Data-efficient computational strategies for tackling biological problems involving protein-ligand co-design, site-of-metabolism prediction, and biosynthetic gene cluster product classification

A dissertation submitted by

Vladimir Porokhin

in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in
Computer Science
Tufts University
February 2026

Adviser: Soha Hassoun

Abstract

Data has been a scarce resource in bioinformatics since its early beginnings. Biological systems are varied, so the fit between the available datasets and the problem at hand is often not perfect. Moreover, due to the inherent complexity of such systems, adapting datasets and methods is a non-trivial task. As such, approaching biological problems from a computational perspective requires a "toolbox" of data-efficient strategies and the ability to customize them. In this thesis, we explore three such problems in detail.

First, we propose the paradigm of simultaneous co-design of proteins and ligands for enhancement of binding affinity in a novel bioelectronic sensing application, contrasting it with the traditional approach of individual design. This new paradigm promises better design outcomes; however, executing it in practice is challenging due to lack of information about the particular protein-ligand interaction and the sheer number of design possibilities. We tackle this problem through advanced structure modeling techniques to gain a better understanding of the interaction and permit efficient exploration of the design space.

Next, we introduce *GNN-SOM*, a Graph Neural Network for Site-Of-Metabolism prediction in enzyme-mediated reactions. Unlike chemical reactions, enzymatic interactions have not yet been extensively cataloged, with existing enzymatic datasets being orders of magnitude smaller than typical molecular datasets. To address this constraint, we select a model architecture uniquely suited to this application and pair it with preprocessing steps to enhance data quality. We show that our approach outperforms baseline methods and demonstrate two downstream biological applications that benefit from SOM prediction.

Finally, we present *BGCat*, a deep neural network for predicting detailed biosynthetic gene cluster (BGC) product types. Unfortunately, there is a severe lack of data associating BGCs with their products, with the largest database listing around 3,000 BGCs. To allow effective learning on such a small dataset, we use a protein language model pretrained on millions of sequences alongside a custom data augmentation scheme. The resulting model

offers better performance compared to existing methods and provides novel insights into families of BGCs. Overall, these three problems demonstrate the challenges of working with biological systems and a persistent need for data-efficient techniques.

Acknowledgments

I would first like to express my deepest gratitude to my advisor, Dr. Soha Hassoun, for her incredible patience and support. She was the one who encouraged me to pursue research when I started my journey at Tufts as an undergraduate student 10 years ago. Thank you Soha – I could not have done this without your help and mentorship.

I would also like to thank my committee members for their guidance and feedback on my work. I am grateful to Dr. Nik Nair for his expertise in systems biology. I thank Dr. Liping Liu for introducing me to machine learning and graph neural networks. I am also grateful to Dr. Fahad Dogar for teaching me efficient programming techniques. I also thank Dr. Anne Brown for her deep understanding of bioinformatics and simulation methods.

I also thank my research colleagues and students in the Hassoun Lab I met over the years, they made this journey very enjoyable and I have benefitted tremendously from their diverse perspectives and skill sets.

Last, but not least, I would like to thank my parents and siblings for their continuous support and encouragement in all of my pursuits.

This work was sponsored in part by Army Research Office (MURI program, contract DOD ARO #W911NF2210239), National Science Foundation (Award [CCF-1909536]), and by National Institute of General Medical Sciences of the National Institutes of Health (Award [R01GM132391]). The content is solely the responsibility of the author and does not necessarily represent the official views of any entity.

Table of Contents

List of Tables

List of Figures

Chapter 1 - Introduction

Since the term was first coined by Hesper and Hogeweg in the early 1970s [1], bioinformatics has been at the forefront of leveraging computational methods for solving difficult problems in biology. The early years of the field were plagued by data scarcity and difficulty of exchanging information, leading many practitioners to focus their efforts on pattern analysis and modeling methods. However, these limitations soon subsided with the introduction of genomic sequencing and other large-scale "omics" data – or so it appeared at first. Biology stands apart from other natural sciences in its lack of determinism and resistance to reduction [2]. Many features and behaviors of biological systems are emergent from complex interactions and cannot be explained by the functions of their individual components, and with many biological phenomena driven by chance, there are no universal natural laws to fall back onto. Every biological application is thus unique and demands holistic consideration, which often limits applicability of existing data and puts the issue of scarcity back into the spotlight for many problems in the field.

This thesis presents solutions to three such problems. The first is computational design of proteins and their small molecule (ligand) binders with the goal of enhancing their binding affinity. Our approach aims to co-design both the protein and the ligand simultaneously, which is a significant departure from the traditional paradigm of individual design where only the protein or only the ligand is allowed to change. Enhancement of binding affinity has a unique application for bioelectronic sensing; however, the data about the particular

protein-ligand combination is extremely limited. Furthermore, allowing variation in both the protein and the ligand increases the space of potential combinations dramatically, making it unfeasible to fully explore. We solve the first issue by using advanced structure prediction, docking, and simulation methods to construct a model of the protein binding pocket and its interaction with the ligand. We then use this model to solve the second issue by constraining the set of design choices only to protein mutations and ligand modifications that are likely to alter the binding affinity. We compare and contrast our proposed co-design approach to the traditional individual design method, show that co-design can more effectively enhance binding affinity, and take the opportunity to empirically evaluate the resulting pairs and offer new biochemical insights on the specifics of the binding interaction.

The second problem is site-of-metabolism (SOM) prediction in small molecules due to enzymatic interactions. Enzymatic reactions are different in nature from chemical reactions, so the traditionally used large molecular datasets such as USPTO-50k [3], USPTO-full [4], and ZINC250k [5] are not appropriate for the task. Instead, a dataset of enzymatic reactions is needed, which in our case was based on KEGG RPAIR [6] and only featured about 21k molecules. To work around this data constraint, we use a graph neural network (GNN) approach to learn molecular representations based on this small dataset. GNNs are naturally suited to graph-like structures, and by treating molecules as graphs, we are able to take advantage of the GNNs' message passing features to capture the full context of the molecules effectively. In addition, we propose a data pre-processing scheme that relabels SOMs in a consistent manner throughout the dataset and ensures the model is always trained on correct SOM labels. We evaluate our approach against several baselines and different types of GNNs and present two biological use cases aided by SOM prediction.

The third problem is classification of natural product molecules arising from the expression of biosynthetic gene clusters (BGCs). Here, the dataset was even more limited, featuring approximately 3,000 human-annotated BGCs with known product structures. Such a low quantity of data poses a major challenge for learning sequence representations, particularly

with Transformer-like architectures. To reduce the impact of this issue, we offload BGC gene representation to ESM Cambrian [7], a large protein language model trained on millions of sequences. The ESM Cambrian embeddings provide a rich context about the biosynthetic activity of BGCs, allowing our model to effectively learn the natural product classification. In addition, we introduce a data augmentation approach that identifies similar BGCs and incorporates them in the training process, which also improved the performance of the model. Overall, these three applications demonstrate an array of data-efficient strategies essential for solving biological challenges.

1.1 Protein-ligand co-design

Synthetic biology seeks to redesign biological systems for performing specific tasks by engineering their constituents, such as protein sequences and molecules. Protein engineering aims to identify mutants with improved properties, such as selectivity, specificity, expression, solubility, and stability. As the combinatorial mutational space of protein sequences is extremely large, a purposeful approach is required to design and explore protein variants. Ligand design aims to select molecules with specific properties, such as affinity, druglikeness, metabolic stability, and bioavailability [8]. However, the space of molecular structures is also extremely large: for example, the number of pharmacologically relevant structures alone may exceed 10^{60} [9]. In both the protein and the ligand case, efficient exploration of the design space is required in order to make the problem tractable.

A common modality of existing design methods is their assumption of the individual design paradigm, where proteins and ligands are designed individually. Individual design is appropriate in situations where either the protein or the ligand is allowed to vary and the other is fixed, for example when developing a drug to target a particular protein or when an enzyme is mutated to maximize yield of a particular product. However, there are emerging applications, most notably in bioelectronic sensing [10] and synthetic biosynthesis [11], where it is desirable to engineer *both* the ligand and protein. A co-design paradigm would

allow for simultaneous changes of ligand and protein, and the consideration of how the change of one component affects the function of the other or the interaction between them. For example, binding affinity is dependent on both enzyme and ligand, and in the space of possible co-modification, more favorable pairings may be identified by matching the binding pocket volume to the size of the molecule and vice versa.

One example of an application that benefits from co-design is the case of type II NADH:quinone oxidoreductase, also known as Ndh2, and its interaction with quinones. Ndh2s comprise a family of membrane-bound proteins acting on nicotinamide adenine dinucleotide (NAD), responsible for oxidation of NADH and reduction of quniones. They play a central role in respiratory chains of many organisms [12] and help maintain the balance of NADH and NAD^+ [13]. Ndh2 has been proposed as a drug target for treatment of numerous infectious and chronic diseases [12, 14, 13, 15], and like other oxidoreductases, may be of interest in measurement and pollution control applications [16]. Recently, an extracellular electron transfer (EET) pathway has been described in *Lactiplantibacillus plantarum* that can reduce an extracellular electrode in the presence of exogenous quinones using Ndh2 as a mediator [17]. The rate of EET varied across different quinone analogs and concentrations, creating distinguishable electronic signals and presenting an exciting opportunity for building inexpensive whole-cell bioelectronic sensors for pharmacologically relevant quinones. However, improving sensitivity or selectivity of this sensing system through Ndh2 and/or quinone co-design remains a challenge, with only a small number of quinones currently characterized for this purpose [18, 19].

1.2 Site-of-metabolism prediction

Identifying SOMs can significantly enhance our understanding of metabolism. Such sites refer to specific sites within molecules that are susceptible to chemical change. SOMs can therefore be defined as specific atoms [20, 21, 22] and/or bonds [23, 24], and, less conventionally, pairs of unshared valence electrons [23]. A primary application that has driven the

development of SOM prediction tools is determining the enzymatic transformations of xenobiotic compounds including drug molecules. Such transformations are generally classified in two groups – "Phase I" activity involving oxidation–reduction and hydrolysis reactions and "Phase II" reactions referring to conjugation transformations [25]. Cytochrome P450 (CYP) enzymes, a superfamily of structurally diverse metabolic enzymes with broad specificity, are known to be responsible for the metabolism of over 70% of all drugs in use [26, 27] and are the primary facilitators of phase I reactions [28]. Non-CYP enzymes, on the other hand, are the primary drivers of phase II metabolism, e.g. UDP-glucuronosyltransferases (UGT) and sulfotransferases (SULT) [29].

Importantly, broad enzyme specificity is not restricted to CYP enzymes or only those associated with phase I and II metabolism. Most, if not all, enzymes are promiscuous, acting on substrates other than the ones they evolved to catalyze [30, 31]. Three applications, constructing of de novo synthesis pathways [32], creating extended metabolic models (EMMs) that account for enzyme promiscuity [33], and identifying metabolic products measured through metabolomics [34], have driven the development of tools to analyze broad promiscuity. The prevailing approach is to first identify a set of reaction rules (e.g. [35, 36, 37]) in the form of a local biochemical transformation, followed by matching them to query molecules. Matching rules specify the site of the transformation as well as its local neighborhood to ensure they are sufficiently specific to generate likely promiscuous products. Tuning this specificity (e.g. by radius adjustment) however is a challenge and matching rules may still yield infeasible biotransformations. When paired with rule-based methods, machine learning (ML)-based SOM prediction approaches provide two major advantages. First, they can account for a wider molecular context and learn specialized representations necessary for accurate predictions. Second, they provide a continuous likelihood estimate for the SOM, allowing the ranking of promiscuous products. These improvements can enrich the results of rule-based methods and broaden their applications.

1.3 Biosynthetic gene cluster product classification

Natural products have been a staple of medicine for much of human history. Natural products represent a vast reservoir of potential drug candidates, with a high proportion of biologically-active species [38, 39]. A large percentage of drugs in current use owe their existence to natural products, including antibiotic lariocidin [40], anti-tumor agent anthramycin [41], antiparasitic drug Ivermectin [42], and approximately 70% of anti-infective drugs [43]. By fall of 2019, over a third (38%) of FDA-approved new molecular entities were either natural products or their derivatives, and in select areas (e.g. cancer drugs) that percentage can be as high as 65% [38]. As such, there is an ongoing interest in natural products and their sources.

In microbes, the specialized genes that are used to synthesize natural products are clustered into biosynthetic gene clusters (BGCs). These genes encode enzymes that synthesize and tailor the structure of the product, as well as regulatory, and transport proteins involved in its synthesis [44]. The recent explosion of microbial genomic data has led to the development of computational techniques for identifying BGCs. Early methods relied on sequence alignment algorithms and manually curated rules. However, with the rapidly growing quantity of identified BGCs and their diversity, algorithmic and rule-based approaches have given the way to machine learning techniques, e.g. DeepBGC [45], BiGCARP [46], and BGCCGB [47], promising a more flexible and comprehensive framework for understanding BGCs. AntiSMASH (ANTIbiotics and Secondary Metabolite Analysis SHell), currently the most widely used tool for mining microbial genomes for BGCs [48], leverages profile Hidden Markov Models in conjunction with a bank of rules for detecting and characterizing BGCs, with the latest version supporting up to 101 cluster types [49].

However, despite the advances in BGC detection, obtaining detailed information about natural products arising from BGCs remains challenging, with the lack of high-quality data associating BGCs and natural products being a key obstacle. For example, the largest repository of such information, MIBiG, provides a selection of 3,013 BGCs. Consequently,

a common strategy has been to group BGCs into gene cluster families (GCFs), which can link BGCs to better-annotated MIBiG entries that include structural information as a starting point for BGC characterization [50, 51]. Implementations of this technique typically rely on network analysis [52], sequence similarity [53, 54], or a combination thereof [55, 51], with statistical correlation techniques sometimes used to leverage mass spectrometry data [53, 56]. However, the focus thus far has been on establishing connections to known clusters as opposed to learning more general patterns of BGC product expression.

1.4 Thesis contributions

This thesis presents data-efficient computational strategies for solving three biological problems: protein-ligand co-design, SOM prediction, and BGC product classification.

We first present a method for the computational co-design of protein-ligand pairs and its application to Ndh2-quinone binding for EET rate maximization. Our proposed paradigm, referred to as "co-design," aims to simultaneously explore the space of ligand and protein modifications that can lead to improved system design. In the case of EET, we seek to improve binding affinity over the Ndh2-quinone complexes currently known for facilitating EET. Our goal herein is therefore to show that the co-design paradigm can yield enhanced biological systems over individual design paradigms, particularly in the context of Ndh2-quinone-mediated EET. We utilize random sampling as our design space exploration strategy, which provides a uniform benchmark for comparing the two paradigms and showcases the advantages of the co-design approach.

Our approach consists of three major steps. First, we build libraries of protein and quinone variants by considering changes likely to impact their mutual binding. For protein variants, we identify a number of residues in close proximity to the quinone binding site and enumerate all possible single amino acid substitutions in those locations. For ligand variants, we construct modified quinones by iteratively adding functional motifs such as phenyl and hydroxy groups to one of the two EET-active molecules: 1,4-dihydroxy-2-naphthoic acid

(DHNA) and menadione [18]. Second, we implement two strategies to search the protein-ligand space for pairings with improved binding affinity: individual design and co-design. In individual design, we explore protein-ligand pairs where only the protein or only the ligand has been modified, as is common in current design paradigms. Meanwhile, with co-design, we consider all combinations of proteins and ligands in our libraries. Third, we evaluate the resulting combinations using molecular docking with AutoDock Vina [57]. The binding affinity predicted by AutoDock Vina represents the favorability of protein-ligand interaction, which we believe is a key contributor to EET activity. However, limited receptor flexibility is one of the major shortcomings of molecular docking methods [58] and is commonly addressed by leveraging the more accurate MD simulations [59]. As such, we subsequently validate our proposed combinations using predicted Molecular Mechanics with Generalized Born and Surface Area Solvation (MM/GBSA) based free energy of binding calculations from Schrödinger Maestro [60, 61, 62].

Next, we introduce a graph-based deep-learning technique for predicting SOM called *GNN-SOM*. Specifically, we use GNNs [63, 64] to learn atom representations in their molecular graph context in an end-to-end fashion [65]. These representations can be utilized in downstream classification tasks: either to predict the likelihood of an atom being an SOM or to predict the likelihood of a bond between two atoms being an SOM. A major advantage of using GNNs for SOM prediction is the more natural problem representation as atoms and bonds in molecular structures trivially correspond to nodes and edges in graphs, respectively. Another important advantage of utilizing GNNs over current traditional ML approaches, e.g. Random Forest (RF) and Support Vector Machines (SVM), is effective representation learning, which helps avoid the burdensome task of feature selection. In contrast, the representation learning capabilities of GNNs allow models to perform well with a handful of basic features, such as atom element types and enzyme category labels. We explore several GNN models and select an architecture featuring the Chebyshev convolutional operator [66] as the optimal design for the SOM prediction task. Further, we demonstrate the utility of

SOM prediction for improving rule-based enzyme promiscuity prediction in the context of creating enzyme promiscuity-aware EMMs and constructing synthesis pathways.

Finally, we introduce *BGCat* (BGC annotation tool), a technique for fine-grained structural classification of BGC-encoded natural products following the NPClassifier nomenclature designed for natural products [67]. Our technique leverages ESM Cambrian, a protein language model trained on a wide range of protein sequences [7], as the engine for condensing relevant biosynthetic genes into meaningful embeddings. We show that our approach surpasses the state-of-the-art techniques for traditional coarse-grained BGC product classification. Next, we demonstrate that our technique is effective for detailed product classification. We then introduce a clustering-based augmentation method that provides additional data for model training and enhances its performance. Next, we investigate product class profiles (PCPs) of GCFs to help identify the most meaningful families. Lastly, we leverage *BGCat* to introduce detailed product labels for thousands of BGCs in the antiSMASH database (antiSMASH DB) [68], including the many clusters with unknown products.

The contributions of this thesis are as follows:

1. We introduce a co-design paradigm for engineering protein-ligand pairs for optimal binding affinity. In contrast to traditional individual design approaches, our method simultaneously varies both the protein and the ligand. This more comprehensive approach explores a larger portion of the protein-ligand landscape and is able to identify more favorable pairings.

2. A structural model of the *L.plantarum* Ndh2 and its quinone binding pocket is proposed. The model is estimated to have favorable functional characteristics based on quality analysis and its similarity to known crystal structures of other Ndh2 proteins. The presented structure enables computational modeling of the protein and identifies the specific amino acids likely to alter its binding behavior.

3. A selection of 10 highest-affinity Ndh2-quinone combinations is proposed and classified into three groups based on the location of the quinone in the pocket. The molecular interactions are empirically considered and residue 386 is identified as a key contributor to successful binding.

4. Formulating the SOM prediction problem across all enzyme classes (not just CYP or phase I/II enzymes) as a classification task on either atoms (the atomic SOM problem) or as a classification task on pairs thereof (the bond SOM problem).

5. Exploring several GNN models and demonstrating the effectiveness of GNNs in representation learning and SOM prediction over traditional ML approaches that require feature selection. A GNN architecture using the Chebyshev convolutional operator achieved the molecule-level AUROC of 0.953 compared to 0.915 of a simpler GNN design and 0.850 of the best-performing baseline model.

6. Demonstrating the relative difficulty of SOM prediction for CYP-mediated reactions and showing that our general-purpose SOM prediction model performs as well as the versions specific to CYP and non-CYP interactions, thus alleviating the need for separate predictors for the two cases. The model achieves molecule-level AUROC of 0.910 for CYP reactions and 0.959 for non-CYP reactions; the use of reaction-specific predictors resulted in 0.002 variation in this metric.

7. Presenting two biological applications of our SOM predictor, where identifying SOMs improves the precision of predicted promiscuous products and identification of 3-HP synthesis pathways in *Escherichia coli*. In particular, we found that introducing an SOM-likelihood cutoff on putative promiscuous products increases the proportion of products that were previously observed in the organism. Additionally, using SOM likelihood as the probability of a reaction step assigned probabilities of 0.987 and 0.956 to two known 3-HP pathways, compared to the probabilities of 0.30 and 0.37 of a typical pathway of the corresponding length.

8. Introducing *BGCat*, a technique for detailed BGC product classification based on the NPClassifier nomenclature and a pretrained protein language model. On traditional coarse-grained product classification, the method outperforms state-of-the-art approaches, achieving AUROC of 0.937. Meanwhile, on detailed classification, the AUROC varied from 0.876 to 0.939 depending on the level of label hierarchy.

9. Demonstrating the method's ability to generalize to undiscovered BGCs and unseen GCFs. On a temporal data split, where the training set consisted of BGCs from an earlier release of MIBiG and the test set of those from a later release, the model achieved comparable performance to a random five-fold cross-validation split. For unseen GCFs, the AUROC varied from 0.772 to 0.835 depending on the level of hierarchy.

10. Implementing an augmentation strategy based on BGC Atlas BGCs sharing the same GCF. The use of an augmented dataset led to a consistent increase in all evaluation metrics except for precision. In particular, a significant increase in recall was observed for class-level labels with the temporal split: increasing from 0.256 to 0.413.

11. Developing the concept of GCF product class profiles, enabling characterization of GCFs based on the product types resulting from the constituent BGCs. Each GCF is associated with a probability distribution of product types and is compared to the overall distribution of the dataset using a statistical test. Out of 18,596 GCFs identified in the BGC Atlas dataset, 5,083 had a distinct profile, indicating that those families may be strongly associated with particular product types.

12. Providing product labels for 121k (46%) BGCs cataloged in antiSMASH DB that currently lack references to clusters in MIBiG with known product structures. Without such references, determining the function of a BGC is challenging due to lack of detailed information about the product. The BGCat labels fill in this gap.

1.5 Thesis organization

In this thesis, we consider data-efficient computational strategies for solving three distinct biological problems. In chapter 2, we describe prior works in the space of each problem and provide the necessary background. In chapter 3, we present a protein-ligand co-design approach for enhancing binding affinity and, by proxy, EET rate in Ndh2-quinone pairs. In chapter 4, we introduce *GNN-SOM*, an ML-based technique for identifying SOMs using GNNs. In chapter 5, we present *BGCat*, a detailed BGC product classification method leveraging a pretrained protein language model and a deep neural network. In chapter 6, we summarize the major conclusions of this thesis and outline future research directions.

Chapter 2 - Background

The three biological problems considered in this thesis are diverse in their scope, yet they all share the challenge of data scarcity and thus require methods that use the available information effectively. This chapter aims to outline the prior work relevant to each of the three problems and provide important biological background where necessary.

2.1 Protein-ligand co-design

The traditional approach to protein and ligand design revolves around the concept of individual design, where only one component of a system is being engineered, while all others are held constant. A popular method for protein engineering is directed evolution (DE), where proteins undergo iterative rounds of gene diversification and screening, mirroring the process of natural evolution on an accelerated timescale. Although labor-intensive, this method has been successfully used for evolving biological pathways and improving catalytic properties of enzymes [69]. Complementing the purely experimental approach of DE, rational design [70, 71] leverages structural and mechanistic properties of proteins to alter function in a desirable way. In addition, a growing selection of computational methods is available, from heuristic-based optimization techniques [72, 73], to highly parallelized molecular dynamics (MD) simulations [74] and ML models [75, 76, 77, 78, 79, 80], helping to elucidate protein-function relationships in a less expensive fashion. On the ligand front, a number of computational approaches have likewise emerged, ranging from fragment-

based ligand design to evolutionary algorithms [81]. Several ML methods have also been introduced [82, 9].

The space of both possible proteins and ligands is extremely large, so one of the objectives has always been efficient exploration of possible design options. This can be achieved, for example, by seeding the design process with a functional protein or molecule as small-scale changes are likely to retain much of the function. Rational knowledge can limit changes to those most likely to produce a desirable outcome. Simulations and model predictions can provide low-cost feedback prior to embarking on more costly experimental studies. And in some cases, the focus on individual design can be a deliberate decision to avoid the combinatorial explosion of the solution space; although, in either case, the space remains intractably large if tackled naively.

However, individual design is highly limiting in applications where both the protein and the ligand may vary as it explores only a single dimension of the design space. This thesis focuses on one such case, design of Ndh2-quinone pairs with the goal of enhancing EET, which has promising applications for bioelectronic sensing and response to pollution concentrations [16]. To this end, we developed a model of Ndh2 and its quinone binding pocket and used established molecular mechanics to limit the scope of mutations to the few likely to impact the binding affinity of the protein to the quinone. On the molecule side, we iteratively "grew" the base quinone structure with biologically-active functional groups, thus avoiding the vast swaths of molecules with minimal change in binding behavior. This resulted in a manageable design space, which we could then explore with random sampling.

2.2 Site-of-metabolism prediction

Much of the existing work in the context of SOM prediction has been driven by interest in understanding enzymatic transformations of xenobiotic compounds, like those that may occur when a pharmaceutical is given to a patient. This can be an important consideration with respect to drug safety and side effects: for instance, a popular over-the-counter drug

Tylenol, when taken in large quantities or together with alcohol, can lead to production of a toxic metabolite NAPQI due to CYP enzyme activity [83]. As a result, many methods specialize in particular types of reactions most relevant to this scenario. An early method in this space, SMARTCyp [84], identifies CYP-mediated SOMs in drug-like molecules using an empirical scoring function, parts of which subsequently found their way into later methods. RS-Predictor [83] leverages various molecular descriptors in conjunction with an SVM-like method to identify SOMs of CYP 3A4. XenoSite [27] uses a similar set of descriptors together with a neural network to predict SOM likelihoods for nine CYP enzymes. Later Rainbow XenoSite [23] also follows this approach with the goal of predicting different SOMs for five classes of Phase I reactions, primarily catalyzed by CYPs but also oxidoreductases and hydrolases. Meanwhile, the model proposed by He et al. [24] uses an ensemble of ML methods for identifying SOMs of six types of oxidoreductase-mediated reactions.

Some methods aim for more general SOM prediction by incorporating support for Phase II reactions, which comprise the other major class of xenobiotic transformations. FAME (FAst MEtabolizer) [21] uses an RF predictor trained on over 20k molecules to predict SOMs in both Phase I and II interactions. A similar approach is also seen in MetScore [20], which includes RF models for both types of reactions trained on 17k transformations. However, not all enzymatic reactions are covered by these two types of interactions.

Learning effective molecular representations is a major challenge for most SOM prediction methods. These methods invariably rely on complex molecular descriptors, specifying localized topological and quantum chemistry information, to provide the necessary context for SOM prediction. In methods such as SMARTCyp, an empirical scoring function was designed to aggregate these descriptors into a meaningful SOM signal. Meanwhile, other methods required complex feature selection schemes, ranging from the screening of specific descriptor combinations [20] to the simultaneous use of four feature selection algorithms [24].

Our proposed approach, *GNN-SOM*, uses GNNs to naturally learn molecular representations from basic element types and the 2D topology of the molecule, thus obviating the need for sophisticated descriptors or feature selection strategies. The method is also not specific to any reaction type, allowing its application to both common xenobiotic reactions and the more exotic scenarios like enzyme promiscuity. The model is trained on a set of 21,023 molecules sourced from the KEGG RPAIR database [6]. In order to effectively learn from this dataset, we introduce several preprocessing steps to improve its quality. Specifically, we ensure that SOMs are consistently annotated in the case of cleaved bonds and are exhaustively labeled in symmetrical structures. In addition, we structure our model as an ensemble of GNNs trained on different folds with the aim of enhancing generalizability and reducing overfitting.

2.3 Biosynthetic gene cluster product classification

Though BGC product classification is an established problem, it is often treated as incidental to BGC identification within an unannotated genome. For example, the leading platform for BGC detection and analysis antiSMASH [49] uses profile-Hidden Markov Models (pHMMs) to recognize 101 cluster types in its latest version; these cluster types provide some information about the product type. Another example of a BGC detection method is DeepBGC [45], which treats the genome as a sequence of protein family (Pfam) [85] domains. Each domain represents a group of amino acids cataloged in the Pfam database, and in the case of DeepBGC, is embedded using a word2vec-like network. DeepBGC then leverages a Bidirectional Long Short-Term Memory Recurrent Neural Network to locate BGC boundaries and provides product classification with an RF classifier. BiGCARP [46] uses a similar arrangement, except is uses the ESM-1b protein language model for Pfam domain embedding and identifies BGCs using a convolutional autoencoding representations of proteins (CARP) [86] model. For product classification, BiGCARP introduces a mask token at the beginning of each BGC sequence to represent the product type – classification

thus becomes an inherent part of the detection model. BGCCGB [47] also adopts this design, representing BGCs as sequences of Pfam domains and learning their embeddings with a BERT [87] model, and is likewise capable of BGC detection and classification.

A common limitation of these methods is their use of a coarse-grained classification system for product types, which we refer to as the "traditional" approach. Though it is less of a concern with recent versions of antiSMASH and its growing library of cluster types, other methods adopt the MIBiG nomenclature, which groups products into seven broad categories. This labeling provides limited information about the product of a BGC and, even in the case of antiSMASH, leaves a lot to be desired.

To provide more insight into the product, some methods take upon the task of structure prediction, which is a formidable challenge because many products are assembled from peptides and modified by tailoring enzymes, leading to complex structures with potentially hundreds of atoms. Nonetheless, several methods were proposed for product structure prediction, including RiPPMiner [88], TransATor [89], and PRISM [90]; however, they are either limited to specific types of BGCs or are unable to recognize novel enzymatic activities, restricting their general applicability. Although antiSMASH offers inference capability for the approximate scaffold of certain types of BGCs, it is not yet able to predict the full product structure accounting for tailoring enzyme modifications. However, once the product structure is available, the corresponding product type can be predicted with a great level of detail using tools such as NPClassifier [67] or ClassyFire [91].

The limitations of the traditional approach to BGC product classification did not go unnoticed and recently, CHAMOIS [92] was introduced as a generalized method for detailed product classification according to the ClassyFire nomenclature. This approach starts off with the familiar Pfam domain representation and uses LASSO logistic regression [93] for its binary classifiers. Though an major advancement over the coarse-grained MIBiG classification, ClassyFire is not the ideal choice for this task. The products of BGCs fall under the umbrella of natural products, which have an established approach to classification that

prioritizes taxonomic and functional properties of compounds. Meanwhile, the ClassyFire system, having been designed for the general organic chemistry community, focuses on the structural aspects of compounds [67].

Another challenge for BGC product classification is the limited information about BGC products, with the largest resource, MIBiG, listing only 3,013 BGCs. Recognizing this gap, BGCCGB introduces a data augmentation scheme based on the concept of synonym replacement borrowed from natural language processing. Each Pfam domain in the vocabulary is treated as a word and embedded with the ESM-1b model, cosine similarity is then calculated between every pair of embedding vectors. Pfam domains with similarity exceeding 0.95 are treated as synonyms for the purposes of substitution. The expanded dataset is then used for training the model, which was found to improve its performance.

In this thesis, we present *BGCat*, an approach to BGC product classification that uses the NPClassifier [67] labeling system. In contrast with other methods, BGCat provides detailed information about the product type at a most appropriate level for natural products. The dataset is constructed using the MIBiG BGCs, with labels generated by NPClassifier. To enable learning on this constrained set, we employ two key strategies. Like BiGCARP and BGCCGB, we leverage a pretrained language model for effective sequence representation, though we use a more recent version of the model and embed genetic sequences at the gene level as opposed to the Pfam domain level. The large context window of the model suitable for protein embedding obviates the need for embedding individual domains. In addition, the focus on the genes allows our model to ignore non-biosynthetic genes with limited effect on the product type. The second strategy we use is data augmentation using GCFs and a diverse set of BGCs from BGC Atlas. Both sets of BGCs are grouped together into families based on the presence of specific Pfam domains [51], which is reflective of their structure and function. BGCs sharing a GCF are presumed to yield similar products, so the product labels from the MIBiG BGCs are used to annotate BGC Atlas BGCs in the same GCFs.

Chapter 3 - Protein-ligand co-design: a case for improving binding affinity between Type II NADH:quinone oxidoreductase and quinones

Simultaneously co-designing the protein and the ligand leads to more favorable binding affinities compared to traditional individual design approaches changing only the protein or only the ligand. However, the combinatorial increase of design possibilities requires a data-efficient exploration strategy. In this chapter, we present a co-design framework for protein-ligand pairs and its application to Ndh2-quinone combinations for binding affinity enhancement. We first model the Ndh2 protein and its interaction with quinones, identifying design options likely to impact the binding affinity. Next, we explore this design space using sampling approaches, which allows us to directly compare individual and co-design paradigms. Finally, we evaluate the resulting combinations and offer biochemical insights on the problem.

3.1 Methods

3.1.1 Modeling *L. plantarum* Ndh2

The sequence of the *L. plantarum* Ndh2 protein was obtained from the UniProtKB database [94] (entry D7VAS3). Structural homology models were then constructed using AlphaFold [95, 96] and Robetta [97] (RoseTTAFold [98] model) web servers, two state-of-the-art protein structure prediction methods. To ensure consistent positioning of residues, cofactors,

and ligands across the structures, one of the Robetta models was arbitrarily chosen as the coordinate reference and all other models were oriented to it using the PyMOL Alignment [99] plugin.

The crystal structure (PDB ID 4G73) [100] of an Ndh2 variant sourced from another organism, *S. cerevisiae* S288C, was used to validate and augment the homology models. This structure from Feng et al. [100] is one of the few depictions of an Ndh2-quinone complex we could locate in the Protein Data Bank (PDB). Feng et al. also proposed an alternative (PDB ID 4G74) of the Ndh2-quinone complex; however, it included a Triton X-100 molecule, which would not be present *in vivo*, so we removed this molecule. The chosen structure from *S. cerevisiae* S288C had 26% sequence identity to the *L. plantarum* Ndh2 as measured by blastp [101] and presented a similar structure, yielding alignment RMSDs of 1.295 Åand 1.457 Åfor AlphaFold and Robetta models, respectively.

The *S. cerevisiae* S288C crystal structure also included poses for flavin adenine dinucleotide (FAD), NADH, and a quinone (Ubiquinone 25) bound to the protein. Those poses were overlaid on our homology models, providing the locations of the cofactors as well as the binding pocket for the quinone. The presence of NADH is not required under all proposed Ndh2 catalysis mechanisms. In particular, the ping-pong mechanism assumes that NADH and quinone are not bound to the protein at the same time, but react with it sequentially [12]. As such, we excluded the NADH pose from our subsequent modeling, but retained FAD. The models were evaluated based on their ability to dock with the quinone and their overall structure quality, and the AlphaFold model was selected moving forward; see the Results section for more details.

3.1.2 Protein and ligand preparation

A library of Ndh2 mutants was constructed by enumerating all single-point mutations of residues comprising the quinone binding pocket. These residues were manually selected based on their close proximity to DHNA when docked with our model of Ndh2. Three

contiguous ranges of amino acid positions were chosen for mutation: 320-322, 353-355, and 381-390. The PyMOL Mutagenesis tool [99] was used to introduce the mutations. In instances where multiple side chain orientations were possible, we selected the most probable rotamer predicted by PyMOL. With 16 mutable positions and 19 possible amino acid replacements at each position, this process yielded a library of 304 mutants.

The resulting mutants were further analyzed to consider the favorability of their corresponding protein mutations. The Schrödinger Maestro [60] Residue Scanning tool was used to predict the change in Prime MM/GBSA free energy ("Δ Prime" hereafter) caused by all single amino acid substitutions. A negative change describes a mutation more favorable than the wildtype, while a positive change indicates the opposite. Changes greater than +10 kcal/mol generally imply a major structural change in the protein, which would likely be detrimental to its ability to bind to a quinone and/or perform EET. Therefore, all mutated variants with Δ Prime exceeding that threshold were discarded from further consideration. After discarding such variants, the final library consisted of 183 mutants.

A library of ligands was constructed by iterative addition of physiologically relevant functional groups to base molecules: initially DHNA and menadione, the two quinones known to facilitate EET in *L. plantarum* Ndh2 [18]. The functional groups were attached to heavy atoms on the periphery of either DHNA or menadione, with the exception of the carbonyl oxygen atoms in positions 1 and 4 (see Figures 3.5e and 3.5f) as to not impede the quinone $\leftrightharpoons$ semiquinone transformation required by EET [18]. The functional groups included benzene, C_2H_5, CF_3, CH_2OH, CH_3, CHF_2, F, NH_2, OCH_3, and OH. [102, 103, 104] In addition, we considered groups CH_2NH_2, NO_2, and C_4H_4NH; however, these motifs did not yield any useful molecules in our process and were excluded accordingly. The addition of a functional group was implemented as a generic reaction described by a SMARTS pattern [105] and applied using RDKit [106]. The resulting derivative molecules were validated for basic chemical validity using RDKit and cross-referenced against PubChem [107]. Invalid molecules or those not found in the database were excluded from further consideration. This

process was repeated up to 6 times to produce all variant molecules in the library, resulting in 822 unique ligands.

3.1.3 Search strategy

We considered two strategies in this study: individual design and co-design. In individual design, only the protein or the ligand is being varied, with the other assumed as unchanged. Therefore, to generate protein-ligand pairs, we exhaustively enumerated all combinations featuring either (1) the unmodified DHNA and menadione quinones and one of the Ndh2 variants from our mutant library, or (2) the wildtype Ndh2 protein and a molecule from our ligand library. The former set corresponded to design options of the protein, while the latter provide design options for the ligand.

In co-design, modified proteins and ligands are being considered for evaluation. This strategy can identify pairs featuring both a mutated Ndh2 variant and a modified quinone molecule. Unfortunately, the combinatorial space of such pairs is large, so the exhaustive enumeration approach we used for individual design would not be feasible. Instead, we randomly sampled a number of pairs corresponding to a subset of all possible protein-ligand pairs accessible to co-design.

3.1.4 Molecular Docking

The binding of *L. plantarum* Ndh2 variants and ligands was modeled using AutoDock Vina [57] with bounding box dimensions 15Å × 17Å × 20Å and center position (38, 44, 2), referenced to the Robetta model selected. The size and placement of the box were manually chosen as to include the quinone binding pocket while rejecting most of the poses inconsistent with the expected quinone orientation with respect to FAD and the protein. The exhaustiveness of the search was set to 8, the maximum number of binding modes to be generated was 9, and the maximum energy difference between displayed modes was set to 3 kcal/mol. When multiple poses were possible, we selected the one with the most favorable

predicted binding affinity.

3.1.5 Evaluating protein-ligand pairings

The obtained pairings were evaluated based on the binding affinity predicted by AutoDock Vina as a part of molecular docking, as well as the predicted free energy of binding calculated using MM/GBSA. To run MM/GBSA, first each protein-ligand pair was converted into the compressed Maestro ("maegz") format. The Ndh2 variant structure was refined with the Protein Preparation Wizard (prepwizard) [108]. The quinone was prepared using the LigPrep program [109]; since this operation moved the molecule away from its docked pose, the LigPrep-processed ligand was subsequently re-aligned to the original output from AutoDock Vina. The prepared protein and quinone were combined into one complex and forwarded to the Prime program [61, 62] for the MM/GBSA calculation. This workflow was automated using the JobDJ toolkit and command line utilities available in the Schrödinger Maestro suite [60]. Finally, the pairings were sorted by their AutoDock Vina binding affinities, and the top 10 were chosen for in-depth consideration.

3.2 Results

3.2.1 Structural model assessment

The modeled structures of the *L. plantarum* Ndh2 protein were first validated by ensuring they can dock with DHNA and menadione, the two quinones known to interact with the protein. Although both AlphaFold and Robetta homology models were considered, only the former yielded meaningful poses. Upon closer inspection of the residues in the pocket, we identified an unfavorable orientation of the Y390 side chain, exclusive to the Robetta models. Such orientation resulted in a clash with the expected position of the quinone that was implied by the placement of Ubiquinone-25 in the reference *S. cerevisiae* S288C Ndh2 structure (Figure 3.1). Replacement of Y390 with alanine resolved the issue, confirming the unfortunate orientation of the side chain as the cause. However, because the original

Robetta structure wasn't effective in modeling quinone binding, we chose to proceed with the AlphaFold structure from this point on.

The selected homology model was then analyzed for quality using the SWISS-MODEL [110, 111, 112, 113, 114] and ProSA [115, 116] web servers, which indicated that our model had favorable structural characteristics in terms of (Φ, Ψ) residue angles, Z-score, and QMEAN assessment.

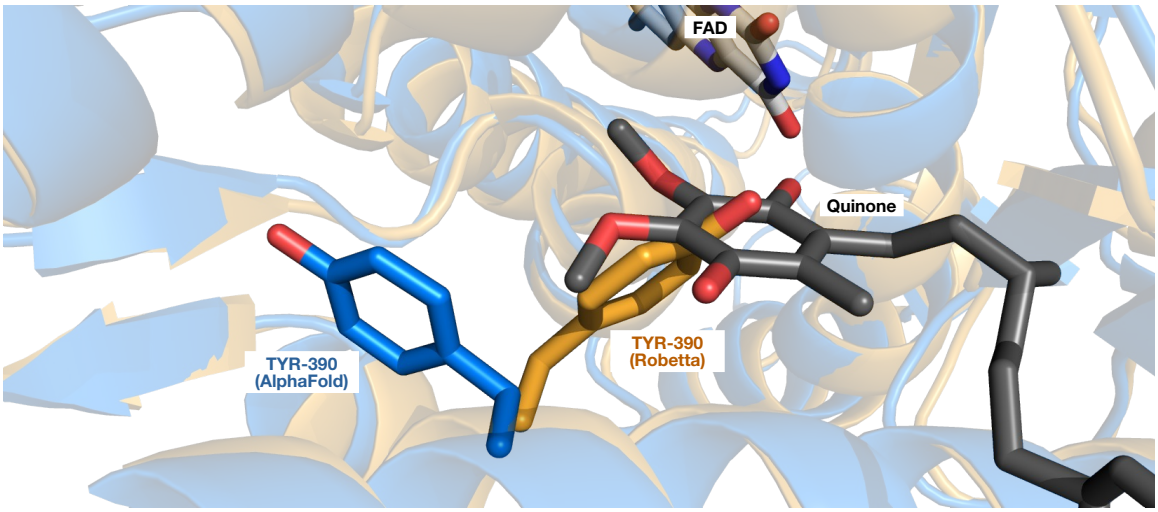

Figure 3.1 Y390 residue placement in AlphaFold (blue) and Robetta (orange) models. The expected position of the quinone (dark gray), superimposed from the *S. cerevisiae* S288C Ndh2 structure, demonstrates a clash unique to the Robetta model.

To explore the characteristics of the *L. plantarum* Ndh2, we compared the residues in its quinone binding pocket with those in the *S. cerevisiae* variant (see Figure 3.2). Although several residues have mutual counterparts in both proteins and enclose the quinone in a similar manner, there are a number of differences that may contribute to unique behavior of the *L. plantarum* Ndh2. For example, the Q353 residue corresponding to L444 in 4G73 [100] has a different affinity to water, which may significantly alter the interaction with the quinone given the pocket's proximity to the cell membrane. Additionally, two residues, E324 (H397 in 4G73) [100] and K383 (Y482), differ in the presence of a positively charged side chain, which may effect the orientation of the ligand. Those differences emphasize the need for further study of the *L. plantarum* Ndh2; however, the ability to dock DHNA and

menadione in a similar position to the previously reported crystal structure lends confidence in our predicted model.

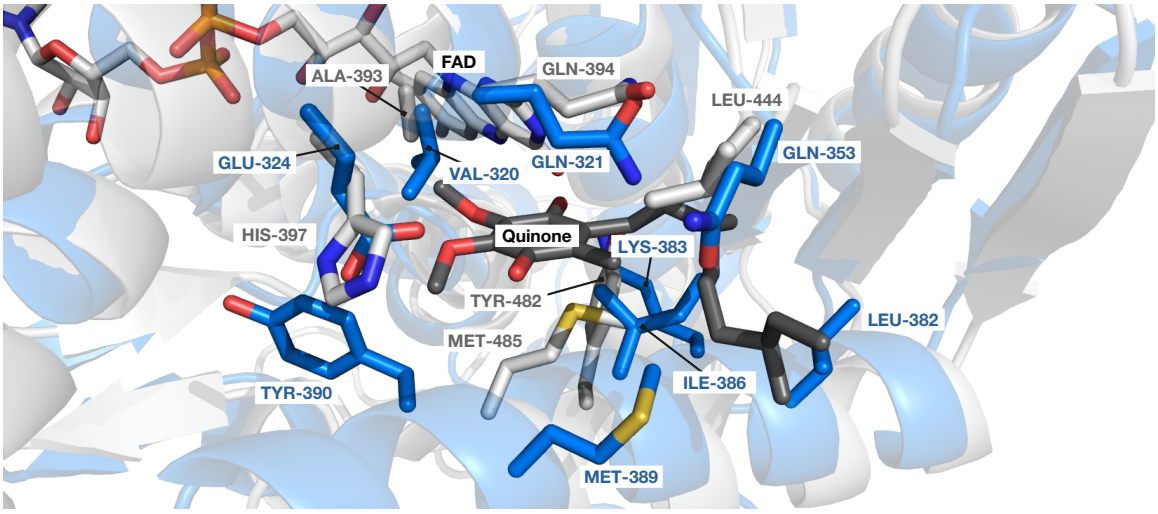

Figure 3.2 Overlay of *L. plantarum* (blue) and *S. cerevisiae* (light gray) structures shows the common features of the quinone (dark gray) binding pocket as well as the differences unique to each variant. Labels indicate the residues of interest.

3.2.2 Strengths and tradeoffs in co-design

Intuitively, the co-design approach significantly expands the solution space, which can be of great benefit but introduces challenges of its own: consideration between capabilities and limitations is necessary to deploy this technique in practice. A key advantage of co-design is its ability to consider the mutual effects of interactions between proteins and ligands as they both are allowed to vary during the design process. In addition, co-design can explore protein-ligand combinations that are out of reach of traditional individual design approaches and thus potentially identify combinations with improved function. However, this larger solution space can also be of detriment: with the spaces of proteins and ligands already being computationally intractable, exploring the combined co-design space can be especially challenging. As such, it is important to aggressively limit the degrees of freedom and focus on variations that are likely to support the design objective.

In the case of Ndh2-quinone co-design, we took care to limit modifications to only

a handful of highly-relevant residues and functional groups. For the individual design strategies focused on the protein and the ligand, this resulted in a very modest number of possibilities: 366 (2 ligands × 183 mutants) and 822 (822 ligands × 1 wildtype Ndh2) pairings, respectively. The co-design space, however, contained over 150,000 (822 ligands × 183 mutants) pairings under the same conditions, making its exploration costly even under these stringent constraints. To alleviate this problem, we employed a random sampling strategy, selecting just over 15,000 pairings, or approximately 10% of the solution space. Despite the limited quantity, this sampling strategy was sufficient to make observations about the capabilities of the co-design approach.

The co-designed protein-ligand combinations spanned a wide range of AutoDock Vina binding scores and outnumbered those evaluated by individual design for any given affinity. Across the entire range of observed binding affinities, 86% or more of the pairings were derived from co-design; ligand and protein individual design contributed only up to 6% or 11% of the pairs, respectively. Furthermore, the pairs with the most improved binding affinities were only attainable by the co-design approach and not individual design (Figure 3.3). It should be noted that both ligand and protein design strategies were exhaustively enumerated, while the co-design space was only partially sampled. As such, it is likely that co-design could achieve an even greater contribution, were it not subject to the limitations of sampling.

3.2.3 Co-design identifies more favorable pairings

Compared with individual design approaches, co-design identifies more favorable protein-ligand pairings. Overall, co-design explores a larger selection of options, which results in higher standard deviation and reduced average performance as measured by both AutoDock Vina binding affinity and MM/GBSA free energy of binding. However, when selecting for the top 50 pairs by AutoDock Vina binding affinity, this pattern reverses. For example, the average binding affinity over top 50 pairings was -6.2 (± 0.6) kcal/mol and -6.5 (± 0.2)

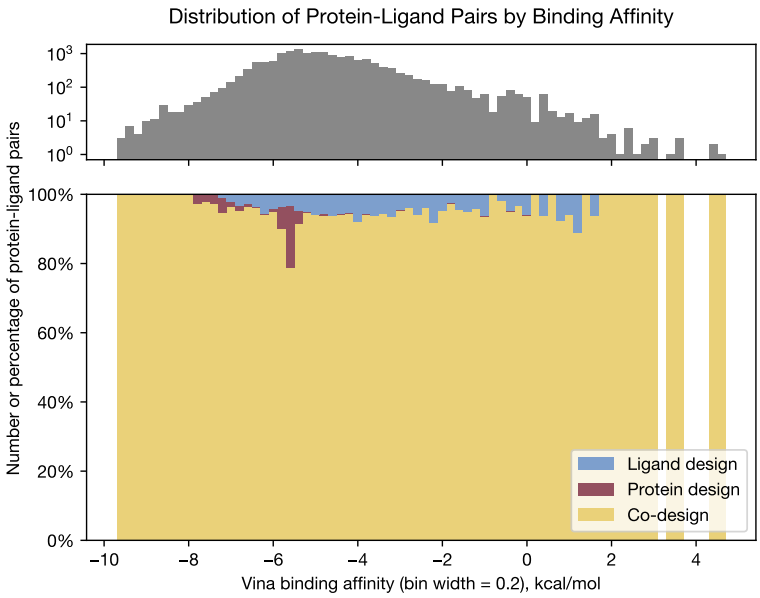

Figure 3.3 Distribution of protein-ligand pairs with respect to binding affinities. Top panel: overall number of pairs explored in this work. Bottom panel: relative percentage of protein-ligand pairs considered by each design strategy. The percentage of pairs evaluated by co-design (yellow) was significantly larger than that evaluated by ligand (blue) or protein (maroon) design, irrespective of the binding affinity, and the optimum affinities are only attainable via co-design.

kcal/mol for protein and ligand individual design, respectively. However, for co-design, the average AutoDock Vina binding affinity was more favorable, -8.9 (± 0.3) kcal/mol. This trend held for MM/GBSA free energy of binding values as well: the average free energy of binding over the same 50 pairings was -21.10 (± 7.69) kcal/mol and 9.31 (± 22.29) kcal/mol for protein and ligand design, respectively. Meanwhile, for co-design, the free energy of binding was once again more favorable, -34.83 (± 9.59) kcal/mol. Additional comparisons are provided in Table 3.1.

For all but three protein variants, the co-design approach improves AutoDock Vina binding affinity over individual protein design. Across all mutation sites and replacement amino acids, co-design commonly achieved an improvement of over -1 kcal/mol, with a maximum of -2.3 kcal/mol. For the V320R, I386E, and Y390R mutants, regressions of +0.4, +0.2, and +0.2 kcal/mol, respectively, were observed in AutoDock Vina binding affinities. In these cases, the best pairing found by co-design had lower interaction affinity than the

Table 3.1 Comparison of individual and co-design strategies. Average and standard deviation values are given for AutoDock Vina binding affinity and MM/GBSA free energy of binding; bold text indicates the best value in each category. Average performance across all pairings is better for individual protein design. However, the performance over the top 50 pairings selected by AutoDock Vina is significantly better for the co-design approach.

Metric	Design Approach	Top 50 pairs per Vina	All pairings
AutoDock Vina binding affinity score, kcal/mol	Individual, protein	-6.2 ± 0.6	**-5.6 ± 0.4**
	Individual, ligand	-6.5 ± 0.2	-4.5 ± 1.3
	Co-design	**-8.9 ± 0.3**	-4.6 ± 1.5
MM/GBSA free energy of binding, kcal/mol	Individual, protein	-21.10 ± 7.69	**-19.15 ± 11.65**
	Individual, ligand	9.31 ± 22.29	8.92 ± 32.32
	Co-design	**-34.83 ± 9.59**	12.48 ± 61.50

mutant-DHNA/menadione pair found by individual design. However, co-design could also propose the same pair, were it not limited by the number of random samples it was allowed to make. As such, this regression is not indicative of limitations of the co-design strategy.

3.2.4 Biochemical insights uncovered by co-design

The top 10 co-designed protein-ligand pairs ranged in binding affinities from -9.7 kcal/mol to -9.3 kcal/mol, which compared favorably to the best interaction affinities achieved by individual design approaches, -7.7 kcal/mol for protein design and -7.5 kcal/mol for ligand design. These pairings also had favorable MM/GBSA free energies of binding, ranging from -25.3 to -49.4 kcal/mol. The full list of the top 10 protein-ligand pairings is given in Table 3.2.

The top 10 pairings were categorized into three types based on the orientation of the underlying 1,4-naphthoquinone structure. Each type has a distinct appearance and we propose that different sets of hydrogen bonds may be present between the ligand and the protein. A distinguishing feature of Type 1 pairings is the placement of the ligand's amine group in close proximity to FAD and N387 side chain, within 2.7 Å and 3.4 Å, respectively. Type 2 pairs are characterized by a possible hydrogen bond between the K383 residue and the ligand with an approximate length of 3.7 Å. The Type 3 pairing is similar in its orientation to Type 1; however, its ligand lacks an amine group. Still, the distance between

Table 3.2 Favorable protein-ligand pairs suggested by the co-design approach. Ligands are identified by their IUPAC names as well as PubChem compound ids (indicated in parentheses). Binding affinity is reported by AutoDock Vina; free energy of binding is calculated using Prime MM/GBSA.

Rank	Ligand	Protein variant	Pair type	Binding affinity, kcal/mol	Free energy of binding, kcal/mol
1	2-Amino-3-(1-phenylethyl) naphthalene-1,4-dione (90135914)	I386D	1	-9.7	-25.3
2	2-(1-Hydroxy-1-phenylethyl) naphthalene-1,4-dione (23730319)	I386C	2	-9.5	-44.5
3	2-Amino-3-(1-phenylethyl) naphthalene-1,4-dione (90135914)	I386C	1	-9.5	-36.1
4	(2-Methylphenyl) 1,4-dihydroxy-5-methylnaphthalene-2-carboxylate (140286449)	I386S	2	-9.4	-36.0
5	(2-Methylphenyl) 1,4-dihydroxy-5-methylnaphthalene-2-carboxylate (140286449)	I386N	2	-9.4	-29.5
6	2-(1-Hydroxy-3-phenylpropyl) naphthalene-1,4-dione (71652350)	I386A	3	-9.4	-32.0
7	2-[Hydroxy(2-hydroxyphenyl)methyl] naphthalene-1,4-dione (245768)	I386S	2	-9.3	-41.8
8	(4-Hydroxyphenyl)1,4-dihydroxy-3-methylnaphthalene-2-carboxylate (156840197)	I386G	1	-9.3	-45.5
9	2-(2-Phenylmethoxypropan-2-yl) naphthalene-1,4-dione (139943538)	I386G	2	-9.3	-37.6
10	2-[Hydroxy(phenyl)methyl] naphthalene-1,4-dione(245925)	I386G	2	-9.3	-46.6

FAD and the ligand is 4.3 Å, raising a possibility of weak electrostatic interaction. All three types may also allow hydrogen bonding with Q321 residue, with minimum residue-ligand distances ranging from 2.9 Å to 3.6 Å. All ligand atoms involved in those possible hydrogen bonds were in close proximity to electron acceptor features identified by the pharmacophore modeling of the ligands using Schrödinger Maestro [60]. Figure 3.4 shows the three types of pairings in greater detail.

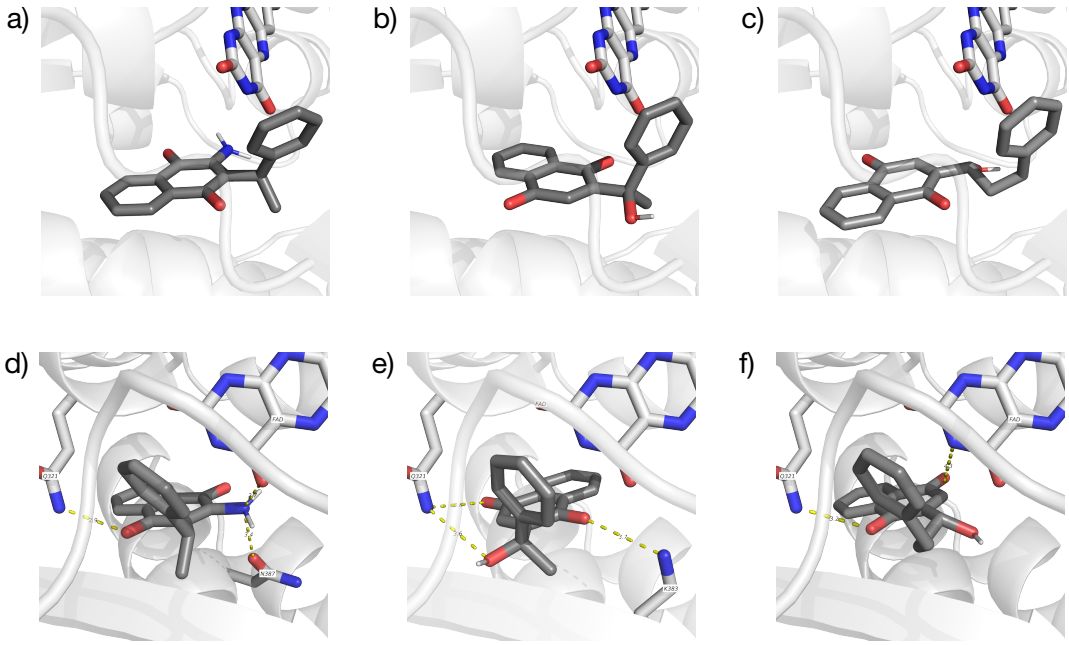

Figure 3.4 Three types of pairings observed among the top 10 co-designed protein-ligand pairs. (a-c): Type 1, 2, and 3 ligand poses seen from the same perspective, showing their relative orientation. (d-f): magnified view showing possible interactions involved in each of the Types 1, 2, and 3, respectively. The potential hydrogen bonds are shown with yellow dashed lines and numbers indicating the distance.

Mutation of the I386 residue is a common feature of the most favorable pairings found by both the co-design and individual protein design strategies, suggesting it's an important contributor to successful binding. This residue is part of the structure making up the quinone binding pocket and is in close contact with the ligand. In the wildtype Ndh2 sequence, this position is occupied by isoleucine, a large neutral amino acid. In mutants with improved affinity, this residue is commonly replaced by aspartic acid or a neutral polar amino acid

(cysteine, serine, or asparagine). Such mutations replace a hydrophobic amino acid with a hydrophilic one. A handful of mutants replace this residue with alanine or glycine: this change has little effect on hydropathy [117], but it moves the interaction further away from the ligand as the alanine and glycine side chains are significantly smaller than that of isoleucine. As such, it appears likely that hydropathy at this location within the binding pocket is a key factor for affinity.

The distribution of binding affinities across all sampled co-designed pairings reveals several trends. On the protein side, binding affinity was most improved by the mutation of residue 386 (Figure 3.5a) and the use of aspartic acid or cysteine as replacement residues (Figure 3.5b). On the ligand side, the largest improvement was obtained by adding functional groups at positions 5 and 15 on DHNA and 3 and 12 on menadione (Figure 3.5c); however, no clear trend could be seen with respect to specific functional groups (Figure 3.5d). Nevertheless, co-design resulted in improved binding affinities over individual design under all types of protein and ligand modifications.

3.3 Conclusion

This work presents a sampling-based approach for co-designing protein-ligand pairs and its application for enhancing Ndh2-quinone-mediated EET in *L. plantarum*. Unlike traditional individual design approaches, our method jointly explores protein-ligand combinations, yielding pairings with enhanced Vina binding affinities and MM/GBSA predicted free energies of binding. Empirical review of the resulting combinations uncovered specific protein mutations and ligand modifications that strengthen the binding capabilities of Ndh2 and quinones. The sampling strategy of the combined protein-ligand space provides a representative distribution of the pairings and highlights potential improvement. The strategy also highlights the difficulty of optimizing for binding affinity as most combinations exhibit a narrow range of binding affinities close to the wildtype interaction, while a number of pairings present less favorable affinities, and a smaller fraction yet shows an improvement.

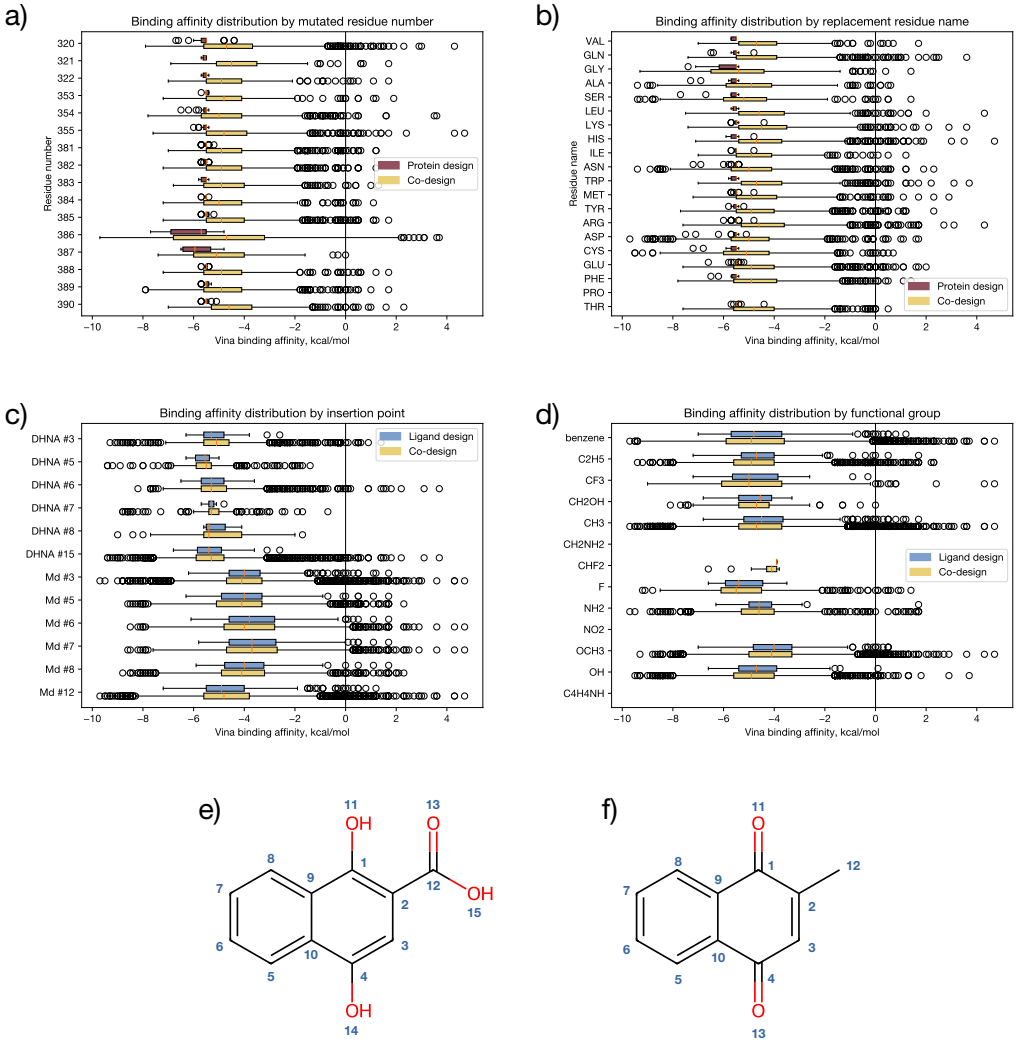

Figure 3.5 Distributions of binding affinities for individual and co-design strategies, broken down by (a) mutated residue number, (b) replacement residue name, (c) addition location on the base quinone, and (d) added functional group. Atom numbering scheme is given for (e) DHNA and (f) menadione.

Chapter 4 - Using graph neural networks for site-of-metabolism prediction and its applications to ranking promiscuous enzymatic products

The appropriate choice of model architecture and data curation strategy can learn effective classifiers for biological problems using relatively small datasets. In this chapter, we present *GNN-SOM*, a GNN-based approach for predicting SOMs of enzymatic reactions. *GNN-SOM* learns effective molecular representations from a small dataset of KEGG RPAIR [6] reactions by treating the problem as an atom- or bond-classification task. These representations are derived from simple atom properties and the topology of the molecule and obviate the need for complex molecular descriptors seen in other SOM prediction methods. A comprehensive data pre-processing strategy is introduced to ensure consistency and correctness of the dataset. We evaluate several GNN architectures and non-GNN baseline models, and apply the resulting *GNN-SOM* approach to two biological test cases.

4.1 Methods

4.1.1 Problem formulation for SOM prediction

As in prior ML methods [23, 24, 20, 21, 27], the SOM prediction problem can be formulated as a binary classification task. We formally define the *atomic SOM* prediction problem as follows: given a query enzyme and a query molecule, the model predicts the likelihood of each *atom* being an SOM. Similarly, we define the *bond SOM* prediction problem as a bond

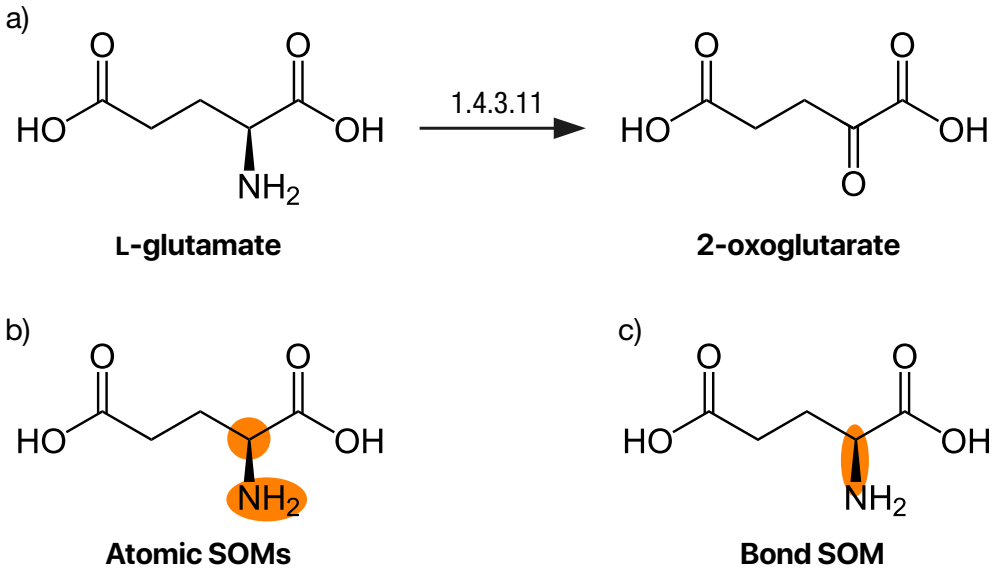

Figure 4.1 An example of a reaction and the corresponding sites of metabolism. (a) In this biotransformation, the enzymatic activity of glutamate-oxidase (EC 1.4.3.11) causes the amine group in L-glutamate (left) to be replaced with a carbonyl group, leading to the formation of 2-oxoglutarate (right). (b) On a structural level, the biotransformation converts a single bond to a double bond, in addition to replacing a nitrogen atom with an oxygen atom. As a result, both atoms (oxygen and nitrogen) across the modified bond are considered *atomic SOMs*. (c) Alternatively, the modified bond itself could be considered a *bond SOM*.

classification problem, where we predict the likelihood of each *bond* being an SOM. Figure 4.1 provides an example of atomic and bond SOMs.

Each molecule is represented using an undirected graph $G = (V, E)$, where every node $i \in V$ represents an atom and every edge $(i, j) \in E$ for nodes $i, j \in V$ represents a bond. The atomic SOM prediction problem then becomes a *node* classification task. Given a representation $\mathbf{x}_i$ for atom $i \in V$, we seek to find its predicted SOM label $\hat{y}_i$.

We formulate the bond SOM prediction problem similarly to a link prediction task: the objective is to make the determination $\hat{y}_{i,j}$ whether a bond SOM exists between a pair of atoms $i, j \in V$ with their respective representations $\mathbf{x}_i$ and $\mathbf{x}_j$. The link prediction is performed by applying a multi-layer perceptron (MLP) to a concatenation of the two atom representations. Since there is no order preference on the atoms, this calculation is performed on both $\mathbf{x}_i \| \mathbf{x}_j$ and $\mathbf{x}_j \| \mathbf{x}_i$ arrangements and the reported prediction is the average of the two. We therefore calculate this prediction as follows:

$$\hat{y}_{i,j} = \frac{\text{MLP}(\mathbf{x}_i \| \mathbf{x}_j)}{2} + \frac{\text{MLP}(\mathbf{x}_j \| \mathbf{x}_i)}{2} \tag{4.1}$$

4.1.2 Representation learning using GNNs

For learning node representations, we consider GNNs that consist of a series of layers operating on atom-level embeddings. Such layers can be used to directly make per-atom SOM predictions as in Figure 4.2a, or using a separate bond classifier MLP for bond SOM predictions as in Figure 4.2b. Each convolution layer extracts local substructure features for individual nodes and learns a compact representation thereof. We evaluate several GNN message passing layer types for predicting SOMs. The graph convolution operator from Graph Isomorphism Networks (GINs) [118] is used as a representative example of a spatial GNN layer. In that work, the authors noted that any aggregation based GNN is at most as powerful as the Weisfeiler-Lehman test and proposed an architecture that generalizes this test. Given input node features $\mathbf{x}$, transformed features $\mathbf{x}'$ are calculated for every node i as follows:

$$\mathbf{x}'_i = \text{MLP}((1+\epsilon)\mathbf{x}_i + \sum_{j \in N(i)} \mathbf{x}_j) \tag{4.2}$$

where ϵ is a learnable parameter and $N(i)$ is the set of neighbors of node i. In this framework, an MLP is applied to the linear combination of features around the node with the aim of generating unique representations for different neighborhoods.

We also consider a GNN convolution layer designed to mimic circular molecular fingerprints [119]. Circular fingerprinting identifies important substructures in a molecule and encodes them in a binary vector. The conventional algorithm for generating such fingerprints features several non-differentiable operations on node features, most notably concatenation and hashing. In their GNN rendition of the molecular fingerprinting (MF) GNN concept, the authors replaced the concatenation by a summation over node features; the hashing was

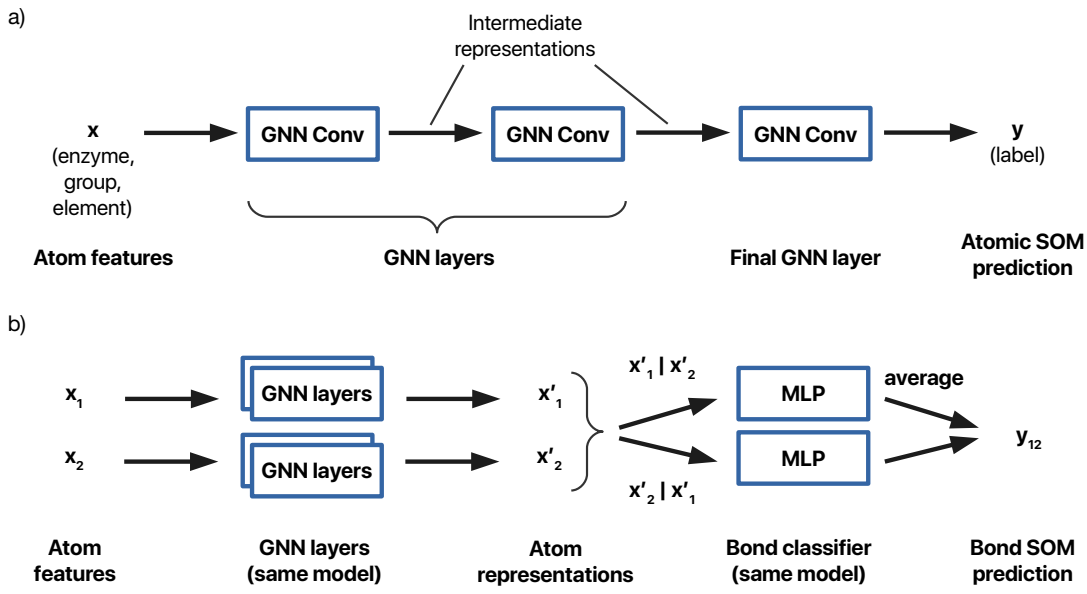

Figure 4.2 Architecture of GNN-SOM models. (a) GNN-SOM for atomic SOM prediction. Atom features $\mathbf{x}$ are processed through a number of GNN layers, each generating intermediate representations of a fixed size. The last intermediate representation is used as the input to the final GNN layer that generates a single value representing the SOM prediction for that atom. (b) GNN-SOM for bond SOM prediction. Atom features for both endpoints of a bond, $\mathbf{x}_1$ and $\mathbf{x}_2$, are processed through the same GNN layers as before; however, the final GNN layer is excluded. The resulting intermediate representations, $\mathbf{x}'_1$ and $\mathbf{x}'_2$, are concatenated two different ways and provided to a bond classifier MLP. The MLP makes two predictions, one for each concatenation, and the average of those represents the final SOM prediction for the bond.

replaced by a multiplication of node features by degree-specific learnable weight matrices – one for a node's own features, $\mathbf{W}$, and one for the sum of its neighbors' features, $\mathbf{H}$. For organic molecules, the maximum number of degrees is expected to be five. The resulting convolution is defined as follows:

$$\mathbf{x}'_i = \mathbf{W}^{(\deg(i))}\mathbf{x}_i + \mathbf{H}^{(\deg(i))} \sum_{j \in N(i)} \mathbf{x}_j \tag{4.3}$$

The key difference between the MF GNN and GIN is its variability with respect to a node's degree, or in other words, its neighborhood size. The molecular fingerprinting convolution learns a separate linear mapping of node features to representations for every

degree. The GIN approach, on the other hand, learns one mapping for all degrees. With more learnable parameters in the model, MF GNNs can capture richer representations.

The third GNN architecture leverages the Chebyshev convolutional operator, originally proposed by [66]. Unlike the previous two approaches, this spectral graph method considers a wide portion of a graph at any given GNN layer. It uses a K-polynomial filter that integrates features within the K-hop neighborhood of a node, which in turn helps it achieve good localization in the node domain [63]. Our approach relies on the following convolution definition [120]:

$$\begin{aligned} \mathbf{x}' &= \sum_{k=1}^{K} \mathbf{Z}(k) \cdot \mathbf{\Theta}_k \\ \mathbf{Z}(1) &= \mathbf{x} \\ \mathbf{Z}(2) &= \mathbf{L} \cdot \mathbf{x} \\ \mathbf{Z}(k) &= 2 \cdot \mathbf{L} \cdot \mathbf{Z}(k-1) - \mathbf{Z}(k-2) \end{aligned} \tag{4.4}$$

where the summation above represents the K-polynomial filter, $\mathbf{Z}(k)$ is the Chebyshev polynomial of order k approximating the spectral graph convolution [63], $\mathbf{\Theta}$ represents the learnable parameters of the filter, $\mathbf{L}$ refers to the normalized graph Laplacian, and K is the Chebyshev filter size. The features at the input of the layer are given by $\mathbf{x}$.

For all three GNN models, the node classification task is implemented by inserting an additional graph convolutional layer, accepting $\mathbf{x}_i$ as the input and generating $\hat{y}_i$ as the output, while the edge classification task is implemented using an MLP on the concatenation of two atom representations $\mathbf{x}_i$ and $\mathbf{x}j$ as described earlier.

The models are trained using Adam optimization [121], with binary cross-entropy as the loss function. Hyperparameters are tuned using the grid search approach. Ranges for the hyperparameters were selected manually based on their expected effects and practical considerations for runtime. The hyperparameters include the size of the latent representations

(set to 64, 128, 256 or 512) and the number of GNN layers (ranging from 1 to 5). In addition, we adjusted the filter size (from 1 to 10) for GNNs for the Chebyshev convolutional operator, and maximum number of degrees (from 1 to 5) for models using the molecular fingerprinting convolution. The models were constructed using the convolutional layer implementations provided by PyTorch Geometric [120].

4.1.3 Baseline models

Random Forest, MLP and AdaBoost are selected as baseline models for the same classification task. These models have been used prior for SOM prediction in on reaction-specific datasets and can be considered representative examples of the current state of the art in SOM prediction. For example, Random Forest (RF) classifiers form the basis of the MetScore [20] and FAME [21] methods. XenoSite [27] and Rainbow XenoSite [23] utilize MLP models for their prediction task. Finally, the work of He et al. [24] leverages a collection of ML techniques, including RF and AdaBoost. We use the same input feature vectors and datasets for both our GNN and baseline models, therefore allowing for a fair comparison between baselines and GNN models.

The MLP models are trained using Adam optimization with cross-entropy loss. For Random Forest models, we used Gini impurity as the splitting algorithm. The SAMME.R boosting algorithm was used in building the AdaBoost models. The models were constructed using the scikit-learn package. Hyperparameter tuning is performed using grid search. For MLP, we varied the size of the latent representations (set to 32, 64, 128, 256 or 512) and the number of hidden layers (from 1 to 5). Meanwhile, for both Random Forest and AdaBoost, we adjusted the number of decision trees (set to 100, 250, 500 or 1000).

4.1.4 Dataset construction

We derive our atom SOM dataset from the KEGG RPAIR database [6], a collection of atom-mapped reactant pairs associated with specific transformation patterns. Each RPAIR

has cross-references to reactions in the KEGG database as well as the relevant Enzyme Commission (EC) numbers. The transformation patterns are encoded using R, D and M (RDM, for short) atom-level tags. "R" distinguishes the reaction center atoms, which we assume to be the sites of metabolism. The "D" (difference) tag points to molecular substructures modified by the biochemical transformation, while the "M" (match) tag refers to substructures neighboring the "R" tagged atom that remain unchanged.

We create separate versions of the dataset, one for the atomic-SOM problem, and one for the bond-SOM problem. To create the atom-centric version of the dataset we assign SOM labels to every atom at the reaction center of an RDM pattern. To create a dataset suited for bond SOM prediction, we identify bonds that change due to a chemical transformation and label them as SOMs instead. As this dataset is not readily available, we curate the KEGG RPAIR database to generate such a set. The dataset was further processed to improve SOM labeling. Some adjustments are specific to a given version of the dataset. A symmetry adjustment is applied to both datasets to account for molecular symmetrical features.

The feature vectors representing atoms consist of three components, which include the atom elemental type, the KEGG atom type (an atom label that represents the atom type and the atom's relationships to nearby elements and bonds), and the first two levels of the Enzyme Commission (EC) number of the enzyme associated with the transformation.

Dataset with atom SOMs

Node features consist of three components: the KEGG atom type, the atom elemental type, and the EC number corresponding to the metabolizing enzyme. The KEGG atom type represents the atom's immediate chemical environment in the form of a single label. These types are defined by a set of 68 small-scale molecular substructure patterns [122], which we further extend to distinguish between atoms connected by single and double bonds (e.g. oxygen and hydroxy group in carboxylic acid). Although there are separate KEGG atom types for different elements, this semantic structure is not conveyed by their

feature representation. Therefore, we include a node feature to describe the atom's element separately. The EC number is also incorporated as a node feature. The EC classification system provides a hierarchy of four levels, allowing enzyme behaviors to be represented to various degrees of specificity. Highly specific levels provide more information about the metabolizing enzyme, albeit with the disadvantage of an increased feature complexity and a diminished number of training examples per category. As a compromise, we use the first two levels of the EC system. All three components of the feature vector correspond to separate categorical labels: as such, they are represented using a one-hot encoding scheme.

The use of KEGG atom types allows for streamlined models with wide applicability. Bond properties such as order, strength, and type are all implicit to various extent in the functional group labeling provided by the types, therefore there is no need to encode and process edge features. Furthermore, because the KEGG atom types are reproducible with publicly available definitions [122] and implementations such as KCF Convoy [123, 124], models developed here can be easily applied to non-KEGG datasets.

Dataset with bond SOMs

Many reactions result in the creation of new bonds that can occur between existing atoms or atoms not originally present in the molecule, for example in ring closure reactions and additions of new functional groups, respectively. To allow these bonds to be considered as potential SOMs, each atom in a molecule is paired with a unique synthetic placeholder node such that the edge between the two represents bonds that may be created at that location. Labeling of bond changes is then done in three stages. First, an atom-to-atom mapping between the substrate and the product of an RPAIR is produced. This mapping is created based on the molecular alignment information provided with the RPAIR database with a modification described below. Any difference-tagged atoms in one of the molecules in an RPAIR are mapped to placeholders in the other molecule. Since there may be multiple such atoms and the mapping is one-to-one, we initially create four placeholders at every atom – a

number sufficient to map atoms in every existing RPAIR.

The next step involves checking atoms and bonds aligned by this mapping for changes. We consider three types of differences as indicators of bond SOMs: 1) changes of element at one of the endpoints of a bond, including a change to or from a placeholder, 2) changes in the bond order, and 3) differences in the formal charge of an atom. The latter is encoded as a change in one of the placeholder nodes, which in this case represents a proton or an electron that is being added or removed. Any bonds where such changes occur are labeled as the SOM bonds.

The four placeholder nodes situated at a given atom are indistinguishable from one another: the additional copies were created only to facilitate a one-to-one mapping and are otherwise not needed. The last step, therefore, entails combining the placeholder nodes such that only one remains per atom. The SOM bond labels associated with the placeholder nodes are merged so that if one of the duplicates is marked as a positive SOM, the final merged bond label would be marked the same, too. As a result, all SOMs detected in the previous stage are preserved in the final molecules.

Dataset adjustments to improve SOM labeling

The node-centric dataset for the atomic-SOM problem is corrected for cleaved bonds. In cases when a bond is broken in a molecule, often only one of the endpoint atoms are marked as the reaction center of the transformation pattern, with no apparent consistency as to which atom that would be. However, both atoms should be considered as the SOM, as they are located at the site of the transformation and are involved to an equal extent. To this end, we track bonds that are cleaved over the course of the reaction, and if one of the end points of such a bond is marked as the reaction center, we make sure to mark the other one as well. Figure 4.3 shows an example reaction where a cleaved bond results in the creation of additional SOMs.

The edge-labeled SOM dataset uses molecular alignment information derived from the

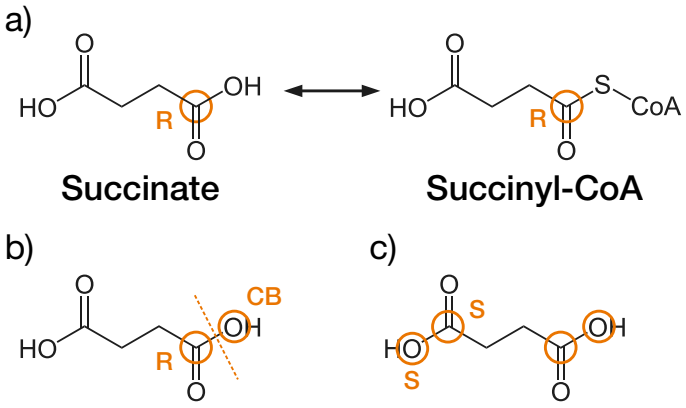

Figure 4.3 Adjustments for cleaved bonds and symmetry demonstrated by a reaction converting between Succinate and Succinyl-CoA. (a) A biotransformation with the reaction center labeled as "R", where the hydroxy group is removed and replaced by a sulfur atom. The hydroxy group in Succinate and the sulfur atom in Succinyl-CoA could be considered as SOMs. (b) The cleaved bond in Succinate and the additional atomic SOM, which is labeled as "CB" (for "cleaved bond"). (c) Two additional atomic SOMs, labeled as "S" (for "symmetry") that are implied by the symmetrical structure of the molecule.

KEGG database. To support our modified KEGG atom types with bond order suffixes, we introduce a preprocessing step to ensure that the molecular alignment has the appropriate atom mapping. The molecular pairs are scanned, searching for mappings with the modified atom types. The mapping is checked to verify the modified types around a common neighbor are mapped without requiring changes in the bond order suffix. If such case is detected, the mapping is reversed, resulting in an accurate alignment of the modified atom types, consistent throughout the dataset.

Another modification of the alignment information is performed with the goal of simplifying the mapping. In some molecules, the replacement of an atom was recorded as the removal of one substructure and the addition of another with no mapping in between, as opposed to a direct replacement. Despite being an identical operation, the former would produce additional SOMs: because one atom on each side of the reaction has no mapped counterpart on the other, each would be mapped to a placeholder node, resulting in placeholder bonds being marked as the positive SOMs as well. In both cases, however, the bonds with the replaced atom would be marked on both the substrate and the product of the

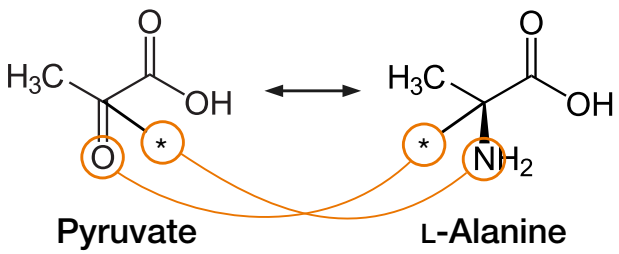

Figure 4.4 Interchange adjustment as demonstrated by a reaction converting between Pyruvate and L-Alanine. In this case, the biotransformation is encoded such that the oxygen atom is removed in Pyruvate and the amino group is added in L-Alanine. Because no direct correspondence is provided between the two, normally each would be mapped to a placeholder atom (only relevant ones shown), resulting in two bond SOMs detected per molecule. However, an equivalent transformation could be obtained by replacing the oxygen atom with the amino group directly, in which case only one bond per molecule would be considered as SOM.

reaction. To ensure such atom replacements are consistently handled as replacements with the simplest possible mapping, we identify those cases and correct them accordingly. Figure 4.4 shows an example reaction where this modification would apply.

Once all SOMs are selected, the final adjustment entails correction for symmetry. Many molecules have symmetrical features, where multiple sites are structurally identical given the context of the molecular graph. Such features may be as large as half of the entire molecule (e.g., Bisphenol A) or as small as one atom (oxygen atoms in the outermost phosphate group of ATP and other molecules). Equivalent sites are indistinguishable from one another and therefore, should have the same SOM classification. Yet only some of the symmetrical sites may be marked as the SOMs based on the genericized transformation pattern or the available alignment information, providing conflicting information during training or evaluation stages of model design. This issue was previously recognized and handled in MetScore [20], but no detailed procedure for this adjustment was provided.

To address the SOM symmetry issue in the dataset, we include a post-processing step that mirrors labeled reaction centers to all symmetric locations as follows. Given a molecular graph, we first assign vertex colors as a function of the KEGG atom type augmented with the bond order suffix. This operation distinguishes atoms by their elements as well as the

types of bonds around them. Then, we compute all automorphism groups of the molecular graph – permutations of same-colored atoms that keep the set of bonds exactly the same – using the nauty program [125]. While graph isomorphism is not solvable in polynomial time in the general case, the molecules we consider do not pose a significant computational challenge: the calculation of automorphism groups for all molecules in practice takes only a few minutes. Furthermore, most organic molecules are planar graphs [126], for which there exists a linear time graph isomorphism algorithm [127]. Finally, we check the sets of vertices that are equivalent by some automorphism for reaction center labels. If such a set contains a labeled SOM, we consider all vertices in the set to be SOMs also.

This symmetry adjustment is applied to both node and edge versions of the dataset. In case of the edge-centric set, the same algorithm is applied to the line graph of the molecule, where every bond is converted to a line graph node. Associations between bonds are preserved by creating line edges for every pair of bonds sharing a common atom. The line graph nodes are colored based on the elements of the atoms connected by the bond and its order. Because the four placeholder nodes located at a given atom are likewise identical by symmetry, this adjustment provides for a convenient way to implement the merging of placeholder nodes used in the edge-labeled SOM dataset.

4.2 Results

4.2.1 Data splits

To assess the effectiveness of the models on the available dataset (21,023 molecules), we employ a cross-validation scheme. We create ten different data splits, with 80% of molecules allocated for training, 10% for validation, and the remaining 10% for testing. The partitions are generated by performing shuffle splits on the set of biotransformations, which ensures there is no information leakage across the substrate and product sides of a reaction or across enzymes associated with the same transformation. The set of training molecules is used to train several versions of a model with different hyperparameter combinations. The best

combination is then selected based on its performance on the validation set. Finally, the selected model is evaluated on the test set. This process is repeated ten times, once for each data split, and we report the average test set performance for each method.

This approach yields ten different models with varying training set molecules and hyperparameters. For use in biological applications, we create a combined model that applies all ten models to a query molecule and reports the average predicted SOM probabilities for each atom.

4.2.2 GNN model evaluation

In our evaluation of various models, we considered several performance metrics, including area under the ROC curve (AUROC) and R-precision. AUROC evaluates the model's ability to rank SOM sites higher than any site where metabolism is not known to occur. R-precision measures the fraction of SOMs present among the top R predictions, where R is the total number of SOMs in the molecule or the test set.

We calculated AUROC and R-precision using two different methods. The *molecular* benchmarks were defined to be the average value of the corresponding metric when calculated on a per-molecule basis. This represents the expected performance in applications where the objective is to identify SOMs in a specific queried molecule. The alternative approach is to calculate AUROC and R-precision on the full set of atoms where all molecules are pooled together. Such *atomic* measurements are representative of a model's performance in scenarios where the goal is to locate the most likely SOMs in a group of molecules. We also considered the *top-two* correctness rate, a molecular metric that is frequently used to evaluate the performance of CYP SOM predictors [23, 27]. It represents the proportion of molecules where a known SOM appears among the molecule's top-two predicted sites. Therefore, in contrast to the other measures, it "saturates" and does not require a completely correct ranking for a perfect score to be assigned to a molecule. Several versions of each model with different hyperparameters were evaluated. Although there were five performance

metrics that we considered important, they were all strongly correlated with one another. As such, model selection was performed by maximizing the molecular R-precision on the validation set.

Model evaluation is reported for the best-in-class GNNs (Table 4.1a). GNNs using the Chebyshev convolutional operator achieved the best performance by all five metrics, with models based on the molecular fingerprinting convolution being a close second. Performance of GIN-based models, however, was markedly worse. For the remainder of the discussion, we refer to the GNN that utilizes the Chebyshev convolutional operator as *GNN-SOM*.

Table 4.1 GNN and non-GNN model evaluation. (a) Different GNN models. (b) Performance on the original dataset that includes the KEGG atom types (c) Model performance with the removal of the KEGG atom types. (d) Expected performance that would be achieved on this dataset via random guessing. Standard deviation for all listed values is 0.02 or less.

	Molecular R-Precision	Molecular AUROC	Top-2 correctness rate	Atomic R-Precision	Atomic AUROC
a) GNN model evaluation					
GNN-SOM (GIN)	0.676	0.915	0.812	0.635	0.939
GNN-SOM (MF)	0.764	0.946	0.851	0.731	0.963
GNN-SOM (Cheb)	**0.789**	**0.953**	**0.868**	**0.771**	**0.971**
b) Models with KEGG atom types					
RF	0.520	0.850	0.716	0.469	0.872
Ada	0.520	0.850	0.713	0.470	0.871
MLP	0.519	0.850	0.714	0.470	0.871
GNN-SOM	**0.789**	**0.953**	**0.868**	**0.771**	**0.971**
c) Models without KEGG atom types					
RF	0.294	0.649	0.501	0.276	0.713
Ada	0.295	0.650	0.502	0.276	0.713
MLP	0.297	0.651	0.505	0.275	0.711
GNN-SOM	**0.749**	**0.934**	**0.849**	**0.725**	**0.963**
d) Expectation via random guessing					
Expectation	0.111	0.500	0.111	0.044	0.500

4.2.3 GNN models outperform non-GNN models

We compared the best GNN model, *GNN-SOM*, to several baseline approaches. *GNN-SOM* outperforms baseline methods on all five considered metrics (Table 4.1b). The use of GNNs

led to an approximately 10% improvement in molecular and atomic AUROCs and as much as a 30% increase in atomic R-precision.

We postulate that the limited information about an atom's local chemical environment is a major factor responsible for performance differences. To evaluate the contributions of the KEGG atom types, we removed the node feature responsible for this property and observed a 15–20% reduction of the baseline models' evaluation metrics, while the performance of GNN-based models was only minorly affected (Table 4.1c). Baseline models are able to utilize the KEGG atom types as a part of the node feature vector, while GNNs can natively infer it from the graph structure.

We provide the expected performance achievable via random guessing as a means of evaluating the inherent complexity of the prediction problem (Table 4.1d). A binary classifier making random guesses with no bias towards any specific class achieves AUROC of 0.5 [128]. Given R relevant SOMs and N total SOM candidates, the expected R-precision and top-two correctness rates for a set of atoms were calculated as follows:

$$\text{R-precision} = \frac{1}{R}\sum_{i=0}^{R-1}\frac{R-i}{N-i}, \tag{4.5}$$

$$\text{Top-two} = \frac{1}{2}\sum_{i=0}^{1}\frac{R-i}{N-i}$$

In each iteration of a summation, we computed the expected number of selected SOMs from the set of the remaining available candidates. For R-precision, we considered the total number of sites selected among the top R prediction attempts and divided it by the number of attempts made, following the definition of the metric. For top-two correctness rate, we found the probability of selecting an SOM within the first two attempts using a similar approach.

4.2.4 Bond SOM prediction and atomic SOM prediction

We found that the bond SOM prediction problem is inherently more difficult than the atomic SOM prediction, with limited advantages offered in the way of combining the two approaches. The *expected* performance metrics, with the exception of AUROC, are lower for the bond SOM prediction task by approximately a factor of two: this is a consequence of there being about twice as many SOM edge candidates in the bond prediction problem with no proportional increase in the number of true sites. To quantify the potential benefits of combining the two problems for more accurate predictions overall, we compared molecular R-precision values achieved by the best models for each data split, counting the number of molecules where the performance of one model exceeded that of the other or where there was a tie. On average across the ten data splits, the node-centric model achieves better molecular R-precision for 733.3 molecules (SD 40.6), the edge-centric model outperforms in 172.3 (SD 25.4) cases and the remaining 1202 (SD 46.0) molecules experienced identical performance on the two models. Therefore, in most cases, the node-centric model would be the ideal choice, with it providing better predictions for a larger number of molecules. The comparison of the overall performance of the two types of models is provided in Table 4.2.

Table 4.2 Performance of GNN-based models on node-centric and edge-centric versions of the dataset. Standard deviations are 0.02 or less.

	Molecular R-Precision	Molecular AUROC	Top-2 correctness rate	Atomic R-Precision	Atomic AUROC
GNN-SOM	0.789	0.953	0.868	0.771	0.971
Expectation, atom	0.111	0.500	0.111	0.044	0.500
GNN-SOM-Bond	0.612	0.940	0.777	0.543	0.946
Expectation, bond	0.059	0.500	0.062	0.024	0.500

4.2.5 CYP versus non-CYP prediction

To evaluate the applicability of our model across different types of reactions, we investigated performance of our models on interactions mediated by CYP and non-CYP enzymes. We

found that CYP-specific reactions are more challenging than non-CYP interactions for all models. When trained and tested on one type of reaction, GNNs and baseline methods alike performed significantly better on non-CYP interactions despite the expected performance of a random classifier being similar in both sets (Table 4.3a and 4.3b). Furthermore, the models specific to non-CYP reactions—unencumbered by the more challenging CYP interactions—demonstrated slightly better performance compared to general models (Table 4.2) on their respective molecule sets. We believe this difference was in part due to a comparatively small number of CYP reactions available for training: in our dataset, there were 18,139 non-CYP reactions and 2,877 CYP-associated reactions. However, the performance gap between the CYP and non-CYP reactions was less pronounced for our GNN approach compared to the baseline ML techniques, indicating that GNNs can more effectively learn a wide range of SOMs when training data is limited. As before, the GNNs consistently outperformed the baseline models given the same circumstances.

Table 4.3 Performance of GNNs and baseline models on CYP and non-CYP mediated interactions separately. Standard deviations are 0.03 or less.

	Molecular R-Precision	Molecular AUROC	Top-2 correctness rate	Atomic R-Precision	Atomic AUROC
a) Models trained and tested on only CYP reactions					
RF	0.359	0.723	0.573	0.359	0.758
Ada	0.358	0.725	0.575	0.360	0.761
MLP	0.362	0.728	0.575	0.358	0.758
GNN-SOM	**0.693**	**0.912**	**0.767**	**0.695**	**0.955**
Expectation	0.108	0.500	0.112	0.051	0.500
b) Models trained and tested on only non-CYP reactions					
RF	0.545	0.870	0.731	0.488	0.885
Ada	0.546	0.870	0.734	0.489	0.885
MLP	0.547	0.870	0.737	0.488	0.886
GNN-SOM	**0.799**	**0.958**	**0.878**	**0.782**	**0.973**
Expectation	0.111	0.500	0.111	0.044	0.500
c) Models trained on all reactions but tested only on CYP or non-CYP					
GNN-SOM, CYP	0.690	0.910	0.775	0.694	0.956
GNN-SOM, non-CYP	0.804	0.959	0.883	0.784	0.973

We also found that our general-purpose SOM prediction model is as effective as the

CYP and non-CYP-specific models. The SOM predictor trained on all available interactions, achieves very similar performance on the CYP and non-CYP subsets compared to models trained on each reaction type specifically (Table 4.3c). As such, our approach alleviates the necessity for separate predictors for CYP and non-CYP interactions.

4.2.6 Applications for SOM prediction

While several techniques predict overall reaction feasibility, such as thermodynamic feasibility (e.g. eQuilibrator [129, 130]), or likelihood of a biochemical conversion between a substrate and a product (e.g. DeepRFC [131] and ELP [132]), SOM prediction determines the likelihood of an enzyme class acting on a particular atom or bond within a molecule. Therefore, when paired with a rule-based method, the SOM likelihood can be assumed a proxy for the likelihood of reaction occurrence. We demonstrate the utility of SOM prediction using two applications: screening promiscuous products generated using rule-based prediction methods, and ranking synthesis pathways based on SOM likelihood of each pathway's individual reaction steps. While we selected PROXIMAL [133] as our companion rule-based method, our *GNN-SOM* method is independent of PROXIMAL, and can be used with other rule-based techniques as well.

Screening promiscuous products generated from rule-based prediction methods to create EMMs

To evaluate *GNN-SOM*'s utility in improving prediction of products arising due to enzyme promiscuity, we leveraged *GNN-SOM* as a screening step for rule-based product prediction methods to eliminate unlikely promiscuous products. We chose to evaluate the impact of including this step on creating EMMs [134], which are intended to account for promiscuous enzymatic activities. In that work, the *i*ML1515 model of *E. coli* [135] was extended with a number of reactions that could arise due to uncatalogued promiscuous enzyme activity. The PROXIMAL tool was used to predict interactions occurring between native enzymes

and substrates listed in *i*ML1515. The resulting putative products were then searched in ECMDB [136, 137] and PubChem to identify promiscuous transformations that were observed previously but are missing from the *i*ML1515 model. Balanced reactions were then constructed for each transformation, and after manual curation, 23 new reactions were recommended for addition into the model.

PROXIMAL generates products via application of RDM patterns on molecular graphs of queried substrates: it does not consider SOMs. As a result, it may propose unfeasible or unlikely biotransformations. Using SOM predictions as a criterion for applying patterns can help lower the number of such unfeasible products, potentially reducing the amount of manual curation required. We use our SOM predictor as a filter on the products. For a biotransformation to be accepted, the reaction center in the substrate at which PROXIMAL applied the pattern must be at or above a certain threshold. If the reaction center is below that value, we consider the transformation to be unlikely and discard it.

Our application of 1,875 PROXIMAL operators presented by Amin et al. to 106 high-concentration metabolites [138] in *E. coli* yielded a set of 1,989 products, 1,390 of which we could cross-reference between the PubChem online REST API and the derivative products presented in the publication. Additionally, there were 55 products that were cross-referenced to ECMDB. Such products were considered to be "verifiable" since there was concrete evidence of their existence in *E. coli*. In comparison, predicted products merely found in PubChem are less likely to occur in *E. coli*, since this set includes a great many metabolites never observed in the organism before. As such, the ratio between the number of compounds confirmed by both databases and PubChem only, represents a metric of interest. Thus, retaining a higher percentage of ECMDB-confirmed products when filtering using *GNN-SOM* compared to the overall predicted products showcases the utility of *GNN-SOM*.

Instead of examining the number of products under a fixed threshold (likelihood), we explore how the threshold impacts the number of products. We applied a range of filter thresholds from 0.0 to 1.0 in 0.1 increments (Figure 4.5). As the threshold increases, fewer

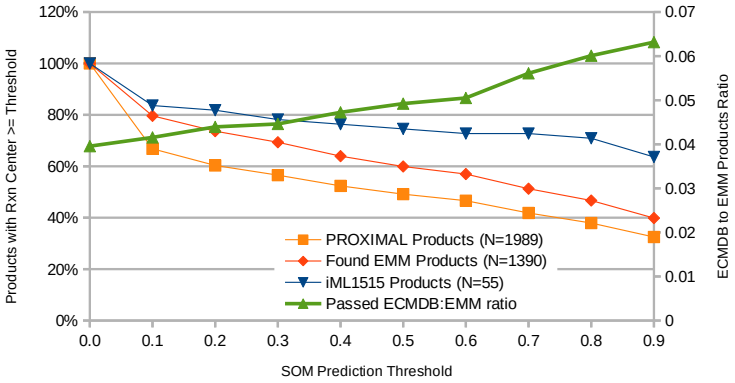

Figure 4.5 Percentage of products passed by the SOM predictor filter for different sets of metabolites, at different thresholds.

products – confirmed and overall – pass the filter. Importantly, the proportion of confirmed products grows with the threshold. The ratio of verifiable products steadily increases as the threshold is raised, indicating that metabolites confirmed to exist in *E. coli* are removed at a lower rate than unconfirmed metabolites; the ratio is at its lowest prior to the application of the filter (threshold = 0.0). Therefore, *GNN-SOM* enables more efficient prediction of metabolites suitable for creating EMMs. While we applied *GNN-SOM* post-promiscuous product generation as a screening tool, *GNN-SOM* can also be used to identify likely SOMs where the rules can be applied.

Ranking synthesis pathways based on SOM likelihood

SOM predictors can guide synthesis pathway construction. In this experiment, we compare known engineered two- and three-step pathways for 3-hydroxypropionic acid (3-HP) synthesis in *E. coli* to a number of putative synthesis pathways generated by PROXIMAL. To compare pathways based on *GNN-SOM* guidance, we calculate a pathway likelihood by taking the product of the model's predictions for the constituent reactions of each pathway. The output of *GNN-SOM* is a continuous value, representing a given site's likelihood of being an SOM for a certain EC number. Since the site and the type of biotransformation is specified,

such likelihood is reflective of the probability of occurrence for the entire reaction. The overall likelihood of a pathway is calculated as the product of the probabilities of its reaction steps. Pathway likelihoods thus provide a relative metric that allows comparing pathways based on SOM likelihood. As pathway construction (without filtering, e.g. based on yield or other metrics) generates a large number of putative pathways, we expect engineered pathways to have, on average, a higher pathway likelihood than the putative pathways.

We considered two engineered pathways for 3-HP synthesis known from the literature. The first pathway is catalyzed by a non-native enzyme, malonyl-CoA reductase, to produce 3-HP in two reaction steps [139, 140]. The second pathway relies on heterologous expression of several genes – *panD*, *gabT*, and *ydfG* – to yield 3-HP in three reaction steps starting from L-aspartate [141]. Both pathways have been successfully used as a part of longer pathways for sourcing 3-HP and derivatives from glucose consumed by *E. coli*. In many reactions comprising those pathways, there were multiple options for the selection of a reaction center. To identify the most plausible set of centers, we calculated *GNN-SOM* likelihoods for all possible combinations and selected the one that maximized the likelihood of the pathway.

The family of generated pathways was constructed via prediction of promiscuous activity involving 3-HP and enzymes natively present in *E. coli* – and since the bacterium does not natively produce this molecule, such interactions are representative of unsuccessful pathway synthesis outcomes. Working back from the target metabolite, PROXIMAL was used to propose precursor metabolites and the necessary reaction steps (enzyme as well as the reaction center) that could ultimately yield 3-HP. Because the number of such pathways can be extremely large, 50,000 pathways of each depth were sampled at random, followed by removing duplicates. This process produced 7,869 two-step and 45,622 three-step synthetic pathways. This set was further refined by removing pathways with intermediates not listed in the PubChem – such intermediates may less likely occur in nature than metabolites previously observed and catalogued in the database. In the end, we obtained 1,226 two-step and 1,118 three-step PubChem-only pathways.

The average likelihood of putative pathways was 0.24 for two-step and 0.17 for three-step interactions. For PubChem-only synthetic pathways, the mean likelihoods were 0.30 and 0.37 for two- and three-step pathways, respectively – the higher likelihood observed in this case suggests the model assigns greater confidence to interactions with evidence of the metabolite's existence. Finally, the likelihoods of the two effective synthesis pathways for the most plausible sets of reaction centers were found to be 0.987 and 0.956 in the cases of malonyl-CoA and L-aspartate pathways, respectively.

The calculated likelihoods allow ranking synthetic pathways based on the likelihood of the SOMs they depend on, and ideally, such ordering would prioritize functional pathways over the putative ones. The quality of this ranking can be quantified using the AUROC metric. The likelihood of the two-step malonyl-CoA pathway exceeded those of all considered two-step putative pathways, leading to the AUROC of 1.0. For the L-aspartate pathway and its putative three-step counterparts, the AUROC was 0.997. Therefore, our SOM predictor allows unlikely candidates to be deprioritized or removed from consideration at a low computational expense.

4.3 Conclusion

Here, we explored GNN-based models that predict atomic and bond SOMs for enzymatic reactions. Our analysis revealed that the bond-SOM prediction problem is more difficult than the atom-SOM prediction problem. Our GNN model, *GNN-SOM*, based on the Chebyshev convolutional operator consistently outperforms baseline ML classification models. Importantly, we showed that training on all enzymatic reactions outperforms the same model when trained on only CYP enzymes. Thus, the SOM prediction task (for CYP and non-CYP enzymes) benefits from a larger and more diverse training dataset. We also showed that the use of *GNN-SOM* can provide ranking on promiscuous products when evaluating the construction of EMMs and synthesis pathways for 3-HP.

Chapter 5 - Fine-grained structural classification of biosynthetic gene cluster products

Generalist models trained on large datasets can provide useful representations for many problems in biology, allowing effective classifiers to be learned using very small datasets. In this chapter, we present *BGCat*, a deep learning model for BGC product classification backed by a protein language model trained on millions of sequences. Our approach advances the state of the art in two key areas: by offering effective product classification in its traditional coarse-grained form, as well as highly detailed classification that was not previously available. We evaluate our method against several baselines and introduce a data augmentation strategy to further bolster the model's performance. Additionally, we use *BGCat* to offer new biological insights by providing new product labels for tens of thousands BGCs in antiSMASH DB and introducing the concept of GCF product class profiles.

5.1 Methods

5.1.1 Dataset construction

Our dataset is derived from MIBiG, the largest manually curated source of experimentally validated BGCs along with their products. The database is supported by an online community of 250+ members and currently includes 3,013 clusters [142], 96% of which are associated with bacteria and fungi. To use the most reliable BGC annotations, we exclude 377 of

those clusters that are marked as retired, resulting in a set of 2,636 MIBiG BGCs. Next, we leverage NPClassifier [67] to obtain detailed product type labels for these BGCs. The nomenclature used by NPClassifier is designed to be informative for natural products and features a hierarchy with three distinct levels: 7 pathways, 70 superclasses, and 672 classes; however, not all of them are covered by the BGC products. The product structures are processed as SMILES strings, which we canonicalize and de-duplicate using RDKit [106] to ensure each unique product is only classified once. When a BGC is known to produce multiple products, their labels are combined in a multi-label fashion. This process results in a dataset associating BGCs and their corresponding product classifications. In the set of MIBiG BGCs, 521 clusters lacked machine-readable product structures (e.g. a valid SMILES string was not specified) and an additional 69 BGCs produced no NPClassifier predictions; both of these BGC types were excluded from further consideration. This process yielded 2,046 BGCs, featuring 3,479 product structures, together resulting in 3,839 unique BGC-product pairs that formed our core MIBiG dataset. The most common compound categories at the pathway level were Polyketides and Amino acids & Peptides, with 77% and 65% of the BGCs featuring at least one product of the respective type. On average, each product was associated with 1.15 unique pathway labels.

5.1.2 Model architecture and training

We structured our model as a feed-forward deep neural network. The network operates on BGC gene embeddings and predicts product classification in a multi-label fashion. We implemented three separate networks for each of the pathway, superclass, and class label types, as well as a combined network that predicts all three types simultaneously. The BGC embeddings are constructed by applying the ESM Cambrian 600M [7] model to all biosynthetic (core and additional) genes; the remaining gene types (regulatory, transport-related, and other) were excluded due to their incidental effect on the product structure. Mean pooling was then used to combine gene-level embeddings into whole-BGC embeddings.

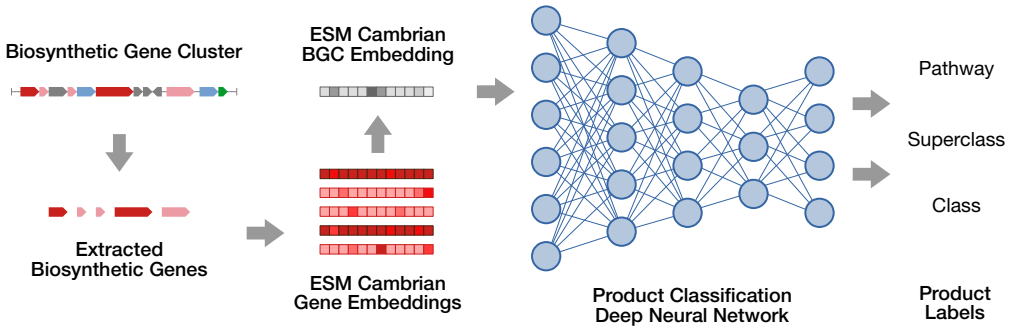

Figure 5.1 An overview of the *BGCat* model. A BGC identified by antiSMASH or a similar tool is accepted as the input. Next biosynthetic genes, shown as shades of red, are extracted and embedded using ESM Cambrian. Regulatory, transport-related, and other genes, represented by green, blue, and gray arrows, respectively, are ignored. The gene-level embeddings are then aggregated to yield an embedding representative of the BGC. A deep neural network is then used to predict pathway, superclass, and class labels of the potential products.

The input dimension of the network is 1,152 to match the size of the BGC embeddings, the hidden layers progressively reduce in their dimensionality, and the final layer predicts the target labels. The model consists of three hidden layers implemented using PyTorch and is trained using the Adam optimizer [121] with binary cross-entropy as the loss function. The architecture of the model is illustrated in Figure 5.1.

5.1.3 Dataset augmentation

The core MIBiG dataset was augmented with BGCs sourced from BGC Atlas [143], an online database containing nearly 1.8 million clusters, derived from 35k bacterial metagenomes found in MGnify [144]. These BGCs were identified by antiSMASH and as such include both complete and incomplete clusters. Incomplete BGCs pose a problem in this instance as they may be insufficient to support product classification; we therefore focused on the subset of 204,661 complete clusters. In addition, the BGC Atlas clusters provide limited details about their product molecules, so we sought to leverage the product information available in similar BGCs in MIBiG. Following the method described in BGC Atlas, we grouped the complete BGC Atlas clusters into GCFs using BiG-SLiCE 2.0.0 [51] with a

distance threshold of 0.4, resulting in a set of 18,596 GCFs. Next, we queried our core MIBiG BGCs against those GCFs, finding at least one similar MIBiG BGC for 743 GCFs from the set. In the event BiG-SLiCE identified multiple candidate GCFs for a given BGC, only the highest-ranking assignment was used. We assume that all BGCs within a given GCF will yield substantially similar products; therefore, product classification labels from the MIBiG BGCs were copied to all BGC Atlas-derived BGCs that are part of the same GCF. This process resulted in a set of 9,055 BGCs sourced from BGC Atlas with their corresponding product labels. These BGCs, combined with our core MIBiG set, resulted in 11,101 BGC-label pairs which we refer to as our augmented set.

5.2 Results

5.2.1 Method outperforms existing techniques in traditional BGC product classification

Currently there is no comparable approach that seeks to classify BGC products based on a detailed nomenclature appropriate for natural products. Most of existing work in this space treats BGC product classification as an incidental problem to BGC detection, classifying products into the 7 coarse-grained classes provided in MIBiG [142]. Although this nomenclature provides somewhat limited information about the product, it offers a benchmark where we can compare against current baseline techniques. We considered four methods as our baselines: the built-in classification functionality of antiSMASH [49], DeepBGC [45], BiGCARP [46], and BGCCGB [47]. We evaluated the model on two versions of the MIBiG database: version 1.4 to allow for direct comparison with previous work, and version 4.0 to use the most up-to-date information. In both cases, the evaluation was performed using 5-fold cross-validation.

As shown in Table 5.1, our approach provides substantial performance gains over the baselines in traditional BGC product classification. In a head-to-head comparison using the MIBiG v. 1.4 database, our method improved upon AUROC and precision relative to prior approaches, although the recall was lower than that of BGCCGB. However, this gap in recall

Table 5.1 Average AUROC, recall, and precision for traditional BGC product classification; table adapted from BGCCGB[47].

Metric	antiSMASH	DeepBGC	BiGCARP	BGCCGB	Our method with MIBiG 1.4	Our method with MIBiG 4.0
AUROC	0.784	0.792	0.855	0.862	0.936	**0.937**
Recall	0.718	0.691	0.539	0.740	0.693	**0.768**
Precision	0.658	0.741	0.593	0.732	0.811	**0.825**

disappeared when the model was evaluated on MIBiG v. 4.0. These trends help illustrate two key areas where our method has improved over existing techniques. First, the use of ESM Cambrian embeddings of gene sequences alongside a deep neural network resulted in substantial gains over all but one baseline methods. Second, the introduction of additional BGC information available in a more recent release in MIBiG allowed our model to better learn the product classification task and surpass all baselines.

5.2.2 *BGCat* is effective for BGC natural product class prediction

Next, we apply our method for detailed natural product class prediction. In this problem, we consider the three NPClassifier classification levels with increasing level of specificity: pathway, superclass, and class [67]. The pathway level consists of seven classes: Alkaloids, Amino acids and Peptides, Carbohydrates, Fatty acids, Polyketides, Shikimates and Phenylpropanoids, and Terpenoids. As such, pathway-level classification is similar to MIBiG class prediction seen in traditional BGC product classification, although the mapping between the two is not one-to-one. Meanwhile, the set of possible superclasses and classes is substantially larger, with an almost order of magnitude increase at each level. We consider two types of models: one that predicts a specific level of classification, and another that predicts all levels simultaneously.

To provide a comprehensive evaluation of the method, we consider four metrics: average precision (AP), area under the receiver operating characteristic (AUROC), recall, and precision. The latter two provide an approximate measure of performance assuming a fixed

likelihood threshold of 0.5, which is a customary default choice [145] for binary classifiers (e.g. whether or not a BGC makes a product of a particular class). However, such a cut-off is not necessarily optimal and its selection is subject to trade-offs [145, 146]. Therefore, we also consider AUROC, a traditional classifier performance metric, as well as average precision, a measure of the quality of relative ranking. A key characteristic of our BGC dataset is class imbalance, with some types of BGC products being vastly overrepresented compared to others. We therefore employ two averaging schemes for our metrics. Micro averaging treats each prediction in a standalone fashion, representing the performance of a typical prediction. Macro averaging computes the metric for each class separately and reports the average across the classes. With all classes being treated equally irrespective of any imbalance, macro averaging represents the performance of the method for a randomly selected class. When applied to average precision, this macro-averaged value corresponds to the mean average precision (mAP) commonly used in multi-label classification.

The results calculated with 5-fold cross-validation on the MIBiG 4.0 set can be found in Table 5.2a. The method achieves high AUROC values across the board; however, other measures are more varied. In general, performance reduces with increasingly more specific classification levels. The prediction of all levels simultaneously presents intermediate performance. Recall values are considerably lower with class-level predictions, suggesting that many classes cannot be identified, likely as a consequence of class imbalance. This is further supported by the significantly lower macro averaging results, where the underrepresented classes have greater impact on the metric. However, precision remains high, indicating that the model is relatively accurate in the predictions it makes.

To showcase the performance of the method as additional BGC information becomes available, we introduce a temporal data split. In this instance, we retained BGCs in a previous version of MIBiG (v. 3.1) for training the model, and reserved the BGCs added in the next version (v. 4.0) for testing the performance. The results can be found in Table 5.2b. In this case, the performance of the method is broadly similar to that with random

Table 5.2 NP label prediction performance, for all labels being predicted at the same time and pathway, superclass, and class labels being predicted separately. a) Results for a random 5-fold cross-validation split. b) Results for a temporal split on MIBiG, with updates between versions 3.1 and 4.0 used as the test set.

Label type	Micro averaging				Macro averaging			
	AP	AUROC	Recall	Precision	AP	AUROC	Recall	Precision
a) NP prediction with MIBiG 4.0								
All labels	0.606	0.930	0.397	0.836	0.302	0.810	0.040	0.815
Pathway	0.854	0.939	0.771	0.829	0.741	0.910	0.642	0.784
Superclass	0.673	0.929	0.467	0.866	0.511	0.852	0.182	0.881
Class	0.362	0.876	0.160	0.781	0.300	0.793	0.016	0.807
b) Temporal split on MIBiG								
All labels	0.655	0.923	0.444	0.874	0.324	0.780	0.071	0.759
Pathway	0.898	0.956	0.811	0.852	0.777	0.920	0.654	0.882
Superclass	0.674	0.905	0.479	0.849	0.460	0.757	0.195	0.834
Class	0.423	0.863	0.256	0.795	0.310	0.778	0.031	0.810

cross-validated split; however, some metrics are varied likely due to the smaller test set size: the test size in the cross-validated split was 768, while in the temporal split it was 183. Nevertheless, this indicates the method remains effective on newly added BGCs.

5.2.3 Dataset augmentation enhances BGC NP class predictions

The core MIBiG dataset is restrictive for machine learning methods that require large amounts of data. As such, we sought to increase the number of training examples by introducing a data augmentation scheme that has expanded the dataset size 5-fold. The effect of that change on prediction performance can be seen in Table 5.3a. The evaluation was performed on the same temporal split as in Table 5.2b, and the BGCs reserved for testing were excluded from the augmentation scheme. We note a consistent increase in all metrics except for precision. This indicates the model was able to recall more of the classes with enhanced quality of ranking, at the expense of slightly lower accuracy of its predictions.

Next, we validated whether the BGCs introduced by augmentation were informative by introducing an augmentation split, where the clusters sourced from BGC Atlas were used for

Table 5.3 NP label prediction performance with dataset augmentation. a) Results for a temporal split on MIBiG, with updates between versions 3.1 and 4.0 used as the test set. b) Results for the augmentation split, where the augmented examples were used for training and non-augmented ones for testing. c) Results for the realistic data split, with larger GCFs used for training and smaller unseen GCFs for testing.

Label type	Micro averaging				Macro averaging			
	AP	AUROC	Recall	Precision	AP	AUROC	Recall	Precision
a) Temporal split								
All labels	0.669	0.937	0.554	0.784	0.426	0.854	0.235	0.750
Pathway	0.889	0.962	0.837	0.809	0.760	0.936	0.689	0.783
Superclass	0.637	0.924	0.521	0.731	0.436	0.856	0.310	0.601
Class	0.495	0.896	0.413	0.735	0.361	0.835	0.196	0.796
b) Augmentation split								
All labels	0.442	0.933	0.508	0.404	0.135	0.819	0.078	0.238
Pathway	0.596	0.871	0.742	0.481	0.453	0.824	0.599	0.408
Superclass	0.403	0.915	0.459	0.337	0.177	0.832	0.144	0.248
Class	0.201	0.894	0.270	0.312	0.124	0.810	0.054	0.300
c) Realistic split								
All labels	0.315	0.835	0.271	0.531	0.112	0.646	0.049	0.455
Pathway	0.527	0.790	0.505	0.530	0.403	0.734	0.367	0.443
Superclass	0.304	0.807	0.242	0.562	0.129	0.691	0.067	0.386
Class	0.151	0.772	0.127	0.448	0.094	0.628	0.032	0.452

training the model and the remaining ground-truth MIBiG BGCs were reserved for testing. The results can be found in Table 5.3b. With lack of access to high-quality MIBiG BGCs, the performance of the method has significantly decreased; however, it remained effective to a degree, suggesting that the added BGCs provide valuable information for the product classification task.

5.2.4 Model generalizes to unseen GCFs

With the landscape of known and well-annotated BGCs being limited, it is highly desirable for the model to be applicable to novel BGCs with no previously described function. To estimate the performance of the method under those circumstances, we introduce a "realistically novel" data split similar to one described in Profile-QSAR [147]. This split seeks

to maximize the difference between training and test sets by separating BGCs according to their GCF membership. The training set is assembled from the BGCs comprising the largest GCFs, while the smallest GCFs are used to build the test set. This reflects a degree of realism because the largest GCFs are likely to contain more common BGCs with a greater likelihood of appearing in our dataset. Conversely, the likely less common BGCs in the smallest GCF are reflective of novel BGCs that we might not encounter in the course of training our model.

The results of this experiment can be found in Table 5.3c. Compared to the temporal split in Table 5.3a, the performance is considerably lower; however, the model retains a degree of effectiveness on the unseen GCFs, demonstrating its generalizability.

5.2.5 BGC product classification offers a new perspective on GCFs

Detailed classification of BGC natural products enables more comprehensive understanding of GCFs. Existing methods for constructing GCFs primarily leverage genomic sequence information for grouping similar BGCs, and although they do not directly consider the natural products, the resulting families are often correlated with specific product structures [52, 53]. However, it remains uncertain whether this pattern holds universally for all types of products and BGCs. As such, there is value in considering the natural products more explicitly.

To offer greater insight into GCFs, we propose the concept of product class profiles (PCPs), whereby each GCF is associated with a probability distribution of natural product types. We construct these distributions accordingly, by applying our model to these BGCs and aggregating the predicted types for each GCF. Thus, a given GCF may be associated with specific classes of molecules to varying degrees. Of particular interest are the GCFs with a distinct PCP, whose product type distribution is meaningfully different from the overall distribution of the dataset (the background), as such GCFs are likely to be strongly associated with particular types. We identify these GCFs using Pearson's chi-squared test

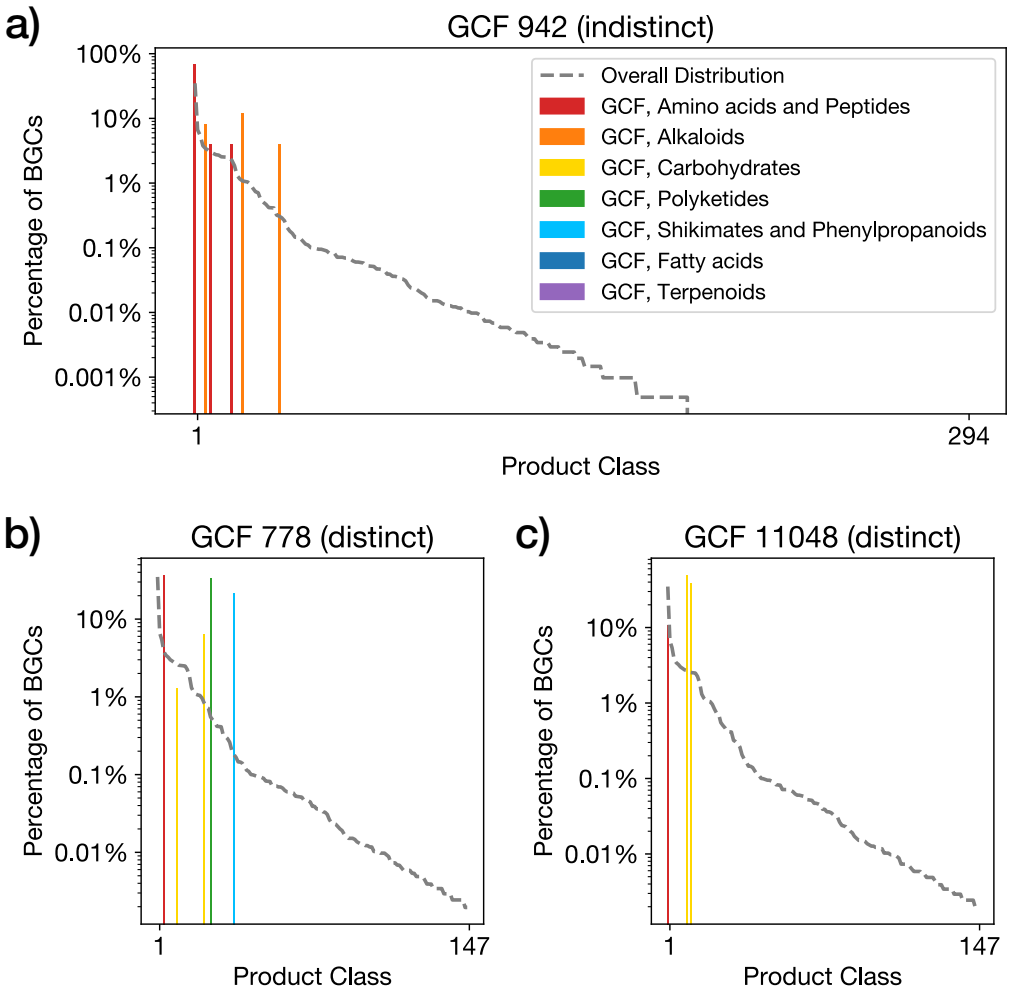

Figure 5.2 Example product class profiles for different GCFs constructed using *BGCat*. (a) A GCF with an indistinct distribution of product types: amino acids and alkaloids are represented similarly to the overall dataset. (b) An example of a GCF with a distinct distribution of product classes featuring a multitude of product types. (c) A GCF with a distinct distribution specialized in specific product classes. In both distinct cases, the proportion of BGCs responsible for a given product type is well above the average for the dataset (dashed gray line).

with a significance threshold of $p < 0.05$. This process yielded 5,083 (27%) distinct GCFs out of the 18,596 derived from BGC Atlas.

Select examples of such PCPs are shown in Figure 5.2. GCF 942 (Figure 5.2a) is representative of an indistinct GCF, whose product type distribution is not significantly different from the background. The two most common pathways, Amino acids & Peptides

and Alkaloids are represented in this family in a similar proportion to the overall dataset. In contrast, GCF 778 (Figure 5.2b) has a distinct distribution with a rich variety of product types, with classes from the Carbohydrates, Polyketides, and Shikimates & Phenylpropanoids pathways notably overrepresented. GCF 11048 (Figure 5.2c) is also an example of a distinct GCF, albeit a more specialized one for production of carbohydrates.

5.2.6 *BGCat* predicts detailed product labels for microbial BGCs in antiSMASH DB

To showcase a practical application of our method, we apply it to antiSMASH DB [68], a public repository of over 260k BGCs from over 36k bacterial genomes precomputed using antiSMASH. Although antiSMASH is able to elucidate the approximate scaffold of some products, in most cases it is unable to predict the full structure or detailed classification of the product. One way to gather deeper context about the product is to leverage the KnownClusterBlast (KCB) module of antiSMASH that identifies similar clusters in MIBiG and thus can provide structures of related natural products. However, in some cases, the MIBiG BGCs may not include the product structure, or a KCB match may not be available for a particular antiSMASH BGC.

We can therefore separate antiSMASH DB into three subsets. Subset 1 consists of 114k BGCs with both a KCB match and one or more product structures available in the corresponding MIBiG BGC. Subset 2 contains 27k BGCs that have a KCB match but no machine readable structure provided in MIBiG. Subset 3 includes 121k BGCs that feature no KCB matches at all. The first two subsets offer an opportunity to evaluate *BGCat*, since the MIBiG BGCs identified by KCB will contain a coarse classification of the product at minimum; subset 1 additionally provides product structures, which can be used to compare NPClassifier's performance to the same standard. The final subset offers opportunity for new discovery using our prediction approach. An overview of the three subsets is shown in Figure 5.3.

To evaluate our model, we implement a mapping from NPClassifier terms to MIBiG

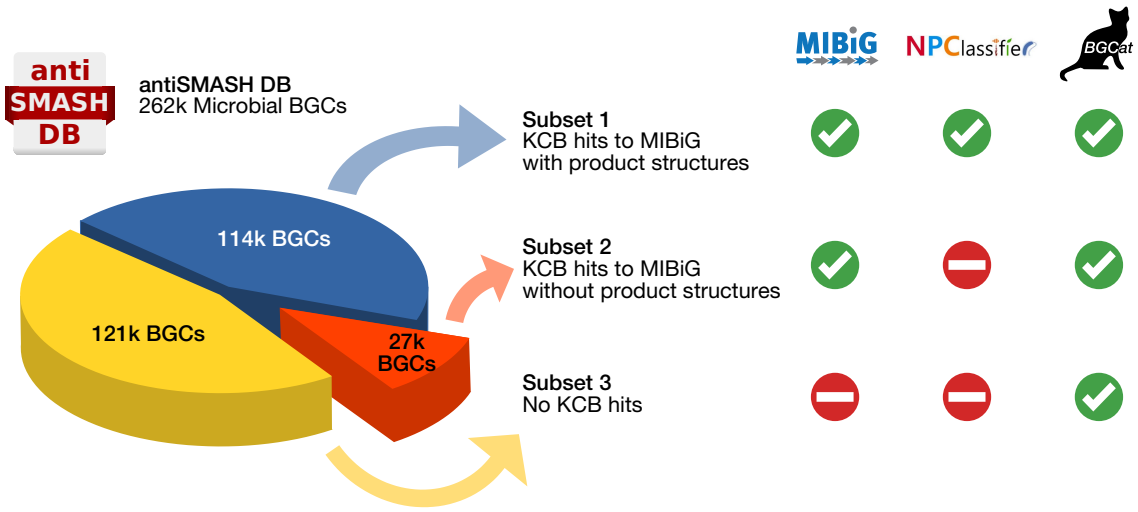

Figure 5.3 An overview of the three antiSMASH DB subsets. BGCs from subset 1 feature KCB hits to MIBiG BGCs with product structures, allowing the use of MIBiG labels and NPClassifier predictions for BGC characterization. Subset 2 also has hits to MIBiG BGCs, albeit without machine-readable product structures, precluding the use of NPClassifier. Finally, subset 3 features no KCB hits at all, which eliminates the availability of MIBiG labels as well. However, *BGCat* remain available for all three subsets.

coarse human-annotated labels and consider the percentage of BGCs for which at least one of *BGCat* predictions is in agreement with the MIBiG classification. This measure effectively represents our model's ability to recall information from MIBiG BGC labels. These labels, being annotated by human experts, can be considered highly accurate; however, due to individual preferences or unknown BGC products these labels may not necessarily be exhaustive. Most categories map trivially; however, Shikimates and Phenylpropanoids had no direct equivalent and had to be mapped to the MIBiG "Other" category. Fatty acids were mapped to Polyketides in MIBiG due to the close structural and evolutionary similarities between their corresponding synthases [148]. RiPPs and NRPs are difficult to distinguish based on the peptide structure alone, so they were grouped into one surrogate MIBiG label "RiPP/NRP." The objective of the model is therefore to predict a detailed product class that maps to the same label found in a MIBiG KCB match.

On subset 1, the top prediction of the model is able to recover a correct MIBiG label for 76% of the BGCs; the percentage increases to 84% when top two predictions are considered.

NPClassifier on average also predicts two labels per BGC, and its rate of agreement with MIBiG is similarly at 85%. Because our dataset is based on the product classification from NPClassifier, this result represents the upper bound on the *BGCat*'s accuracy and its ability to nearly match it is encouraging. On subset 2, the machine readable structure of the MIBiG product is not available, which precludes the use of NPClassifier. However, our model is still able to recover at least one MIBiG label for 54% of BGCs with its top prediction, and for 67% of BGCs with its top two predictions. The lower performance is expected in this case because the lack of structure in a MIBiG BGC likely means the structure is not part of the MIBiG database, which would prevent our model from observing it in the training set.

Finally, we consider subset 3, for which no KCB data is available. In this case, *BGCat* can provide unique insights about the products of these BGCs. To evaluate utility of these new results relative to the available GCF groupings, we calculated the Normalized Mutual Information (NMI) between the GCF assignments and the labels. NMI is an established metric for evaluating the agreement between two clusterings based on information theory. For pathway-, superclass-, and class-level labels, the NMI was found to be 0.16, 0.27, and 0.38, respectively. These low values indicate that product labels and GCF membership share relatively little mutual information and therefore capture largely orthogonal aspects of BGC biology: GCFs reflect sequence similarity, whereas *BGCat*'s predictions reflect chemical outcomes. The increasing NMI from pathway to class level further suggests that finer-grained product labels align more closely, but still only modestly, with genomic similarity, whereas coarse labels obscure these relationships. Thus, *BGCat* provides substantial, nonredundant information beyond what GCF groupings alone can offer.

Figure 5.4 illustrates the breadth and diversity of product types predicted across antiSMASH DB subset 3. At the pathway level (Figure 5.4a), all seven natural product pathways are well represented, with Amino acids & Peptides and Polyketides dominating both BGC and GCF counts. This broad coverage suggests that *BGCat* predictions span the full metabolic landscape of microbial natural products. In contrast, the class-level

distributions (Figure 5.4b) exhibit a characteristic long-tail pattern: a handful of classes contain thousands of BGCs, whereas many others are sparsely populated. This imbalance underscores both the chemical richness of BGC-encoded metabolites and the difficulty of fine-grained classification. Notably, GCF counts broadly track BGC frequencies, indicating that highly populated classes correspond to genuinely diverse biosynthetic families rather than duplicated clusters. In other words, classes with many BGCs also contain many distinct genomic architectures, showing that their abundance reflects true biosynthetic diversity rather than repeated instances of the same cluster. Together, these figures validate the utility of fine-grained structural classification and highlight the data challenges inherent to learning at this level of resolution.

5.3 Conclusion

BGCat is a deep neural network for fine-grained BGC product classification. It builds upon the NPClassifier [67] nomenclature for natural products, making it particularly suitable for describing metabolites arising from BGCs. By leveraging a large pretrained protein language model, our method takes advantage of the vast repository of available protein sequence information to create biologically meaningful representations of biosynthetic genes. In addition, our novel data augmentation strategy allows the neural network to learn from a greater variety of BGCs than would otherwise be possible with presently existing datasets. We show that our method is effective both for traditional coarse-grained as well as fine-grained BGC product classification. The fine-grained approach is especially interesting in the context of BGC products as there is currently no method capable of predicting the complete structure of such molecules in the general case; thus, detailed product classification offers the greatest insight into the function of BGCs currently available. Furthermore, we anticipate that *BGCat* can improve paired omics analyses by improved linking and ranking of predicted BGCs in genomes to mass spectra with predicted compound classes [149].

In contrast to prior algorithmic methods, our machine learning-based approach aims to

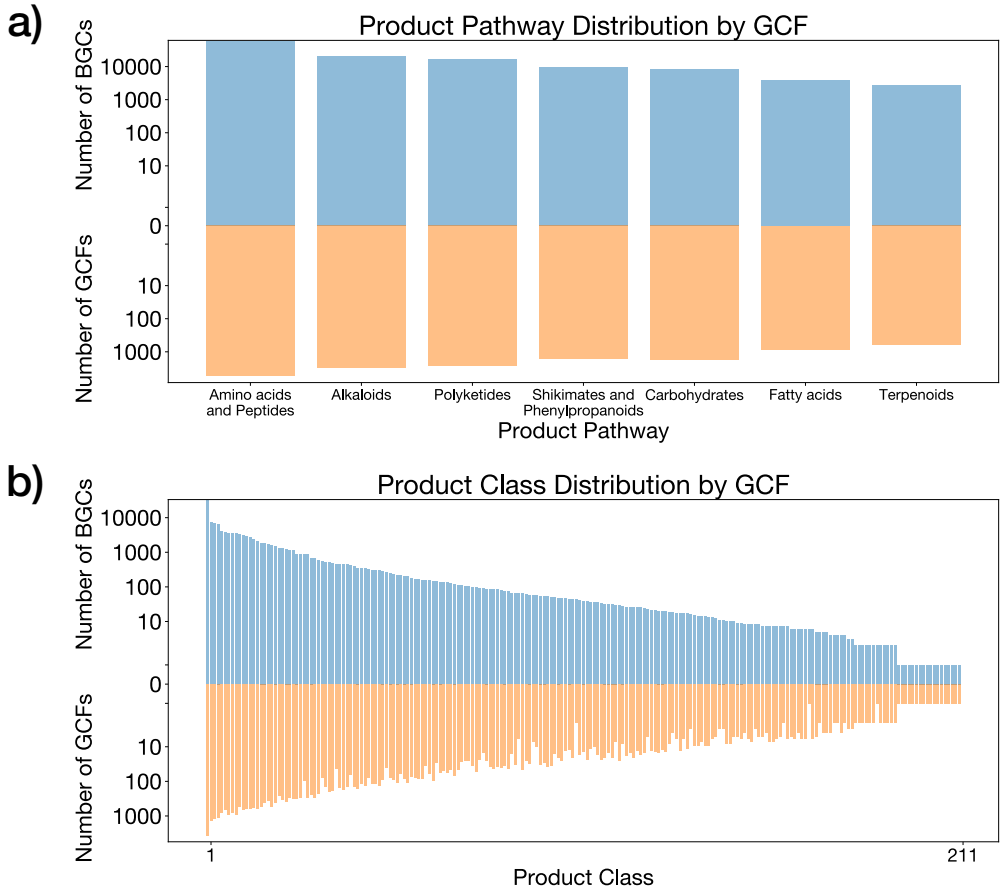

Figure 5.4 Distributions of BGCs and GCFs in the antiSMASH DB subset 3, stratified by: (a) the 7 pathway labels, and (b) product class labels. The x-axis represents different product labels predicted by the model. The y-axis represents the number of BGCs or GCFs which contain a given product class. The BGCs are shown as blue bars in the upper half of each plot, while the GCFs are shown as orange bars in the bottom half. These distributions highlight the broad biochemical coverage of subset 3 and reveal substantial fine-grained diversity, underscoring the value of *BGCat*'s product-based predictions for characterizing BGCs beyond sequence-based groupings.

be generalizable to different types of clusters and products. We have shown that the model is effective for predicting product classes for new BGCs introduced over time and maintains a degree of accuracy on novel types of BGCs when grouped by GCFs. As new information about BGCs becomes available, our model can be trained on new data to further enhance its accuracy and coverage of product types. It should be noted, however, that our model may need further adaptations to apply beyond microbial BGCs, as BGCs found in other

organisms often require special considerations [150, 151, 152].

To highlight a practical application of our method, we have used it to provide product labels for a diverse set of microbial BGCs deposited in antiSMASH DB. We have shown that our model provides product class labels in agreement with MIBiG labels for at least 54% of BGCs with KCB matches. However, of particular interest was the subset of 121k BGCs with no KCB matches: in this case, our model provided the only means of predicting the BGC product type.

Chapter 6 - Conclusion

Applications of computational methods to biological problems are often met with severe data constraints that must be addressed one way or another on a case-by-case basis. In this thesis, we explored three such problems that required the use of customized data-efficient strategies. First, we developed a co-design approach for protein-ligand pairs intended for a novel bio-electronic sensing application. We alleviated the knowledge limitations about the particular strain of Ndh2 through careful modeling of the protein-quinone interaction, anticipating the effects of various design choices and limiting the design space to a computationally tractable region. Next, we introduced *GNN-SOM*, a technique for SOM prediction hallmarked by its GNN-based model architecture. We overcame the limitations of the relatively small enzymatic reaction dataset by the use of a well-suited model design as well as SOM-specific data preprocessing steps. Finally, we proposed *BGCat*, a method for detailed BGC product classification. The constraints of an even smaller BGC classification dataset were addressed by leveraging a pretrained embedding model and introducing a data augmentation strategy. These biological problems showcase two distinct angles for improving data efficiency: adapting the computational method itself and enhancing the dataset it is backed by. Despite the common themes, however, these problems also emphasize the necessity of domain-specific customizations and the diversity of possible approaches. Overall, this thesis demonstrates a persistent need for data efficiency and tailor-made strategies to achieve it when developing solutions for biological problems.

6.1 Research summary

This thesis provides several key advancements. The first problem we considered is protein-ligand co-design, aiming to enhance binding affinity and, ultimately, Ndh2-quinone-mediated EET. This approach stands apart from the traditional paradigm of individual design that modifies only one aspect of the pair and leads to improved design outcomes. To address the lack of structural information about the specific Ndh2 strain, we developed and computationally validated a homology model of the protein, which enabled us to model the Ndh2-quinone binding and explore the dual protein-ligand landscape in an efficient manner. The results from this exploration further allowed us to reason about residue-level interactions and mutations with potentially outsized impact on affinity, advancing the scientific understanding of the phenomenon more generally.

The second problem is SOM prediction in enzymatic reactions acting on small molecules. To tackle this problem, we developed *GNN-SOM*, a GNN-based model for classifying atoms or bonds as SOMs given an EC number. By taking advantage of the message passing framework of GNNs, our method is able to learn informative representations through local and global contexts within molecular graphs, all while relying on a comparatively small dataset. The method is more accurate than traditional baseline approaches for SOM prediction and does not require specialized models to recognize CYP and non-CYP SOMs. In addition, *GNN-SOM* was shown to possess broader applications in metabolic engineering scenarios, such as building EMMs and ranking synthesis pathways.

The final problem is detailed BGC product classification. For this task, we proposed *BGCat*, a deep neural network capable of predicting the NPClassifier labeling of BGC products. In contrast to existing methods, our approach provides classification using a detailed nomenclature intended for natural products. By leveraging BGC representations from a pretrained protein language model, our method learns accurate classifiers given a restrictive core dataset of only 2,046 BGCs, surpassing the performance of other methods;

the introduction of augmented dataset further bolsters *BGCat*'s performance. Beyond classifying BGC products, the method offers novel insights into the functions of GCFs through PCPs.

6.2 Future directions

This thesis explored computational approaches to three biological problems with significant data constraints. The proposed methods address the issue of data scarcity in a variety of ways, including through the use of specific models and enhancements of datasets. The resulting approaches demonstrate a consistent performance advantage over the existing techniques and utility in specific biological applications. Future work includes a number of new directions that may further enhance these results and broaden their impact.

Our approach to protein-ligand co-design demonstrates the benefits of the simultaneous design paradigm. One promising direction involves expanding the design space by allowing multiple simultaneous mutations similarly to Combinatorial Saturation Mutagenesis [153], introducing replacements with non-canonical amino acids [154], and increasing the number of modifications on the quinone. Another direction is leveraging generative ML models to provide more favorable starting points for design [155, 156, 157, 158] or objective-guided exploration [159, 160]. Further improvements may be realized by structure refinement with the help of ML [78, 79, 80], MD [161], or insights into specific mechanisms of action related to the active site [162, 163]. The evaluation objective can also be enhanced by considering the need to *release* the product of the Ndh2-quinone interaction (a semiquinone), or other factors pertinent to EET such as redox potential [164] and cLogP [18]. These changes would allow more thorough exploration of the design space with more accurate means of identifying pairs of interest.

GNN-SOM learns informative molecular representations by leveraging the GNN message passing framework. One possible improvement is incorporating other reaction datasets in addition to KEGG RPAIR: for example, the RetroRules database [35] can provide several

thousands of additional biochemical reactions spanning over 5,000 EC numbers [165]. Introducing these additional reactions would yield more generalizable models and potentially permit effective learning of enzyme classification beyond the top two levels. Another avenue for improvement is introducing the model to broader chemistry knowledge via pretraining on datasets from adjacent domains. This strategy has already been shown useful for a number of molecule-related tasks [166, 167, 168, 169] and the benefits of such learned representations may extend to SOM prediction as well. Separately, the addition of phyisochemical molecular descriptors, though not strictly required, may further aid in SOM prediction by providing context not otherwise implied by the 2D molecular structure. Overall, these improvements are likely to improve robustness and performance of the method.

Our approach to BGC product classification takes advantage of a pretrained model in conjunction with data augmentation to learn detailed classifiers with a restrictive data set. The ESM Cambrian model used for BGC representation was originally trained on protein sequences; thus, one possible future direction is fine tuning it on genomic datasets to improve the applicability of its embeddings to BGCs. The current BGC data landscape lends itself to a multi-stage fine tuning strategy, where large-scale datasets like BGC Atlas [143] and antiSMASH DB [68] could be used for the initial alignment to the genomic domain, followed by human-curated datasets such as MIBiG [142] for the precise tuning. Such an embedding model tailored to BGCs would benefit product classification as well as a slew of other BGC-focused tasks. Another possibility is enriching BGC representations with other types of embeddings. For example, Pfam domains [85] have been popular for BGC detection and classification [45, 47], demonstrating their expressive power for describing BGCs. Introducing embeddings of such domains alongside the gene-level embeddings may thus provide the classification network with useful context that eludes the ESM Cambrian embeddings. Lastly, we believe dataset advancements in the BGC space will lead to further performance gains, similarly to those observed with the upgrade from MIBiG v. 1.4 to v. 4.0. Creation of datasets specifically designed for product classification is an avenue

for expediting this process as the current curation efforts seek to provide either a complete product structure – arguably, a more ambitious task – or a coarse-grained description; there is presently no resource offering high-confidence detailed product labelings. Taken together, these improvements seek to enhance the model performance through improved BGC representations and expanding its access to genomic data.

References

[1] P. Hogeweg, "The roots of bioinformatics in theoretical biology," *PLOS Computational Biology*, vol. 7, no. 3, pp. 1–5, Mar. 2011.

[2] E. Mayr, *What Makes Biology Unique? Considerations on the Autonomy of a Scientific Discipline*. Cambridge University Press, 2004.

[3] D. M. Lowe, "Extraction of chemical structures and reactions from the literature," Ph.D. Dissertation, University of Cambridge, 2012.

[4] H. Dai, C. Li, C. W. Coley, B. Dai, and L. Song, *Retrosynthesis prediction with conditional graph logic network*, 2020. arXiv: 2001.01408 [cs.LG].

[5] M. J. Kusner, B. Paige, and J. M. Hernández-Lobato, "Grammar variational autoencoder," in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ser. ICML'17, Sydney, NSW, Australia: JMLR.org, 2017, pp. 1945–1954.

[6] T. Yamada, M. Hattori, M. A. Oh, S. Goto, and M. Kanehisa, "Rpair: A database of chemical transformation patterns in enzymatic reactions," in *Proceedings of GIW 2005: The Sixteenth International Conference on Genome Informatics*, Pacifico Yokohama, Japan, 2005.

[7] ESM Team, *Esm cambrian: Revealing the mysteries of proteins with unsupervised learning*, 2024.

[8] S.-F. Zhou and W.-Z. Zhong, "Drug design and discovery: Principles and applications," *Molecules*, vol. 22, no. 2, 2017.

[9] W. Gao and C. W. Coley, "The synthesizability of molecules proposed by generative models," *Journal of Chemical Information and Modeling*, vol. 60, no. 12, pp. 5714–5723, 2020, PMID: 32250616. eprint: https://doi.org/10.1021/acs.jcim.0c00174.

[10] M. M. Shanbhag, G. Manasa, R. J. Mascarenhas, K. Mondal, and N. P. Shetti, "Fundamentals of bio-electrochemical sensing," *Chemical Engineering Journal Advances*, vol. 16, p. 100 516, 2023.

[11] J. U. Bowie, S. Sherkhanov, T. P. Korman, M. A. Valliere, P. H. Opgenorth, and H. Liu, "Synthetic biochemistry: The bio-inspired cell-free approach to commodity chemical production," *Trends Biotechnol.*, vol. 38, no. 7, pp. 766–778, Jul. 2020.

[12] J. N. Blaza et al., "The mechanism of catalysis by type-II NADH:quinone oxidoreductases," *Scientific Reports*, vol. 7, no. 1, p. 40 165, Jan. 2017.

[13] B. C. Marreiros, F. V. Sena, F. M. Sousa, A. P. Batista, and M. M. Pereira, "Type II NADH:quinone oxidoreductase family: Phylogenetic distribution, structural diversity and evolutionary divergences," *Environmental Microbiology*, vol. 18, no. 12, pp. 4697–4709, 2016. eprint: https://ami-journals.onlinelibrary.wiley.com/doi/pdf/10.1111/1462-2920.13352.

[14] B. B. Seo, T. Kitajima-Ihara, E. K. L. Chan, I. E. Scheffler, A. Matsuno-Yagi, and T. Yagi, "Molecular remedy of complex I defects: Rotenone-insensitive internal NADH-quinone oxidoreductase of *Saccharomyces cerevisiae* mitochondria restores the NADH oxidase activity of complex I-deficient mammalian cells," *Proceedings of the National Academy of Sciences*, vol. 95, no. 16, pp. 9167–9171, 1998. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.95.16.9167.

[15] G. Peltier, E.-M. Aro, and T. Shikanai, "NDH-1 and NDH-2 plastoquinone reductases in oxygenic photosynthesis," *Annual Review of Plant Biology*, vol. 67, no. 1, pp. 55–80, 2016, PMID: 26735062. eprint: https://doi.org/10.1146/annurev-arplant-043014-114752.

[16] S. W. May, "Applications of oxidoreductases," *Current Opinion in Biotechnology*, vol. 10, no. 4, pp. 370–375, 1999.

[17] E. T. Stevens et al., "Lactiplantibacillus plantarum uses ecologically relevant, exogenous quinones for extracellular electron transfer," *bioRxiv*, 2023. eprint: https://www.biorxiv.org/content/early/2023/03/14/2023.03.13.532228.full.pdf.

[18] S. Li, C. De Groote Tavares, J. G. Tolar, and C. M. Ajo-Franklin, "Selective bioelectronic sensing of pharmacologically relevant quinones using extracellular electron transfer in lactiplantibacillus plantarum," *Biosensors and Bioelectronics*, vol. 243, p. 115 762, 2024.

[19] B. T. Blackburn et al., "Identifying key properties that drive redox mediator activity in lactiplantibacillus plantarum," *Angewandte Chemie International Edition*, vol. 64, no. 19, 2025. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/anie.202424867.

[20] A. R. Finkelmann, D. Goldmann, G. Schneider, and A. H. Göller, "Metscore: Site of metabolism prediction beyond cytochrome p450 enzymes," *ChemMedChem*, vol. 13, no. 21, pp. 2281–2289, 2018. eprint: https://chemistry-europe.onlinelibrary.wiley.com/doi/pdf/10.1002/cmdc.201800309.

[21] J. Kirchmair et al., "Fast metabolizer (fame): A rapid and accurate predictor of sites of metabolism in multiple species by endogenous enzymes," *Journal of Chemical Information and Modeling*, vol. 53, no. 11, pp. 2896–2907, Nov. 2013.

[22] J. D. Tyzack and J. Kirchmair, "Computational methods and tools to predict cytochrome p450 metabolism for drug discovery," *Chemical Biology & Drug Design*, vol. 93, no. 4, pp. 377–386, 2019. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cbdd.13445.

[23] N. L. Dang, M. K. Matlock, T. B. Hughes, and S. J. Swamidass, "The metabolic rainbow: Deep learning phase i metabolism in five colors," *Journal of Chemical Information and Modeling*, vol. 60, no. 3, pp. 1146–1164, Mar. 2020.

[24] S. He et al., "Site of metabolism prediction for oxidation reactions mediated by oxidoreductases based on chemical bond," *Bioinformatics*, vol. 33, no. 3, pp. 363–372, Sep. 2016. eprint: https://academic.oup.com/bioinformatics/article-pdf/33/3/363/49037731/bioinformatics_33_3_363.pdf.

[25] B. Testa, A. Pedretti, and G. Vistoli, "Reactions and enzymes in the metabolism of drugs and other xenobiotics," *Drug Discovery Today*, vol. 17, no. 11, pp. 549–560, 2012.

[26] U. M. Zanger and M. Schwab, "Cytochrome p450 enzymes in drug metabolism: Regulation of gene expression, enzyme activities, and impact of genetic variation," *Pharmacology & Therapeutics*, vol. 138, no. 1, pp. 103–141, 2013.

[27] J. Zaretzki, M. Matlock, and S. J. Swamidass, "Xenosite: Accurately predicting cyp-mediated sites of metabolism with neural networks," *Journal of Chemical Information and Modeling*, vol. 53, no. 12, pp. 3373–3383, Dec. 2013.

[28] A. M. McDonnell and C. H. Dang, "Basic review of the cytochrome p450 system," *J. Adv. Pract. Oncol.*, vol. 4, no. 4, pp. 263–268, Jul. 2013.

[29] V. A. Dixit, L. A. Lal, and S. R. Agrawal, "Recent advances in the prediction of non-cyp450-mediated drug metabolism," *WIREs Computational Molecular Science*, vol. 7, no. 6, e1323, 2017. eprint: https://wires.onlinelibrary.wiley.com/doi/pdf/10.1002/wcms.1323.

[30] I. Nobeli, A. D. Favia, and J. M. Thornton, "Protein promiscuity and its implications for biotechnology," *Nature Biotechnology*, vol. 27, no. 2, pp. 157–167, Feb. 2009.

[31] D. S. Tawfik, "Enzyme promiscuity and evolution in light of cellular metabolism," *The FEBS Journal*, vol. 287, no. 7, pp. 1260–1261, 2020. eprint: https://febs.onlinelibrary.wiley.com/doi/pdf/10.1111/febs.15296.

[32] I. Otero-Muras and P. Carbonell, "Automated engineering of synthetic metabolic pathways for efficient biomanufacturing," *Metabolic Engineering*, vol. 63, pp. 61–80, 2021, Tools and Strategies of Metabolic Engineering.

[33] V. Porokhin, S. A. Amin, T. B. Nicks, V. E. Gopinarayanan, N. U. Nair, and S. Hassoun, "Analysis of metabolic network disruption in engineered microbial hosts due to enzyme promiscuity," *Metabolic Engineering Communications*, vol. 12, e00170, 2021.

[34] J. Strutz, K. M. Shebek, L. J. Broadbelt, and K. E. J. Tyo, "Mine 2.0: Enhanced biochemical coverage for peak identification in untargeted metabolomics," *Bioinformatics*, vol. 38, no. 13, pp. 3484–3487, May 2022. eprint: https://academic.oup.com/bioinformatics/article-pdf/38/13/3484/49883998/btac331.pdf.

[35] T. Duigou, M. du Lac, P. Carbonell, and J.-L. Faulon, "Retrorules: A database of reaction rules for engineering biology," *Nucleic Acids Research*, vol. 47, no. D1, pp. D1229–D1235, Oct. 2018. eprint: https://academic.oup.com/nar/article-pdf/47/D1/D1229/27436292/gky940.pdf.

[36] M. Kotera, Y. Okuno, M. Hattori, S. Goto, and M. Kanehisa, "Computational assignment of the ec numbers for genomic-scale analysis of enzymatic reactions," *Journal of the American Chemical Society*, vol. 126, no. 50, pp. 16 487–16 498, Dec. 2004.

[37] T. V. Sivakumar, V. Giri, J. H. Park, T. Y. Kim, and A. Bhaduri, "Reactpred: A tool to predict and analyze biochemical reactions," *Bioinformatics*, vol. 32, no. 22, pp. 3522–3524, Aug. 2016. eprint: https://academic.oup.com/bioinformatics/article-pdf/32/22/3522/49027026/bioinformatics_32_22_3522.pdf.

[38] D. J. Newman and G. M. Cragg, "Natural products as sources of new drugs over the nearly four decades from 01/1981 to 09/2019," *Journal of Natural Products*, vol. 83, no. 3, pp. 770–803, Mar. 2020.

[39] A. L. Demain, "Importance of microbial natural products and the need to revitalize their discovery," *Journal of Industrial Microbiology and Biotechnology*, vol. 41, no. 2, pp. 185–201, Feb. 2014. eprint: https://academic.oup.com/jimb/article-pdf/41/2/185/36813622/jimb0185.pdf.

[40] M. Jangra et al., "A broad-spectrum lasso peptide antibiotic targeting the bacterial ribosome," *Nature*, vol. 640, no. 8060, pp. 1022–1030, Apr. 2025.

[41] W. Leimgruber, A. D. Batcho, and F. Schenker, "The structure of anthramycin," *Journal of the American Chemical Society*, vol. 87, no. 24, pp. 5793–5795, Dec. 1965.

[42] B. Shen, "A new golden age of natural products drug discovery," *Cell*, vol. 163, no. 6, pp. 1297–1300, Dec. 2015.

[43] R. Chen, H. L. Wong, and B. P. Burns, "New approaches to detect biosynthetic gene clusters in the environment," *Medicines*, vol. 6, no. 1, 2019.

[44] I. Kjærbølling, U. H. Mortensen, T. Vesth, and M. R. Andersen, "Strategies to establish the link between biosynthetic gene clusters and secondary metabolites," *Fungal Genetics and Biology*, vol. 130, pp. 107–121, 2019.

[45] G. D. Hannigan et al., "A deep learning genome-mining strategy for biosynthetic gene cluster prediction," *Nucleic Acids Research*, vol. 47, no. 18, e110–e110, Aug. 2019. eprint: https://academic.oup.com/nar/article-pdf/47/18/e110/30070680/gkz654.pdf.

[46] C. Rios-Martinez, N. Bhattacharya, A. P. Amini, L. Crawford, and K. K. Yang, "Deep self-supervised learning for biosynthetic gene cluster detection and product classification," *PLOS Computational Biology*, vol. 19, no. 5, pp. 1–14, May 2023.

[47] Z. Du, N. Zhong, and J. Li, "Enhancing gene cluster identification and classification in bacterial genomes through synonym replacement and deep learning," in *2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2024, pp. 19–24.

[48] K. Blin et al., "Antismash 7.0: New and improved predictions for detection, regulation, chemical structures and visualisation," *Nucleic Acids Research*, vol. 51, no. W1, W46–W50, May 2023. eprint: https://academic.oup.com/nar/article-pdf/51/W1/W46/50736745/gkad344.pdf.

[49] K. Blin et al., "Antismash 8.0: Extended gene cluster detection capabilities and analyses of chemistry, enzymology, and regulation," *Nucleic Acids Research*, vol. 53, no. W1, W32–W38, Apr. 2025. eprint: https://academic.oup.com/nar/article-pdf/53/W1/W32/63005537/gkaf334.pdf.

[50] S. A. Kautsar, K. Blin, S. Shaw, T. Weber, and M. H. Medema, "Big-fam: The biosynthetic gene cluster families database," *Nucleic Acids Research*, vol. 49, no. D1, pp. D490–D497, Oct. 2020. eprint: https://academic.oup.com/nar/article-pdf/49/D1/D490/35364220/gkaa812.pdf.

[51] S. A. Kautsar, J. J. J. van der Hooft, D. de Ridder, and M. H. Medema, "Big-slice: A highly scalable tool maps the diversity of 1.2 million biosynthetic gene clusters," *GigaScience*, vol. 10, no. 1, giaa154, Jan. 2021. eprint: https://academic.oup.com/gigascience/article-pdf/10/1/giaa154/60688704/giaa154.pdf.

[52] P. Cimermancic et al., "Insights into secondary metabolism from a global analysis of prokaryotic biosynthetic gene clusters," *Cell*, vol. 158, no. 2, pp. 412–421, Jul. 2014.

[53] J. R. Doroghazi et al., "A roadmap for natural product discovery based on large-scale genomics and metabolomics," *Nature Chemical Biology*, vol. 10, no. 11, pp. 963–968, Nov. 2014.

[54] T. Leao et al., "Comparative genomics uncovers the prolific and distinctive metabolic potential of the cyanobacterial genus ¡i¿moorea¡/i¿," *Proceedings of the National Academy of Sciences*, vol. 114, no. 12, pp. 3198–3203, 2017. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1618556114.

[55] J. C. Navarro-Muñoz et al., "A computational framework to explore large-scale biosynthetic diversity," *Nature Chemical Biology*, vol. 16, no. 1, pp. 60–68, Jan. 2020.

[56] A. W. Goering et al., "Metabologenomics: Correlation of microbial gene clusters with metabolites drives discovery of a nonribosomal peptide with an unusual amino acid monomer," *ACS Central Science*, vol. 2, no. 2, pp. 99–108, Feb. 2016.

[57] O. Trott and A. J. Olson, "AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading," *Journal of Computational Chemistry*, vol. 31, no. 2, pp. 455–461, 2010. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jcc.21334.

[58] S. Forli, R. Huey, M. E. Pique, M. F. Sanner, D. S. Goodsell, and A. J. Olson, "Computational protein–ligand docking and virtual drug screening with the autodock suite," *Nature Protocols*, vol. 11, no. 5, pp. 905–919, May 2016.

[59] L. H. S. Santos, R. S. Ferreira, and E. R. Caffarena, "Integrating molecular docking and molecular dynamics simulations," in *Docking Screens for Drug Discovery*. New York, NY: Springer New York, 2019, pp. 13–34, ISBN: 978-1-4939-9752-7.

[60] Schrödinger, LLC, "Schrödinger: Maestro," Release 2023-2, Feb. 2023.

[61] M. P. Jacobson et al., "A hierarchical approach to all-atom protein loop prediction," *Proteins: Structure, Function, and Bioinformatics*, vol. 55, no. 2, pp. 351–367, 2004. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/prot.10613.

[62] M. P. Jacobson, R. A. Friesner, Z. Xiang, and B. Honig, "On the role of the crystal environment in determining protein side-chain conformations," *Journal of Molecular Biology*, vol. 320, no. 3, pp. 597–608, 2002.

[63] S. Zhang, H. Tong, J. Xu, and R. Maciejewski, "Graph convolutional networks: A comprehensive review," *Computational Social Networks*, vol. 6, no. 1, p. 11, Nov. 2019.

[64] J. Zhou et al., "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, 2020.

[65] P. L. Donti, B. Amos, and J. Z. Kolter, "Task-based end-to-end model learning in stochastic optimization," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17, NIPS, 2017, pp. 5490–5500.

[66] M. Defferrard, X. Bresson, and P. Vandergheynst, *Convolutional neural networks on graphs with fast localized spectral filtering*, 2017. arXiv: 1606.09375 [cs.LG].

[67] H. W. Kim et al., "Npclassifier: A deep neural network-based structural classification tool for natural products," *Journal of Natural Products*, vol. 84, no. 11, pp. 2795–2807, 2021, PMID: 34662515. eprint: https://doi.org/10.1021/acs.jnatprod.1c00399.

[68] K. Blin, S. Shaw, M. H. Medema, and T. Weber, "The antismash database version 4: Additional genomes and bgcs, new sequence-based searches and more," *Nucleic Acids Research*, vol. 52, no. D1, pp. D586–D589, Oct. 2023. eprint: https://academic.oup.com/nar/article-pdf/52/D1/D586/55040357/gkad984.pdf.

[69] Y. Wang, P. Xue, M. Cao, T. Yu, S. T. Lane, and H. Zhao, "Directed evolution: Methodologies and applications," *Chemical Reviews*, vol. 121, no. 20, pp. 12 384–12 444, Oct. 2021.

[70] H. W. Hellinga, "Rational protein design: Combining theory and experiment," *Proceedings of the National Academy of Sciences*, vol. 94, no. 19, pp. 10 015–10 017, 1997. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.94.19.10015.

[71] I. V. Korendovych, U. T. Bornscheuer, and M. Höhne, "Rational and semirational protein design," in *Protein Engineering: Methods and Protocols*. New York, NY: Springer New York, 2018, pp. 15–23, ISBN: 978-1-4939-7366-8.

[72] A. P. de Abreu et al., "An approach for engineering peptides for competitive inhibition of the sars-cov-2 spike protein," *Molecules*, vol. 29, no. 7, 2024.

[73] L. P. Scott, J. Chahine, and J. R. Ruggiero, "Using genetic algorithm to design protein sequence," *Applied Mathematics and Computation*, vol. 200, no. 1, pp. 1–9, 2008.

[74] Q. Shao, Y. Jiang, and Z. J. Yang, "EnzyHTP computational directed evolution with adaptive resource allocation," *Journal of Chemical Information and Modeling*, vol. 63, no. 17, pp. 5650–5659, Sep. 2023.

[75] K. K. Yang, Z. Wu, and F. H. Arnold, "Machine-learning-guided directed evolution for protein engineering," *Nature Methods*, vol. 16, no. 8, pp. 687–694, Aug. 2019.

[76] D. Belanger et al., "Biological sequences design using batched bayesian optimization," in *NeurIPS workshop on Bayesian Deep Learning*, 2019.

[77] J. Linder and G. Seelig, "Fast activation maximization for molecular sequence design," *BMC Bioinformatics*, vol. 22, no. 1, Oct. 2021.

[78] J. Abramson et al., "Accurate structure prediction of biomolecular interactions with alphafold 3," *Nature*, vol. 630, no. 8016, pp. 493–500, Jun. 2024.

[79] Chai Discovery, "Chai-1: Decoding the molecular interactions of life," *bioRxiv*, 2024. eprint: https://www.biorxiv.org/content/early/2024/10/11/2024.10.10.615955.full.pdf.

[80] J. Wohlwend et al., "Boltz-1 democratizing biomolecular interaction modeling," *bioRxiv*, 2025. eprint: https://www.biorxiv.org/content/early/2025/05/06/2024.11.19.624167.full.pdf.

[81] Y. Tang, R. Moretti, and J. Meiler, "Recent advances in automated structure-based de novo drug design," *Journal of Chemical Information and Modeling*, vol. 64, no. 6, pp. 1794–1805, Mar. 2024.

[82] J. Jiménez-Luna, F. Grisoni, and G. Schneider, "Drug discovery with explainable artificial intelligence," *Nature Machine Intelligence*, vol. 2, no. 10, pp. 573–584, Oct. 2020.

[83] J. Zaretzki, P. Rydberg, C. Bergeron, K. P. Bennett, L. Olsen, and C. M. Breneman, "Rs-predictor models augmented with smartcyp reactivities: Robust metabolic regioselectivity predictions for nine cyp isozymes," *Journal of Chemical Information and Modeling*, vol. 52, no. 6, pp. 1637–1659, Jun. 2012.

[84] P. Rydberg, D. E. Gloriam, J. Zaretzki, C. Breneman, and L. Olsen, "Smartcyp: A 2d method for prediction of cytochrome p450-mediated drug metabolism," *ACS Medicinal Chemistry Letters*, vol. 1, no. 3, pp. 96–100, Jun. 2010.

[85] A. Bateman et al., "The pfam protein families database," *Nucleic Acids Research*, vol. 32, no. suppl_1, pp. D138–D141, Jan. 2004. eprint: https://academic.oup.com/nar/article-pdf/32/suppl_1/D138/7622065/gkh121.pdf.

[86] K. K. Yang, N. Fusi, and A. X. Lu, "Convolutions are competitive with transformers for protein sequence pretraining," *bioRxiv*, 2024. eprint: https://www.biorxiv.org/content/early/2024/02/05/2022.05.19.492714.full.pdf.

[87] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, *Bert: Pre-training of deep bidirectional transformers for language understanding*, 2019. arXiv: 1810.04805 [cs.CL].

[88] P. Agrawal, S. Khater, M. Gupta, N. Sain, and D. Mohanty, "Rippminer: A bioinformatics resource for deciphering chemical structures of ripps based on prediction of cleavage and cross-links," *Nucleic Acids Research*, vol. 45, no. W1, W80–W88, May 2017. eprint: https://academic.oup.com/nar/article-pdf/45/W1/W80/18137594/gkx408.pdf.

[89] E. J. N. Helfrich et al., "Automated structure prediction of trans-acyltransferase polyketide synthase products," *Nature Chemical Biology*, vol. 15, no. 8, pp. 813–821, Aug. 2019.

[90] M. A. Skinnider et al., "Comprehensive prediction of secondary metabolite structure and biological activity from microbial genome sequences," *Nature Communications*, vol. 11, no. 1, p. 6058, Nov. 2020.

[91] K. Dührkop et al., "Systematic classification of unknown metabolites using high-resolution fragmentation mass spectra," *Nature Biotechnology*, vol. 39, no. 4, pp. 462–471, Apr. 2021.

[92] M. Larralde and G. Zeller, "Machine learning inference of natural product chemistry across biosynthetic gene cluster types," *bioRxiv*, 2025. eprint: https://www.biorxiv.org/content/early/2025/03/15/2025.03.13.642868.full.pdf.

[93] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, Dec. 2018. eprint: https://academic.oup.com/jrsssb/article-pdf/58/1/267/49098631/jrsssb_58_1_267.pdf.

[94] The UniProt Consortium, "UniProt: the universal protein knowledgebase in 2021," *Nucleic Acids Research*, vol. 49, no. D1, pp. D480–D489, Nov. 2020. eprint: https://academic.oup.com/nar/article-pdf/49/D1/D480/35364103/gkaa1100.pdf.

[95] J. Jumper et al., "Highly accurate protein structure prediction with AlphaFold," *Nature*, vol. 596, no. 7873, pp. 583–589, Aug. 2021.

[96] M. Varadi et al., "AlphaFold Protein Structure Database: massively expanding the structural coverage of protein-sequence space with high-accuracy models," *Nucleic Acids Research*, vol. 50, no. D1, pp. D439–D444, Nov. 2021. eprint: https://academic.oup.com/nar/article-pdf/50/D1/D439/43502749/gkab1061.pdf.

[97] D. E. Kim, D. Chivian, and D. Baker, "Protein structure prediction and analysis using the Robetta server," *Nucleic Acids Research*, vol. 32, no. suppl2, W526–W531, Jul. 2004. eprint: https://academic.oup.com/nar/article-pdf/32/suppl_2/W526/6211563/gkh468.pdf.

[98] M. Baek et al., "Accurate prediction of protein structures and interactions using a three-track neural network," *Science*, vol. 373, no. 6557, pp. 871–876, 2021. eprint: https://www.science.org/doi/pdf/10.1126/science.abj8754.

[99] Schrödinger, LLC, "The PyMOL molecular graphics system," Version 2.5.4, Aug. 2022.

[100] Y. Feng et al., "Structural insight into the type-II mitochondrial NADH dehydrogenases," *Nature*, vol. 491, no. 7424, pp. 478–482, Nov. 2012.

[101] C. Camacho et al., "Blast+: Architecture and applications," *BMC Bioinformatics*, vol. 10, no. 1, p. 421, Dec. 2009.

[102] D. A. DiRocco, K. Dykstra, S. Krska, P. Vachal, D. V. Conway, and M. Tudge, "Late-stage functionalization of biologically active heterocycles through photoredox catalysis," *Angewandte Chemie International Edition*, vol. 53, no. 19, pp. 4802–4806, 2014. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/anie.201402023.

[103] T. Cernak, K. D. Dykstra, S. Tyagarajan, P. Vachal, and S. W. Krska, "The medicinal chemist's toolbox for late stage functionalization of drug-like molecules," *Chem. Soc. Rev.*, vol. 45, pp. 546–576, 3 2016.

[104] M. C. White and J. Zhao, "Aliphatic c–h oxidations for late-stage functionalization," *Journal of the American Chemical Society*, vol. 140, no. 43, pp. 13 988–14 009, Oct. 2018.

[105] Daylight Chemical Information Systems, Inc., *SMARTS - a language for describing molecular patterns*, https://www.daylight.com/dayhtml/doc/theory/theory.smarts.html, Accessed: 2024-01-11.

[106] RDKit contributors, "RDKit: Open-source cheminformatics," Release 2022.09.1, Oct. 2022.

[107] S. Kim et al., "PubChem 2023 update," *Nucleic Acids Research*, vol. 51, no. D1, pp. D1373–D1380, Oct. 2022. eprint: https://academic.oup.com/nar/article-pdf/51/D1/D1373/48441598/gkac956.pdf.

[108] G. Madhavi Sastry, M. Adzhigirey, T. Day, R. Annabhimoju, and W. Sherman, "Protein and ligand preparation: Parameters, protocols, and influence on virtual screening enrichments," *Journal of Computer-Aided Molecular Design*, vol. 27, no. 3, pp. 221–234, Mar. 2013.

[109] Schrödinger, LLC, "Ligprep," Release 2023-2, Feb. 2023.

[110] A. Waterhouse et al., "SWISS-MODEL: homology modelling of protein structures and complexes," *Nucleic Acids Research*, vol. 46, no. W1, W296–W303, May 2018. eprint: https://academic.oup.com/nar/article-pdf/46/W1/W296/25110428/gky427.pdf.

[111] S. Bienert et al., "The SWISS-MODEL Repository—new features and functionality," *Nucleic Acids Research*, vol. 45, no. D1, pp. D313–D319, Nov. 2016. eprint: https://academic.oup.com/nar/article-pdf/45/D1/D313/8846950/gkw1132.pdf.

[112] N. Guex, M. C. Peitsch, and T. Schwede, "Automated comparative protein structure modeling with SWISS-MODEL and Swiss-PdbViewer: A historical perspective," *ELECTROPHORESIS*, vol. 30, no. S1, S162–S173, 2009. eprint: https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/pdf/10.1002/elps.200900140.

[113] G. Studer, C. Rempfer, A. M. Waterhouse, R. Gumienny, J. Haas, and T. Schwede, "QMEANDisCo—distance constraints applied on model quality estimation," *Bioinformatics*, vol. 36, no. 6, pp. 1765–1771, Nov. 2019. eprint: https://academic.oup.com/bioinformatics/article-pdf/36/6/1765/50554203/bioinformatics_36_6_1765.pdf.

[114] M. Bertoni, F. Kiefer, M. Biasini, L. Bordoli, and T. Schwede, "Modeling protein quaternary structure of homo- and hetero-oligomers beyond binary interactions by homology," *Scientific Reports*, vol. 7, no. 1, p. 10 480, Sep. 2017.

[115] M. Wiederstein and M. J. Sippl, "ProSA-web: Interactive web service for the recognition of errors in three-dimensional structures of proteins," *Nucleic Acids Research*, vol. 35, no. suppl2, W407–W410, Jul. 2007. eprint: https://academic.oup.com/nar/article-pdf/35/suppl_2/W407/9584436/gkm290.pdf.

[116] M. J. Sippl, "Recognition of errors in three-dimensional structures of proteins," *Proteins: Structure, Function, and Bioinformatics*, vol. 17, no. 4, pp. 355–362, 1993. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/prot.340170404.

[117] C. Pommié, S. Levadoux, R. Sabatier, G. Lefranc, and M.-P. Lefranc, "Imgt standardized criteria for statistical analysis of immunoglobulin v-region amino acid properties," *Journal of Molecular Recognition*, vol. 17, no. 1, pp. 17–32, 2004. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jmr.647.

[118] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, *How powerful are graph neural networks?* 2019. arXiv: 1810.00826 [cs.LG].

[119] D. Duvenaud et al., "Convolutional networks on graphs for learning molecular fingerprints," in *Proceedings of the 29th International Conference on Neural Infor-*

mation Processing Systems - Volume 2, ser. NIPS'15, Montreal, Canada: MIT Press, 2015, pp. 2224–2232.

[120] M. Fey and J. E. Lenssen, *Fast graph representation learning with pytorch geometric*, 2019. arXiv: 1903.02428 [cs.LG].

[121] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, 2017. arXiv: 1412.6980 [cs.LG].

[122] K. /. K. Laboratories, *Kegg atom types*, https://www.genome.jp/kegg/reaction/KCF.html, Accessed: 2025-11-19, 2025.

[123] M. Kotera et al., "Kcf-s: Kegg chemical function and substructure for improved interpretability and prediction in chemical bioinformatics," *BMC Systems Biology*, vol. 7, no. 6, S2, Dec. 2013.

[124] M. Sato, H. Suetake, and M. Kotera, "Kcf-convoy: Efficient python package to convert kegg chemical function and substructure fingerprints," *bioRxiv*, 2018. eprint: https://www.biorxiv.org/content/early/2018/10/24/452383.full.pdf.

[125] B. D. McKay and A. Piperno, "Practical graph isomorphism, ii," *Journal of Symbolic Computation*, vol. 60, pp. 94–112, 2014.

[126] C. Rücker and M. Meringer, "How many organic compounds are graph-theoretically nonplanar?" *MATCH Communications in Mathematical and in Computer Chemistry*, vol. 45, pp. 153–172, 2002.

[127] J. Torán and F. Wagner, "The complexity of planar graph isomorphism," 2009, pp. 60–82.

[128] T. Fawcett, "An introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006, ROC Analysis in Pattern Recognition.

[129] M. E. Beber et al., "Equilibrator 3.0: A database solution for thermodynamic constant estimation," *Nucleic Acids Research*, vol. 50, no. D1, pp. D603–D609, Nov. 2021. eprint: https://academic.oup.com/nar/article-pdf/50/D1/D603/42057662/gkab1106.pdf.

[130] E. Noor, H. S. Haraldsdóttir, R. Milo, and R. M. T. Fleming, "Consistent estimation of gibbs energy using component contributions," *PLoS Computational Biology*, vol. 9, no. 7, e1003098, 2013.

[131] Y. Kim, J. Y. Ryu, H. U. Kim, W. D. Jang, and S. Y. Lee, "A deep learning approach to evaluate the feasibility of enzymatic reactions generated by retrobiosynthesis," *Biotechnology Journal*, vol. 16, no. 5, p. 2 000 605, 2021. eprint:

https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/pdf/10.1002/biot.202000605.

[132] J. Jiang, L.-P. Liu, and S. Hassoun, "Learning graph representations of biochemical networks and its application to enzymatic link prediction," *Bioinformatics*, vol. 37, no. 6, pp. 793–799, Oct. 2020. eprint: https://academic.oup.com/bioinformatics/article-pdf/37/6/793/50356282/btaa881.pdf.

[133] M. Yousofshahi, S. Manteiga, C. Wu, K. Lee, and S. Hassoun, "Proximal: A method for prediction of xenobiotic metabolism," *BMC Systems Biology*, vol. 9, no. 1, p. 94, Dec. 2015.

[134] S. A. Amin, E. Chavez, V. Porokhin, N. U. Nair, and S. Hassoun, "Towards creating an extended metabolic model (emm) for e. coli using enzyme promiscuity prediction and metabolomics data," *Microbial Cell Factories*, vol. 18, no. 1, p. 109, Jun. 2019.

[135] J. M. Monk et al., "Iml1515, a knowledgebase that computes escherichia coli traits," *Nature Biotechnology*, vol. 35, no. 10, pp. 904–908, Oct. 2017.

[136] A. C. Guo et al., "Ecmdb: The e. coli metabolome database," *Nucleic Acids Research*, vol. 41, no. D1, pp. D625–D630, Oct. 2012. eprint: https://academic.oup.com/nar/article-pdf/41/D1/D625/3732442/gks992.pdf.

[137] T. Sajed et al., "Ecmdb 2.0: A richer resource for understanding the biochemistry of e. coli," *Nucleic Acids Research*, vol. 44, no. D1, pp. D495–D501, Oct. 2015. eprint: https://academic.oup.com/nar/article-pdf/44/D1/D495/9482189/gkv1060.pdf.

[138] N. Tepper, E. Noor, D. Amador-Noguez, H. S. Haraldsdóttir, R. Milo, J. Rabinowitz, et al., "Steady-state metabolite concentrations reflect a balance between maximizing enzyme efficiency and minimizing total metabolite load," *PLOS ONE*, vol. 8, no. 9, e75370, 2013.

[139] Z. Cheng, J. Jiang, H. Wu, Z. Li, and Q. Ye, "Enhanced production of 3-hydroxypropionic acid from glucose via malonyl-coa pathway by engineered escherichia coli," *Bioresource Technology*, vol. 200, pp. 897–904, 2016.

[140] C. Rathnasingh, S. M. Raj, Y. Lee, C. Catherine, S. Ashok, and S. Park, "Production of 3-hydroxypropionic acid via malonyl-coa pathway using recombinant escherichia coli strains," *Journal of Biotechnology*, vol. 157, no. 4, pp. 633–640, 2012, Special Issue: IBS2010 Part II (Biotechnology for a more sustainable environment decontamination and energy production).

[141] Q. Wang, P. Yang, M. Xian, L. Feng, J. Wang, and G. Zhao, "Metabolic engineering of escherichia coli for poly(3-hydroxypropionate) production from glycerol and glucose," *Biotechnology Letters*, vol. 36, no. 11, pp. 2257–2262, Nov. 2014.

[142] M. M. Zdouc et al., "Mibig 4.0: Advancing biosynthetic gene cluster curation through global collaboration," *Nucleic Acids Research*, vol. 53, no. D1, pp. D678–D690, Dec. 2024. eprint: https://academic.oup.com/nar/article-pdf/53/D1/D678/61003145/gkae1115.pdf.

[143] C. Bağcı et al., "Bgc atlas: A web resource for exploring the global chemical diversity encoded in bacterial genomes," *Nucleic Acids Research*, vol. 53, no. D1, pp. D618–D624, Oct. 2024. eprint: https://academic.oup.com/nar/article-pdf/53/D1/D618/60197927/gkae953.pdf.

[144] L. Richardson et al., "Mgnify: The microbiome sequence data analysis resource in 2023," *Nucleic Acids Research*, vol. 51, no. D1, pp. D753–D759, Dec. 2022. eprint: https://academic.oup.com/nar/article-pdf/51/D1/D753/48441116/gkac1080.pdf.

[145] E. A. Freeman and G. G. Moisen, "A comparison of the performance of threshold criteria for binary classification in terms of predicted prevalence and kappa," *Ecological Modelling*, vol. 217, no. 1, pp. 48–58, 2008.

[146] C. Parker, "An analysis of performance measures for binary classifiers," in *2011 IEEE 11th International Conference on Data Mining*, 2011, pp. 517–526.

[147] E. J. Martin, V. R. Polyakov, L. Tian, and R. C. Perez, "Profile-qsar 2.0: Kinase virtual screening accuracy comparable to four-concentration ic50s for realistically novel compounds," *Journal of Chemical Information and Modeling*, vol. 57, no. 8, pp. 2077–2088, 2017, PMID: 28651433. eprint: https://doi.org/10.1021/acs.jcim.7b00166.

[148] G. S. Kohli, U. John, F. M. Van Dolah, and S. A. Murray, "Evolutionary distinctiveness of fatty acid and polyketide synthesis in eukaryotes," *The ISME Journal*, vol. 10, no. 8, pp. 1877–1890, Jan. 2016. eprint: https://academic.oup.com/ismej/article-pdf/10/8/1877/56172496/41396_2016_article_bfismej2015263.pdf.

[149] J. J. J. van der Hooft, H. Mohimani, A. Bauermeister, P. C. Dorrestein, K. R. Duncan, and M. H. Medema, "Linking genomics and metabolomics to chart specialized metabolic diversity," *Chem. Soc. Rev.*, vol. 49, pp. 3297–3314, 11 2020.

[150] S. A. Kautsar, H. G. Suarez Duran, K. Blin, A. Osbourn, and M. H. Medema, "Plantismash: Automated identification, annotation and expression analysis of plant biosynthetic gene clusters," *Nucleic Acids Research*, vol. 45, no. W1, W55–W63, Apr. 2017. eprint: https://academic.oup.com/nar/article-pdf/45/W1/W55/18137272/gkx305.pdf.

[151] E. Del Pup et al., "Plantismash 2.0: Improvements to detection, annotation, and prioritization of plant biosynthetic gene clusters," *bioRxiv*, 2025. eprint: https://www.biorxiv.org/content/early/2025/10/28/2025.10.28.683968.full.pdf.

[152] T. Kwon and B. T. Hovde, "Global characterization of biosynthetic gene clusters in non-model eukaryotes using domain architectures," *Scientific Reports*, vol. 14, no. 1, p. 1534, Jan. 2024.

[153] M. Alcalde, M. Zumarraga, J. Polaina, A. Ballesteros, and F. J. Plou, "Combinatorial saturation mutagenesis by in vivo overlap extension for the engineering of fungal laccases," *Combinatorial Chemistry & High Throughput Screening*, vol. 9, no. 10, pp. 719–727, 2006.

[154] J. L. Hickey, D. Sindhikara, S. L. Zultanski, and D. M. Schultz, "Beyond 20 in the 21st century: Prospects and challenges of non-canonical amino acids in peptide drug discovery," *ACS Medicinal Chemistry Letters*, vol. 14, no. 5, pp. 557–565, May 2023.

[155] S. Biswas, G. Khimulya, E. C. Alley, K. M. Esvelt, and G. M. Church, "Low-n protein engineering with data-efficient deep learning," *Nature Methods*, vol. 18, no. 4, pp. 389–396, Apr. 2021.

[156] A. Madani et al., "Large language models generate functional protein sequences across diverse families," *Nature Biotechnology*, vol. 41, no. 8, pp. 1099–1106, Aug. 2023.

[157] V. Bagal, R. Aggarwal, P. K. Vinod, and U. D. Priyakumar, "Molgpt: Molecular generation using a transformer-decoder model," *Journal of Chemical Information and Modeling*, vol. 62, no. 9, pp. 2064–2076, May 2022.

[158] Y. Xu et al., "Deep learning for molecular generation," *Future Medicinal Chemistry*, vol. 11, no. 6, pp. 567–597, 2019, PMID: 30698019. eprint: https://doi.org/10.4155/fmc-2018-0358.

[159] N. Gruver et al., *Protein design with guided discrete diffusion*, 2023. arXiv: 2305.20009 [cs.LG].

[160] C. Vignac, I. Krawczuk, A. Siraudin, B. Wang, V. Cevher, and P. Frossard, "Digress: Discrete denoising diffusion for graph generation," in *The Eleventh International Conference on Learning Representations*, 2023.

[161] A. Dodaro et al., "Post-docking refinement of peptide or protein-rna complexes using thermal titration molecular dynamics (ttmd): A stability insight," *Journal of Chemical Information and Modeling*, vol. 65, no. 3, pp. 1441–1452, 2025, PMID: 39818831. eprint: https://doi.org/10.1021/acs.jcim.4c01393.

[162] L. F. Di Costanzo et al., "Structural insights into temperature-dependent dynamics of metpsc1, a miniaturized electron-transfer protein," *Journal of Inorganic Biochemistry*, vol. 264, p. 112 810, 2025.

[163] P. Li, M. Shi, Y. Wang, Q. Liu, X. Du, and X. Wang, "Ph-dependent assembly and stability of toll-like receptor 3/dsrna signaling complex: Insights from constant ph molecular dynamics and metadynamics simulations," *Advanced Science*, vol. 12, no. 1, p. 2 411 445, 2025. eprint: https://advanced.onlinelibrary.wiley.com/doi/pdf/10.1002/advs.202411445.

[164] A. Kalia et al., "The role of structural, pharmacokinetic and energy properties in the high-throughput prediction of redox potentials for organic molecules with experimental calibration," *ChemRxiv*, 2024.

[165] T. Duigou, P. Meyer, and J.-L. Faulon, "Retrorules 2026: An expanded database combining biochemical and organic reaction templates for pathway discovery," *Nucleic Acids Research*, gkaf1261, Dec. 2025. eprint: https://academic.oup.com/nar/advance-article-pdf/doi/10.1093/nar/gkaf1261/65801091/gkaf1261.pdf.

[166] O. Méndez-Lucio, C. Nicolaou, and B. Earnshaw, *Mole: A molecular foundation model for drug discovery*, 2022. arXiv: 2211.02657 [q-bio.QM].

[167] W. Hu et al., *Strategies for pre-training graph neural networks*, 2020. arXiv: 1905.12265 [cs.LG].

[168] Y. Wang, J. Wang, Z. Cao, and A. Barati Farimani, "Molecular contrastive learning of representations via graph neural networks," *Nature Machine Intelligence*, vol. 4, no. 3, pp. 279–287, Mar. 2022.

[169] K. Yang et al., "Analyzing learned molecular representations for property prediction," *Journal of Chemical Information and Modeling*, vol. 59, no. 8, pp. 3370–3388, Aug. 2019.