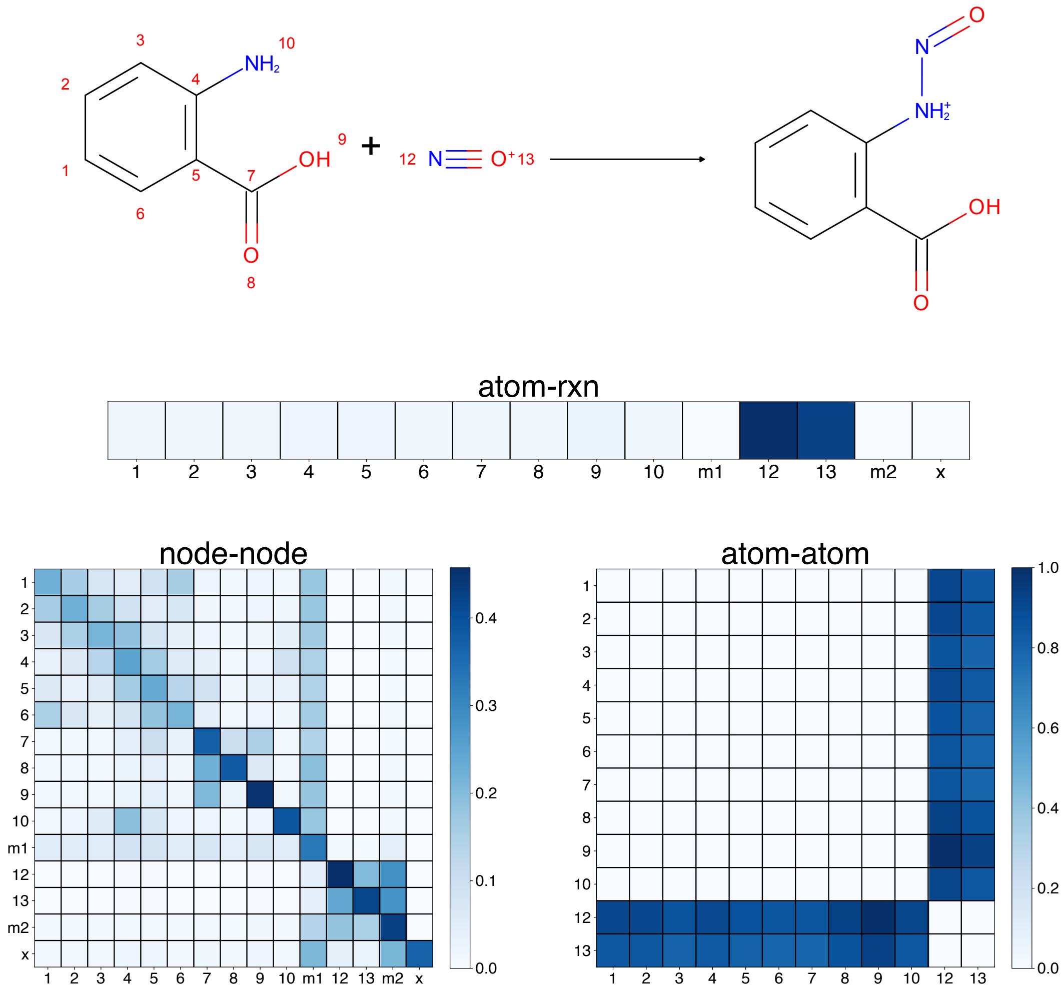


Figure 4.4: Interpretability plots for a polar reaction for the task of classifying the mechanistic reactions.

Chapter 5

Conclusion

The accurate prediction of chemical reaction outcomes poses a significant challenge from both experimental and computational standpoints. Reaction predictor systems that rely on quantum mechanical simulations or hand-crafted rules often suffer from various limitations, including slow or non-robust predictions. However, in the past decade, machine-learning reaction models have emerged as promising solutions, effectively addressing these drawbacks. These models exhibit the capability to generalize their predictions across a wide range of chemical reactions. Moreover, their high-speed inference enables the utilization of high throughput experimentation, facilitating rapid progress in the field. The advancements in machine learning-based reaction prediction have enabled high-impact applications in pharmaceutical, atmospheric, and organic chemistry. Nonetheless, it is important to acknowledge that current machine learning-based reaction predictors still encounter certain limitations. One such limitation is the availability of a limited source of training and development data.

In order to tackle these limitations, our research aimed to introduce a novel approach to reaction prediction. Instead of solely predicting the final outcome of overall transformations, we shifted our focus towards the fundamental components of reactions, known as elementary step reactions. By adopting this perspective, we developed comprehensive and open-access databases known as RMechDB and PMechDB. These databases encompass various types of elementary step reactions, offering a valuable resource for the development of new machine-learning reaction predictors. Specifically, RMechDB focuses on radical chemical reactions, while PMechDB covers polar chemical reactions. Leveraging the availability of our open-access databases, we successfully developed novel reaction predictors specifically tailored for radical reactions, functioning at the level of elementary step reactions. Our newly devised reaction predictor surpasses previous models by offering enhanced capabilities, including chemical interpretability and pathway interpretability. Notably, our predictor ensures the preservation of reaction balance throughout the entire prediction process, maintaining consistency across the reaction chain. To showcase the performance of our reaction predictor, we conducted a comprehensive evaluation benchmark on the RMechDB database, providing insights into its predictive accuracy.

In an effort to contribute to the scientific community, we have made our developed reaction predictor openly accessible to the public. This includes providing access to the databases we curated, as well as the corresponding reaction predictors. To facilitate easy utilization and accessibility, we have hosted these resources on the DeepRXN platform. Interested users can access the platform and explore our reaction predictors and databases by visiting the following link: `https://deeprxn.ics.uci.edu/`. By making these tools available, we aim to encourage collaboration, further research, and advancement in the field of reaction prediction.

Bibliography

[1] Mohammadamin Tavakoli, Yin Ting T Chiu, Pierre Baldi, Ann Marie Carlton, and David Van Vranken. Rmechdb: A public database of elementary radical reaction steps. *Journal of Chemical Information and Modeling*.

[2] Lars Hertel, Julian Collado, Peter Sadowski, Jordan Ott, and Pierre Baldi. Sherpa: Robust hyperparameter optimization for machine learning. *SoftwareX*, 12:100591, 2020.

[3] Peter Sadowski, David Fooshee, Niranjan Subrahmanya, and Pierre Baldi. Synergies between quantum mechanics and machine learning in reaction prediction. *Journal of Chemical Information and Modeling*, 56(11):2125–2128, 2016.

[4] Aaron Mood, Mohammadamin Tavakoli, Eugene Gutman, Dora Kadish, Pierre Baldi, and David L Van Vranken. Methyl anion affinities of the canonical organic functional groups. *J. Org. Chem.*, 85(6):4096–4102, 2020.

[5] Dora Kadish, Aaron D Mood, Mohammadamin Tavakoli, Eugene S Gutman, Pierre Baldi, and David L Van Vranken. Methyl cation affinities of canonical organic functional groups. *J. Org. Chem.*, 2021.

[6] J. Chen and P. Baldi. Synthesis explorer: Organic chemistry tutorial system for multi-step synthesis design and reaction prediction. *Journal of Chemical Education*, 85(12):1699–1703, 2008.

[7] J. Chen and P. Baldi. No electron left-behind: a rule-based expert system to predict chemical reactions and reaction mechanisms. *Journal of Chemical Information and Modeling*, 49(9):2034–2043, 2009.

[8] Greg Landrum. Rdkit: Open-source cheminformatics software. 2016.

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

[10] James Law, Zsolt Zsoldos, Aniko Simon, Darryl Reid, Yang Liu, Sing Yoong Khew, A Peter Johnson, Sarah Major, Robert A Wade, and Howard Y Ando. Route designer: a retrosynthetic analysis tool utilizing automated retrosynthetic rule generation. *Journal of chemical information and modeling*, 49(3):593–602, 2009.

[11] Connor W Coley, William H Green, and Klavs F Jensen. Rdchiral: An rdkit wrapper for handling stereochemistry in retrosynthetic template extraction and application. *Journal of chemical information and modeling*, 59(6):2529–2537, 2019.

[12] David R Fooshee, Tran B Nguyen, Sergey A Nizkorodov, Julia Laskin, Alexander Laskin, and Pierre Baldi. Cobra: A computational brewing application for predicting the molecular composition of organic aerosols. *Environmental science & technology*, 46(11):6048–6055, 2012.

[13] David R Fooshee, Paige K Aiona, Alexander Laskin, Julia Laskin, Sergey A Nizkorodov, and Pierre F Baldi. Atmospheric oxidation of squalene: molecular study using cobra modeling and high-resolution mass spectrometry. *Environmental science & technology*, 49(22):13304–13313, 2015.

[14] David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *J. Chem. Inf. Model.*, 50(5):742–754, 2010.

[15] Sara Szymkuć, Ewa P Gajewska, Tomasz Klucznik, Karol Molga, Piotr Dittwald, Michał Startek, Michał Bajczyk, and Bartosz A Grzybowski. Computer-assisted synthetic planning: the end of the beginning. *Angewandte Chemie International Edition*, 55(20):5904–5937, 2016.

[16] Shuan Chen and Yousung Jung. Deep retrosynthetic reaction prediction using local reactivity and global attention. *JACS Au*, 1(10):1612–1620, 2021.

[17] Vignesh Ram Somnath, Charlotte Bunne, Connor Coley, Andreas Krause, and Regina Barzilay. Learning graph models for retrosynthesis prediction. *Advances in Neural Information Processing Systems*, 34:9405–9415, 2021.

[18] Mikołaj Sacha, Mikołaj Błaz, Piotr Byrski, Paweł Dabrowski-Tumanski, Mikołaj Chrominski, Rafał Loska, Paweł Włodarczyk-Pruszynski, and Stanisław Jastrzebski. Molecule edit graph attention network: modeling chemical reactions as sequences of graph edits. *Journal of Chemical Information and Modeling*, 61(7):3273–3284, 2021.

[19] Bhuvanesh Sridharan, Manan Goel, and U Deva Priyakumar. Modern machine learning for tackling inverse problems in chemistry: molecular design to realization. *Chemical Communications*, 58(35):5316–5331, 2022.

[20] Pierre Baldi. Call for a public open database of all chemical reactions. *Journal of Chemical Information and Modeling*.

[21] Zhengkai Tu and Connor W Coley. Permutation invariant graph-to-sequence model for template-free retrosynthesis and reaction prediction. *Journal of chemical information and modeling*, 62(15):3503–3513, 2022.

[22] Kelong Mao, Xi Xiao, Tingyang Xu, Yu Rong, Junzhou Huang, and Peilin Zhao. Molecular graph enhanced transformer for retrosynthesis prediction. *Neurocomputing*, 457:193–202, 2021.

[23] Mohammadamin Tavakoli and Pierre Baldi. Continuous representation of molecules using graph variational autoencoder. *arXiv preprint arXiv:2004.08152*, 2020.

[24] Connor W Coley, Wengong Jin, Luke Rogers, Timothy F Jamison, Tommi S Jaakkola, William H Green, Regina Barzilay, and Klavs F Jensen. A graph-convolutional neural network model for the prediction of chemical reactivity. *Chem. Sci.*, 10(2):370–377, 2019.

[25] Alessandro Lusci, Gianluca Pollastri, and Pierre Baldi. Deep architectures and deep learning in chemoinformatics: the prediction of aqueous solubility for drug-like molecules. *Journal of Chemical Information and Modeling*, 53(7):1563–1575, 2013.

[26] Gregor Urban, Niranjan Subrahmanya, and Pierre Baldi. Inner and outer recursive neural networks for chemoinformatics applications. *Journal of chemical information and modeling*, 58(2):207–211, 2018.

[27] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. *arXiv preprint arXiv:1704.01212*, 2017.

[28] Mohammadamin Tavakoli, Aaron Mood, David Van Vranken, and Pierre Baldi. Quantum mechanics and machine learning synergies: Graph attention neural networks to predict chemical reactivity. *arXiv preprint arXiv:2103.14536*, 2021.

[29] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, Yoshua Bengio, et al. Graph attention networks. *stat*, 1050(20):10–48550, 2017.

[30] Connor W Coley, Regina Barzilay, Tommi S Jaakkola, William H Green, and Klavs F Jensen. Prediction of organic reaction outcomes using machine learning. *ACS Cent. Sci.*, 3(5):434–443, 2017.

[31] Nicola De Cao and Thomas Kipf. Molgan: An implicit generative model for small molecular graphs. *arXiv preprint arXiv:1805.11973*, 2018.

[32] David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.*, 28(1):31–36, 1988.

[33] Qingyi Yang, Vishnu Sresht, Peter Bolgar, Xinjun Hou, Jacquelyn L Klug-McLeod, Christopher R Butler, et al. Molecular transformer unifies reaction prediction and retrosynthesis across pharma chemical space. *Chemical communications*, 55(81):12152–12155, 2019.

[34] Kangjie Lin, Youjun Xu, Jianfeng Pei, and Luhua Lai. Automatic retrosynthetic route planning using template-free models. *Chemical science*, 11(12):3355–3364, 2020.

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

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

[37] Mia Xu Chen, Orhan Firat, Ankur Bapna, Melvin Johnson, Wolfgang Macherey, George Foster, Llion Jones, Niki Parmar, Mike Schuster, Zhifeng Chen, et al. The best of both worlds: Combining recent advances in neural machine translation. *arXiv preprint arXiv:1804.09849*, 2018.

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

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

[40] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[41] Liliana C Gallegos, Guilian Luchini, Peter C St. John, Seonah Kim, and Robert S Paton. Importance of engineered and learned molecular representations in predicting organic reactivity, selectivity, and chemical properties. *Accounts of Chemical Research*, 54(4):827–836, 2021.

[42] Mohammadamin Tavakoli, Alexander Shmakov, Francesco Ceccarelli, and Pierre Baldi. Rxn hypergraph: a hypergraph attention model for chemical reaction representation. *arXiv preprint arXiv:2201.01196*, 2022.

[43] Daniel Lowe. Chemical reactions from us patents (1976-sep2016).

[44] Derek T Ahneman, Jesús G Estrada, Shishi Lin, Spencer D Dreher, and Abigail G Doyle. Predicting reaction performance in c–n cross-coupling using machine learning. *Science*, 360(6385):186–190, 2018.

[45] Damith Perera, Joseph W Tucker, Shalini Brahmbhatt, Christopher J Helal, Ashley Chong, William Farrell, Paul Richardson, and Neal W Sach. A platform for automated nanomole-scale reaction screening and micromole-scale synthesis in flow. *Science*, 359(6374):429–434, 2018.

[46] Dennis P Curran, Ned A Porter, and Bernd Giese. *Stereochemistry of radical reactions: concepts, guidelines, and synthetic applications.* John Wiley & Sons, 2008.

[47] Muthyala Ramaiah. Radical reactions in organic synthesis. *Tetrahedron*, 43(16):3541–3676, 1987.

[48] János Fehér, Géza Csomós, and András Vereckei. *Free radical reactions in medicine.* Springer Science & Business Media, 2012.

[49] Bok Nam Jang, Marius Costache, and Charles A Wilkie. The relationship between thermal degradation behavior of polymer and the fire retardancy of polymer/clay nanocomposites. *Polymer*, 46(24):10678–10687, 2005.

[50] Michel Le Bras, Serge Bourbigot, Christelle Delporte, Catherine Siat, and Yannick Le Tallec. New intumescent formulations of fire-retardant polypropylene—discussion of the free radical mechanism of the formation of carbonaceous protective material during the thermo-oxidative treatment of the additives. *Fire and materials*, 20(4):191–203, 1996.

[51] Marta I Litter. Introduction to photochemical advanced oxidation processes for water treatment. *Environmental photochemistry part II*, pages 325–366, 2005.

[52] Wenshuai Zhu, Chao Wang, Hongping Li, Peiwen Wu, Suhang Xun, Wei Jiang, Zhigang Chen, Zhen Zhao, and Huaming Li. One-pot extraction combined with metal-free photochemical aerobic oxidative desulfurization in deep eutectic solvent. *Green Chemistry*, 17(4):2464–2472, 2015.

[53] Philippe Schwaller, Alain C Vaucher, Teodoro Laino, and Jean-Louis Reymond. Prediction of chemical reaction yields using deep learning. *Mach. Learn.: Sci. Technol.*, 2(1):015016, 2021.

[54] Mohammadamin Tavakoli, Aaron Mood, David Van Vranken, and Pierre Baldi. Quantum mechanics and machine learning synergies: graph attention neural networks to predict chemical reactivity. *Journal of Chemical Information and Modeling*, 62(9):2121–2132, 2022.

[55] Dora Kadish, Aaron D Mood, Mohammadamin Tavakoli, Eugene S Gutman, Pierre Baldi, and David L Van Vranken. Methyl cation affinities of canonical organic functional groups. *The Journal of Organic Chemistry*, 86(5):3721–3729, 2021.

[56] David Fooshee, Aaron Mood, Eugene Gutman, Mohammadamin Tavakoli, Gregor Urban, Frances Liu, Nancy Huynh, David Van Vranken, and Pierre Baldi. Deep learning for chemical reaction prediction. *Mol. Syst. Des. Eng.*, 2018.

[57] Philippe Schwaller, Theophile Gaudin, David Lanyi, Costas Bekas, and Teodoro Laino. “found in translation”: predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. *Chemical science*, 9(28):6091–6098, 2018.

[58] Pierre Baldi. *Deep learning in science*. Cambridge University Press, 2021.

[59] Daniel Mark Lowe. *Extraction of chemical structures and reactions from the literature*. PhD thesis, University of Cambridge, 2012.

[60] S Das, Sabita Patel, and Bijay K Mishra. Oxidation by permanganate: synthetic and mechanistic aspects. 2009.

[61] Matthew A. Kayala and Pierre Baldi. Reactionpredictor: Prediction of complex chemical reactions at the mechanistic level using machine learning. *Journal of Chemical Information and Modeling*, 52(10):2526–2540, 2012. PMID: 22978639.

[62] I Fleming. *Frontier Orbitals and Organic Chemical Reactions*. Wiley, 1977.

[63] Kristina Preuer, Günter Klambauer, Friedrich Rippmann, Sepp Hochreiter, and Thomas Unterthiner. Interpretable deep learning in drug discovery. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pages 331–345. Springer, 2019.

[64] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296*, 2017.

[65] Mengjie Liu, Alon Grinberg Dana, Matthew S Johnson, Mark J Goldman, Agnes Jocher, A Mark Payne, Colin A Grambow, Kehang Han, Nathan W Yee, and Emily J Mazeau. Reaction mechanism generator v3. 0: advances in automatic mechanism generation. *Journal of Chemical Information and Modeling*, 61(6):2686–2696, 2021.

[66] ME Jenkin, JC Young, and AR Rickard. The mcm v3. 3.1 degradation scheme for isoprene. *Atmospheric Chemistry and Physics*, 15(20):11433–11459, 2015.

[67] Matthew A. Kayala, Chloé-Agathe Azencott, Jonathan H. Chen, and Pierre Baldi. Learning to predict chemical reactions. *Journal of Chemical Information and Modeling*, 51(9):2209–2222, 2011. PMID: 21819139.

[68] Steven M Kearnes, Michael R Maser, Michael Wleklinski, Anton Kast, Abigail G Doyle, Spencer D Dreher, Joel M Hawkins, Klavs F Jensen, and Connor W Coley. The open reaction database. *Journal of the American Chemical Society*, 143(45):18820–18826, 2021.

[69] Connor W. Coley, Regina Barzilay, Tommi S. Jaakkola, William H. Green, and Klavs F. Jensen. Prediction of organic reaction outcomes using machine learning. *ACS Central Science*, 3(5):434–443, 2017. PMID: 28573205.

[70] O'Hagan and Lloyd. The iconic curly arrow.

[71] William Ogilvy Kermack and Robert Robinson. Li.—an explanation of the property of induced polarity of atoms and an interpretation of the theory of partial valencies on an electronic basis. *J. Chem. Soc., Trans.*, 121:427–440, 1922.

[72] John T Herron and David S Green. Chemical kinetics database and predictive schemes for nonthermal humid air plasma chemistry. part ii. neutral species reactions. *Plasma chemistry and plasma processing*, 21(3):459–481, 2001.

[73] António J. M. Ribeiro, Gemma L. Holliday, Nicholas Furnham, Jonathan D. Tyzack, Katherine Ferris, and Janet M. Thornton. Mechanism and catalytic site atlas (m-csa): a database of enzyme reaction mechanisms and active sites. *Nucleic Acids Research*, 46:D618 – D623, 2018.

[74] Michael E. Jenkin, Sandra M. Saunders, and Michael J. Pilling. The tropospheric degradation of volatile organic compounds: a protocol for mechanism development. *Atmospheric Environment*, 31(1):81–104, 1997.

[75] S. M. Saunders, M. E. Jenkin, R. G. Derwent, and M. J. Pilling. Protocol for the development of the master chemical mechanism, mcm v3 (part a): tropospheric degradation of non-aromatic volatile organic compounds. *Atmospheric Chemistry and Physics*, 3(1):161–180, 2003.

[76] M. E. Jenkin, S. M. Saunders, V. Wagner, and M. J. Pilling. Protocol for the development of the master chemical mechanism, mcm v3 (part b): tropospheric degradation of aromatic volatile organic compounds. *Atmospheric Chemistry and Physics*, 3(1):181–193, 2003.

[77] C. Bloss, V. Wagner, M. E. Jenkin, R. Volkamer, W. J. Bloss, J. D. Lee, D. E. Heard, K. Wirtz, M. Martin-Reviejo, G. Rea, J. C. Wenger, and M. J. Pilling. Development of a detailed chemical mechanism (mcmv3.1) for the atmospheric oxidation of aromatic hydrocarbons. *Atmospheric Chemistry and Physics*, 5(3):641–664, 2005.

[78] M. E. Jenkin, K. P. Wyche, C. J. Evans, T. Carr, P. S. Monks, M. R. Alfarra, M. H. Barley, G. B. McFiggans, J. C. Young, and A. R. Rickard. Development and chamber evaluation of the mcm v3.2 degradation scheme for -caryophyllene. *Atmospheric Chemistry and Physics*, 12(11):5275–5308, 2012.

[79] M. E. Jenkin, J. C. Young, and A. R. Rickard. The mcm v3.3.1 degradation scheme for isoprene. *Atmospheric Chemistry and Physics*, 15(20):11433–11459, 2015.

[80] Allen Buskirk and Hediyeh Baradaran. Can reaction mechanisms be proven? *Journal of Chemical Education*, 86(5):551, 2009.

[81] Brown, Foote, Iverson, and Anslyn. *Organic Chemistry, 5th Ed.* Brooks-Cole, 2008.

[82] S. Ege, Kleinman R. W., and P. Zitek. *Organic Chemistry, Structure and Reactivity.* Cengage Learning, Mifflin Company, 2004.

[83] M. Loudon and J. Parise. *Organic Chemistry 6th Ed.* W. H. Freeman, 2015.

[84] J. E. McMurry. *Organic Chemistry with Biological Applications.* Cengage Learning, 2014.

[85] J. Smith. *Organic Chemistry 5th Ed.* McGraw Hill, 2016.

[86] T. W. Solomons and C. B Fryhle. *Organic Chemistry 11th Ed.* Wiley, 2013.

[87] P. Vollhardt. *Organic Chemistry Structure and Function.* W. H. Freeman, 2005.

[88] T. W. Solomons and C. B Fryhle. *Organic Chemistry 8th Ed.* Pearson, 2012.

[89] F. A. Carey and R. J. Sundberg. *Advanced Organic Chemistry Part B: Reactions and Synthesis, 5th Ed.* Springer, 2010.

[90] R. Bruckner. *Organic Mechanisms: Reactions, Stereochemistry and Synthesis.* Springer, 2010.

[91] J.H. Seinfeld and S. N. Pandis. *Atmospheric Chemistry and Physics: From Air Pollution to Climate Change, 2nd Ed.* Wiley, 2016.

[92] Paul O. Wennberg, Kelvin H. Bates, John D. Crounse, Leah G. Dodson, Renee C. McVay, Laura A. Mertens, Tran B. Nguyen, Eric Praske, Rebecca H. Schwantes, Matthew D. Smarte, Jason M. St Clair, Alexander P. Teng, Xuan Zhang, and John H. Seinfeld. Gas-phase reactions of isoprene and its major oxidation products. *Chemical Reviews*, 118(7):3337–3390, 2018. PMID: 29522327.

[93] Wai-To Chan and Ian Hamilton. Mechanisms for the ozonolysis of ethene and propene: Reliability of quantum chemical predictions. *The Journal of Chemical Physics*, 118:1688–1701, 01 2003.

[94] Stefan Simkovics and Paul Petersgasse. *Enhancement of the ANSI SQL Implementation of PostgreSQL.* na, 1998.

[95] Raghu Ramakrishnan, Johannes Gehrke, and Johannes Gehrke. *Database management systems*, volume 3. McGraw-Hill New York, 2003.

[96] Openeye-Scientific-Software. http://www.eyesopen.com. *Inc., Santa Fe, NM, USA*, 2022.

[97] TK OEChem. Openeye scientific software. *Inc., Santa Fe, NM, USA*, 2022.

[98] TK OEDepict. Openeye scientific software. *Inc., Santa Fe, NM, USA*, 2022.

[99] TK GraphSim. Openeye scientific software. *Inc., Santa Fe, NM, USA*, 2022.

[100] Marvin. Chemaxon. *http://www.chemaxon.com*, 2019.

[101] Matthew A Kayala, Chloé-Agathe Azencott, Jonathan H Chen, and Pierre Baldi. Learning to predict chemical reactions. *Journal of chemical information and modeling*, 51(9):2209–2222, 2011.

[102] Bowen Liu, Bharath Ramsundar, Prasad Kawthekar, Jade Shi, Joseph Gomes, Quang Luu Nguyen, Stephen Ho, Jack Sloane, Paul Wender, and Vijay Pande. Retrosynthetic reaction prediction using neural sequence-to-sequence models. *ACS Cent. Sci.*, 3(10):1103–1113, 2017.

[103] Matthew A Kayala and Pierre Baldi. Reactionpredictor: prediction of complex chemical reactions at the mechanistic level using machine learning. *Journal of chemical information and modeling*, 52(10):2526–2540, 2012.

[104] Jane Bromley, James W Bentz, Léon Bottou, Isabelle Guyon, Yann LeCun, Cliff Moore, Eduard Säckinger, and Roopak Shah. Signature verification using a "siamese" time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(04):669–688, 1993.

[105] P. Baldi and Y. Chauvin. Neural networks for fingerprint recognition. *Neural Computation*, 5(3):402–418, 1993.

[106] Matthew A. Kayala and Pierre F. Baldi. A machine learning approach to predict chemical reactions. In J. Shawe-Taylor, R.S. Zemel, P.L. Bartlett, F. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 747–755. Curran Associates, Inc., 2011.

[107] Nadine Schneider, Daniel M Lowe, Roger A Sayle, and Gregory A Landrum. Development of a novel fingerprint for chemical reactions and its application to large-scale reaction classification and similarity. *Journal of chemical information and modeling*, 55(1):39–53, 2015.

[108] Raymond E Carhart, Dennis H Smith, and R Venkataraghavan. Atom pairs as molecular features in structure-activity studies: definition and applications. *Journal of Chemical Information and Computer Sciences*, 25(2):64–73, 1985.

[109] Ramaswamy Nilakantan, Norman Bauman, J Scott Dixon, and R Venkataraghavan. Topological torsion: a new molecular descriptor for sar applications. comparison with other descriptors. *Journal of Chemical Information and Computer Sciences*, 27(2):82–85, 1987.

[110] Daniel Probst, Philippe Schwaller, and Jean-Louis Reymond. Reaction classification and yield prediction using the differential reaction fingerprint drfp. *Digital discovery*, 1(2):91–97, 2022.

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

[112] Pavel Karpov, Guillaume Godin, and Igor V Tetko. A transformer model for retrosynthesis. In *Artificial Neural Networks and Machine Learning–ICANN 2019: Workshop and Special Sessions: 28th International Conference on Artificial Neural Networks, Munich, Germany, September 17–19, 2019, Proceedings*, pages 817–830. Springer, 2019.

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

[114] Wengong Jin, Connor Coley, Regina Barzilay, and Tommi Jaakkola. Predicting organic reaction outcomes with weisfeiler-lehman network. *Advances in neural information processing systems*, 30, 2017.

[115] Philippe Schwaller, Daniel Probst, Alain C Vaucher, Vishnu H Nair, David Kreutter, Teodoro Laino, and Jean-Louis Reymond. Mapping the space of chemical reactions

using attention-based neural networks. *Nature Machine Intelligence*, 3(2):144–152, 2021.

[116] Dragomir R Radev, Hong Qi, Harris Wu, and Weiguo Fan. Evaluating web-based question answering systems. In *LREC*. Citeseer, 2002.

[117] Paul O Wennberg, Kelvin H Bates, John D Crounse, Leah G Dodson, Renee C McVay, Laura A Mertens, Tran B Nguyen, Eric Praske, Rebecca H Schwantes, Matthew D Smarte, et al. Gas-phase reactions of isoprene and its major oxidation products. *Chemical reviews*, 118(7):3337–3390, 2018.

[118] RDKit: Open-source cheminformatics. `http://www.rdkit.org`. [Online; accessed 11-April-2013].

[119] J Bredt, Jos Houben, and Paul Levy. Ueber isomere dehydrocamphersäuren, lauronolsäuren und bihydrolauro-lactone. *Berichte der deutschen chemischen Gesellschaft*, 35(2):1286–1292, 1902.

[120] Philippe Schwaller, Riccardo Petraglia, Valerio Zullo, Vishnu H Nair, Rico Andreas Haeuselmann, Riccardo Pisoni, Costas Bekas, Anna Iuliano, and Teodoro Laino. Predicting retrosynthetic pathways using transformer-based models and a hyper-graph exploration strategy. *Chemical science*, 11(12):3316–3325, 2020.

[121] Kevin Yang, Kyle Swanson, Wengong Jin, Connor Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian Kelley, Miriam Mathea, et al. Analyzing learned molecular representations for property prediction. *Journal of chemical information and modeling*, 59(8):3370–3388, 2019.

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

[123] Robert C Glen, Andreas Bender, Catrin H Arnby, Lars Carlsson, Scott Boyer, and James Smith. Circular fingerprints: flexible molecular descriptors with applications from physical chemistry to adme. *IDrugs*, 9(3):199, 2006.

[124] Philippe Schwaller, Benjamin Hoover, Jean-Louis Reymond, Hendrik Strobelt, and Teodoro Laino. Extraction of organic chemistry grammar from unsupervised learning of chemical reactions. *Science Advances*, 7(15):eabe4166, 2021.

[125] Giorgio Pesciullesi, Philippe Schwaller, Teodoro Laino, and Jean-Louis Reymond. Transfer learning enables the molecular transformer to predict regio-and stereoselective reactions on carbohydrates. *Nature communications*, 11(1):1–8, 2020.

[126] David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. In *Adv. Neural Inf. Process. Syst*, pages 2224–2232, 2015.

[127] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *Euro. Sem. Web. Conf.*, pages 593–607. Springer, 2018.

[128] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[129] Oliver Wieder, Stefan Kohlbacher, Mélaine Kuenemann, Arthur Garon, Pierre Ducrot, Thomas Seidel, and Thierry Langer. A compact review of molecular property prediction with graph neural networks. *Drug Discovery Today: Technologies*, 2020.

[130] Michael Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In Aldo Gangemi, Roberto Navigli, Maria-Esther Vidal, Pascal Hitzler, Raphaël Troncy, Laura Hollink, Anna Tordai, and Mehwish Alam, editors, *The Semantic Web*, pages 593–607, Cham, 2018. Springer International Publishing.

[131] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks, 2018.

[132] Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks?, 2021.

[133] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR, 2017.

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

[135] John S Carey, David Laffan, Colin Thomson, and Mike T Williams. Analysis of the reactions used for the preparation of drug candidate molecules. *Organic & biomolecular chemistry*, 4(12):2337–2347, 2006.

[136] Stephen D Roughley and Allan M Jordan. The medicinal chemist's toolbox: an analysis of reactions used in the pursuit of drug candidates. *Journal of medicinal chemistry*, 54(10):3451–3479, 2011.

[137] Hans Kraut, Josef Eiblmaier, Guenter Grethe, Peter low, Heinz Matuszczyk, and Heinz Saller. Algorithm for reaction classification. *Journal of chemical information and modeling*, 53(11):2884–2895, 2013.

[138] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[139] Davide Chicco. Siamese neural networks: An overview. *Artificial Neural Networks*, pages 73–94, 2021.

[140] Marius Köppel, Alexander Segner, Martin Wagener, Lukas Pensel, Andreas Karwath, and Stefan Kramer. Pairwise learning to rank by neural networks revisited: Reconstruction, theoretical analysis and practical performance. In Ulf Brefeld, Elisa Fromont, Andreas Hotho, Arno Knobbe, Marloes Maathuis, and Céline Robardet, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 237–252, Cham, 2020. Springer International Publishing.

[141] T. N. Tideman. Independence of clones as a criterion for voting rules. *Social Choice and Welfare*, 4(3):185–206, 1987.